Journal of applied logics (Print) ISSN 2631-9810
Journal of applied logics (Online) ISSN 2631-9829

Journal of Applied Logics

The IfCoLog Journal of Logics and their Applications

Volume 8 • Issue 6 • July 2021

Special Issue on Formal Argumentation

Guest Editors

Dov M. Gabbay
Massimiliano Giacomin
Guillermo R. Simari
Matthias Thimm

Available online at
www.collegepublications.co.uk/journals/ifcolog/
Free open access

Journal of Applied Logics - IfCoLog
Journal of Logics and their Applications

Volume 8, Number 6

July 2021

Disclaimer

Statements of fact and opinion in the articles in Journal of Applied Logics - IfCoLog Journal of Logics and their Applications (JALs-FLAP) are those of the respective authors and contributors and not of the JALs-FLAP. Neither College Publications nor the JALs-FLAP make any representation, express or implied, in respect of the accuracy of the material in this journal and cannot accept any legal responsibility or liability for any errors or omissions that may be made. The reader should make his/her own evaluation as to the appropriateness or otherwise of any experimental technique described.

ISBN 978-1-84890-371-5
ISSN (E) 2631-9829
ISSN (P) 2631-9810

College Publications
Scientific Director: Dov Gabbay
Managing Director: Jane Spurr

http://www.collegepublications.co.uk

Editorial Board

Scope and Submissions

This journal considers submission in all areas of pure and applied logic, including:

pure logical systems
proof theory
constructive logic
categorical logic
modal and temporal logic
model theory
recursion theory
type theory
nominal theory
nonclassical logics
nonmonotonic logic
numerical and uncertainty reasoning
logic and AI
foundations of logic programming
belief change/revision
systems of knowledge and belief
logics and semantics of programming
specification and verification
agent theory
databases

dynamic logic
quantum logic
algebraic logic
logic and cognition
probabilistic logic
logic and networks
neuro-logical systems
complexity
argumentation theory
logic and computation
logic and language
logic engineering
knowledge-based systems
automated reasoning
knowledge representation
logic in hardware and VLSI
natural language
concurrent computation
planning

This journal will also consider papers on the application of logic in other subject areas: philosophy, cognitive science, physics etc. provided they have some formal content.

Submissions should be sent to Jane Spurr (jane@janespurr.net) as a pdf file, preferably compiled in LaTeX using the IFCoLog class file.

Contents

ARTICLES

Preface 1335
Dov M. Gabbay, Massimiliano Giacomin, Guillermo R. Simari and Matthias Thimm

Higher-Order Interactions (Bipolar or not) in Abstract Argumentation: A State of the Art 1339
Claudette Cayrol, Andrea Cohen and Marie-Christine Lagasquie-Schiex

Joint Attacks and Accrual in Argumentation Frameworks 1437
Antonis Bikakis, Andrea Cohen, Wolfgang Dvořák, Giorgos Flouris and Simon Parsons

Collective Acceptability in Abstract Argumentation 1503
Dorothea Baumeister, Daniel Neugebauer and Jörg Rothe

Value-based Argumentation 1543
Katie Atkinson and Trevor Bench-Capon

Weighted Argumentation 1589
Stefano Bistarelli and Francesco Santini

Enforcement in Formal Argumentation 1623
Ringo Baumann, Sylvie Doutre, Jean-Guy Mailly and Johannes P. Wallner

Strategic Argumentation **1679**
Guido Governatori, Michael J. Maher and Francesco Olivieri

On the Incremental Computation of Semantics in Dynamic Argumentation **1749**
Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, Gerardo I. Simari and Guillermo R. Simari

Logic-Based Approaches to Formal Argumentation **1793**
Ofer Arieli, AnneMarie Borg, Jesse Heyninck and Christian Strasser

Preface

Dov M. Gabbay
King's College, London, UK and University of Luxembourg, Luxembourg
dov.gabbay@kcl.ac.uk

Massimiliano Giacomin
Dipartimento di Ingegneria dell'Infomazione, Università degli Studi di Brescia, Italy
massimiliano.giacomin@unibs.it

Guillermo R. Simari
Department of Computer Science and Engineering, Universidad Nacional del Sur (UNS) & Institute for Computer Science and Engineering (ICIC UNS-CONICET), Bahia Blanca, Argentina
grsimari@gmail.com

Matthias Thimm
Institute for Web Science and Technologies, University of Koblenz-Landau, Germany
thimm@uni-koblenz.de

This Special Issue of *IfCoLog Journal of Logics and their Applications* is dedicated to a set of papers that, with other additional contributions, will be part of the second volume of the Handbook of Formal Argumentation. The Handbook represents a continuation of the community effort to produce a series of volumes containing survey articles and personal views of recognized researchers of the field to promote work in the area. As such, a central goal favored by this endeavor is to help students and researchers interested in contributing to Formal Argumentation to access both state-of-the-art and future research perspectives in the field. The aim is to stimulate the work in the area by addressing progress in existing research lines, describing open problems, and presenting emerging topics.

In preparation for this second volume, the authors met in Bertinoro, Italy, at the Bertinoro international Center for informatics (BiCi) in a workshop to discuss

"Current Trends in Formal Argumentation", November 4-6, 2019. During the workshop, the attendants presented drafts of the contents of the future chapters giving rise to many suggestions and improvements in the coordination among the topics to be covered, completing the process of envisioning the new volume in the series.

The papers have been reviewed and the final versions presented here have considered the suggestions made by the specialists. The reviewers' feedback represented an essential contribution to improving the final versions, culminating with the manuscripts submitted by the authors in the winter of 2020/21, and offered here.

The papers accepted can be broadly thought of as part of the general area of formal argumentation, being related to extensions to abstract argumentation, dynamics and dialogues, and meta investigations. Next, we provide summaries of the papers that are part of this special issue.

In *Higher-Order Interactions (Bipolar or not) in Abstract Argumentation: a State of the Art*, C. Cayrol, A. Cohen, M-C. Lagasquie-Schiex, start recalling the essential elements of abstract argumentation, then introducing higher-order attacks and summarizing five existing approaches for these attacks. They continue with a brief introduction to traditional bipolar argumentation frameworks and their three variants related to the three possible types of support. Using the different frameworks presented, the authors introduce extended frameworks using higher-order interactions and analyze some contributions in structured argumentation. The computational issues and applications are also described and analyzed, and a comparative synthesis of all the presented approaches is included.

The article *Joint Attacks and Accrual in Argumentation Frameworks*, A. Bikakis, A. Cohen, W. Dvořák, G. Flouris, and S. Parsons, considers the case when multiple arguments jointly attack another, introduced as "joint attacks". This possibility of analyzing joint attacks represents an extension of abstract argumentation with added expressive power. Various works considering joint attacks are analyzed from various perspectives, which include abstract and structured frameworks. Also, guidelines for future research considering current research on the subject are presented.

In *Collective Acceptability in Abstract Argumentation*, D. Baumeister, D. Neugebauer, J. Rothe explore and survey the various approaches to collective acceptability in multi-agent argumentation, which is related to the problem of collective decision-making in the field of computational social choice that collects contributions from social choice theory, theoretical computer science, and artificial intelligence. Also, the paper describes practical methods for structural aggregation of argumentation frameworks and presents their properties.

Value-based Argumentation, by K. Atkinson and T. Bench-Capon, presents an extension of abstract argumentation known as Value-based Argumentation Framework (VAF), its motivations, a formal description, and its properties. The notion of

Audience-Specific VAF that incorporates to the framework the focus in an audience is presented. Also, an argumentation scheme and its associated critical questions and some of the applications of value-based argumentation that have been implemented are included.

In *Weighted Argumentation*, S. Bistarelli and F. Santini introduce Weighted Argumentation Frameworks (WAFs), summarizing different critical points of their formalization; developing weight-related concepts such as relaxation of attacks, new semantics based on weighted acceptability and relaxation, and real-world applications related to information coming from social networks and reviewing platforms.

Enforcement in Formal Argumentation by R. Baumann, S. Doutre, J-G. Mailly, and J. P. Wallner, offers an overview of the notion of enforcement in abstract argumentation. The authors center their presentation on extension enforcement, its general characterization, and how it can be algorithmically achieved. The premise assumed is that the various changes applied to the structure of the argumentation framework, and to the semantics associated, should be minimal. The complexity of enforcement, and associated algorithms, and a discussion on the feasibility of this approach are presented.

In *Strategic Argumentation*, G. Governatori, M. Maher, F. Olivieri, study games where players have perfect information of the moves players make; however, the information on the possible moves (arguments) that other players have available is incomplete. The authors look at games using logically structured arguments and games using abstract arguments, showing that playing these games can be computationally hard. Also, they consider how corruption can affect the argumentation games, and examine forms of countering it.

On the Incremental Computation of Semantics in Dynamic Argumentation by G. Alfano, F. Parisi, S. Greco, G. I. Simari, and G. R. Simari, examines the efficiency of recomputing extensions of abstract argumentation frameworks and warranted literals from defeasible knowledge bases in dynamic environments. An incremental algorithmic solution is presented, making use of an initial extension of a framework and an update with the aim of identifying a subset of the framework enough to compute an extension after the update. The incremental technique for the computation of extensions of abstract argumentation frameworks is considered, exploring how transferred concepts can be employed in the computation of warranted literals in Defeasible Logic Programming.

In *Logic-Based Approaches to Formal Argumentation*, O. Arieli, A. Borg, J. Heyninck, and C. Strasser, the logical foundations of Dung-style argumentation frameworks are presented. Two perspectives on logic-based methods in the context of argumentation theory are offered. First, a survey of logic-based instantiations of argumentation frameworks is introduced, along with their properties and relations, and then logical methods for studying argumentation dynamics are reviewed. The work is focused to Tarskian logics, based on propositional languages and the associated constructive semantics or syntactic rule-based systems.

In closing, we would like to thank the authors of this special issue for their contributions, the reviewers, and the colleagues for their valuable help in providing comments, suggestions, critiques, and encouragement during the development. The papers included have accomplished our two essential objectives by first providing material for the researcher that is coming to the area of argumentation, and also facilitating their acquisition of the elements to have a good view of the work at the forefront of research, our second goal. Finally, we would like to give special thanks to College Publications and, in particular, to Jane Spurr for her unwavering continued support.

Dov M. Gabbay
Massimiliano Giacomin
Guillermo R. Simari
Matthias Thimm

Received May 2021

Higher-Order Interactions (Bipolar or not) in Abstract Argumentation: A State of the Art

Claudette Cayrol
IRIT, Université Toulouse 3
Toulouse, France

Andrea Cohen
Institute for Computer Science and Engineering, CONICET-UNS
Dept. of Computer Science and Engineering, Universidad Nacional del Sur
Bahía Blanca, Argentina
ac@cs.uns.edu.ar

Marie-Christine Lagasquie-Schiex
IRIT, Université Toulouse 3
Toulouse, France
lagasq@irit.fr

Abstract

In Dung's seminal work, an argumentation framework was defined by a set of abstract arguments and a binary (and also abstract) relation between these arguments, called attack relation and expressing conflicts between arguments. Due to its simplicity and the power of its abstraction, this representation has been intensively used by the community for over 25 years. Another advantage of this approach is the ease with which we can extend the framework, weighting arguments or attacks, using priorities or pre-orderings on the sets of arguments, considering that these interactions are no longer binary ones over the set of arguments (*e.g.* collective attacks), adding new kinds of interactions (*e.g.* supports), and proposing that the targets of these interactions can also be interactions themselves (*i.e.* higher-order interactions).

These last two points are the core of this chapter, in which we present a survey of the proposed approaches existing around the notion of higher-order interactions (attacks and supports) in an abstract argumentation framework.

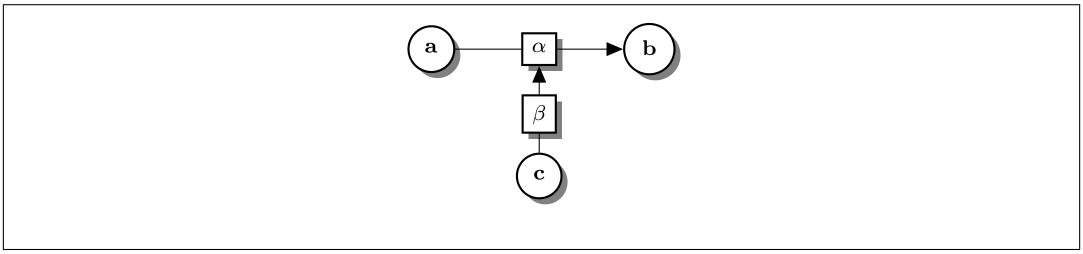


Figure 1: An acyclic recursive framework: arguments are vertices in circles, attacks are directed edges labelled by their name in squares

1 Introduction

Argumentation has become an essential paradigm for knowledge representation and, especially, for reasoning from contradictory information [4; 51] and for formalizing the exchange of arguments between agents in, *e.g.* , negotiation [6] (see [88] for a general overview on the role of argumentation in AI). Formal abstract frameworks have greatly eased the modelling and study of argumentation. For instance, a Dung's argumentation framework (AF) [51] consists of a collection of arguments interacting with each other through an attack relation, enabling to determine "acceptable" sets of arguments called *extensions*.

A natural generalization of Dung's argumentation frameworks consists in allowing higher-order attacks (also called recursive attacks in the relevant literature) that target other attacks. Here is an example from the legal domain, borrowed from [8].

Example 1 ([8])**.** *The lawyer says that the defendant did not have intention to kill the victim (Argument b). The prosecutor says that the defendant threw a sharp knife towards the victim (Argument a). So, there is an attack from a to b, denoted by α. And the intention to kill should be inferred. Then, the lawyer says that the defendant was in a habit of throwing the knife at his wife's foot once drunk. This latter argument (Argument c) is better considered as attacking the attack from a to b, rather than argument a itself (so there is now another attack from c to α, denoted by β). Now the prosecutor's argumentation seems no longer sufficient for proving the intention to kill. This example is represented as a recursive framework in Figure 1.*

The idea of encompassing attacks to attacks in abstract argumentation frameworks was first considered in [13] in the context of an extended framework handling argument strengths and their propagation. Then, a semantics for *recursive frameworks* was introduced in [69], motivated by the fact that attacks to attacks come from preferences between conflicting arguments. More recently, recursive frameworks have been studied in [11] under the name of *AFRA* (Argumentation Framework with Recursive Attacks), extending Modgil's work by considering higher-order attacks and not only second-order attacks (interactions can be either attacks between arguments or attacks from an argument to another attack at any level). Then, the $AFRA$ has been extended in order to handle recursive support interactions together with recursive attacks [44; 45]. Another variant of $AFRA$, called RAF has been proposed in [29] and extended in turn to take into account for support interactions. Similar works have proposed to handle recursive frameworks through the definition of a Meta-Argumentation Framework. The idea goes back to [17; 19; 56; 57].

A common point of all these approaches for taking into account higher-order attacks, and then higher-order supports, is the fact that they somehow change the role that attacks play in Dung's frameworks. Moreover, in addition to accounting for the acceptance status of arguments in the framework, some of these works go further by also extending the traditional notion of extension from Dung's AFs to also account for the acceptance of sets of interactions (either attacks or supports). In this chapter many different approaches are presented, trying to highlight their key points and establishing comparisons between them. In order to do this presentation, some choices have been made.

The first one is to present each approach using the main definitions and results given in its seminal paper. Sometimes it occurs that, for a same line of research, many other variants are produced (for completing something that was missing, for adapting it to a specific context, for correcting some undesired behaviours, etc). In such cases, the presentation of each variant is not detailed. Indeed, in this survey, we want to give the most synthetic point of view of each approach (to the extent we can) in order to compare them.

The second choice is the presentation frame we follow for each approach with higher-order interactions: first the definition of the framework (the basic components), second its semantics (extension-based then labelling-based, when provided) and finally some other elements that may exist, such as translation mechanisms; moreover, some comparison points with the approaches presented previously in the chapter will also be given. Of course, this presentation frame will be adjusted since the degree of attention received from the scientific community varies depending on the approach (for instance, labelling-based semantics do not exist for some approaches whereas for others there is no translation mechanism, and so on).

The third choice is the organization of the chapter itself.

- In Section 2, we first recall the cornerstone of the abstract argumentation, the *first-order abstract argumentation framework* defined by Dung.

- Section 3 describes the main contributions on abstract argumentation frameworks using *higher-order attacks.* In this section the reader can find the *EAF* proposed by Modgil, the *HLAF* discussed by Gabbay, the *AFRA* defined by Baroni *et al*, the inductive approach introduced by Hanh *et al* and the *RAF* presented by Cayrol *et al.* Section 3 ends with a succinct subsection summarizing all the comparison points between the five approaches that are presented throughout this section.

- Section 4 contains a succinct presentation of *bipolar first-order argumentation frameworks* with three variants: the general support (Cayrol *et al*), the necessary support (Nouioua *et al*) and the evidential support (Oren *et al*). A short subsection is included at the end of this section, linking the first-order argumentation frameworks presented there with other approaches: structured argumentation systems that take supports into account; also, works using support relations for performing legal reasoning, for mining arguments and relations from debates, and for identifying arguments and their relations in an empirical study.

- Then, the works taking into account *higher-order attacks and supports* are presented in Section 5. In this section, another kind of support (the deductive one) is discussed since it is directly introduced as a component in an higher-order framework by Boella *et al*; then, for the necessary support, two approaches are presented: the *ASAF* and the *RAFN* respectively defined by Cohen *et al* and Cayrol *et al*; and finally, we study the *REBAF* defined by Cayrol *et al* for the evidential support. As in Section 3, Section 5 ends with a succinct subsection that summarizes all the comparison points between the four approaches that can be found throughout this section.

- Section 6 is dedicated to some computational issues and applications.

- A comparative synthesis is presented in Section 7 covering all the presented approaches.

- Finally, we conclude in Section 8.

Figure 2 shows how the reader can explore the presentation of each type of approach among the sections of the chapter.

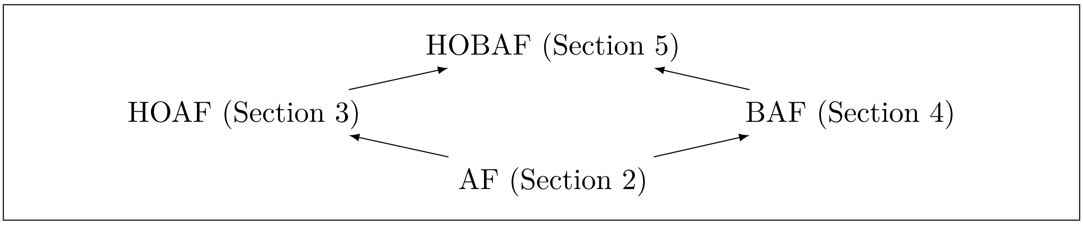


Figure 2: Roadmap for this chapter (HO: Higher-Order, AF: argumentation framework, BAF: bipolar AF)

2 Dung's approach: a first-order abstract argumentation framework

In this section, we will introduce the abstract argumentation framework proposed in [51], the corner stone of most of the developments in abstract argumentation for the past 25 years.

As defined in [51], an (abstract) argumentation framework is characterized by a set of abstract entities called arguments and a conflict relation among them.

Definition 1 (Def. 2 in [51])**.** *An argumentation framework (AF) is a pair $\langle Ar, att \rangle$, where Ar is a set of arguments and $att \subseteq Ar \times Ar$.*

For any two arguments $a, b \in Ar$, the meaning of $(a, b) \in att$ is that a attacks b or, equivalently, that a is an attacker of b. Also, an AF can be graphically represented through a directed graph, where the nodes depict the arguments and the edges correspond to the attack relation.

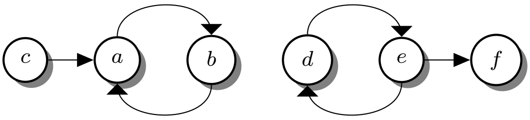
Dung then moves forward to formally characterizing the outcome of an AF, expressed in terms of sets of accepted arguments or *extensions*. As different outcomes may be obtained under different criteria, referred to as *semantics*, Dung started by proposing some basic semantic notions.

Definition 2 (Defs. 5 and 6 in [51])**.** *Let $\langle Ar, att \rangle$ be an AF and $S \subseteq Ar$:*

- *S is conflict-free iff there are no arguments $a, b \in S$ such that $(a, b) \in att$.*
- *An argument $a \in Ar$ is acceptable w.r.t. S iff for each argument $b \in Ar$ such that $(b, a) \in att$, there exists an argument $c \in S$ such that $(c, b) \in att$.*
- *S is admissible iff it is conflict-free and each argument in S is acceptable w.r.t. S.*

To illustrate these notions, let us consider the following example.

Example 2. *The AF* $\langle\{a,b,c,d,e,f\},\{(a,b),(b,a),(c,a),(e,d),(d,e),(e,f)\}\rangle$ *can be represented by the graph illustrated below:*



Some examples of conflict-free sets of this AF are $\varnothing$, $\{a\}$, $\{b\}$, $\{c\}$, $\{d\}$, $\{e\}$, $\{f\}$, $\{b,c\}$, $\{d,f\}$ *and* $\{a,e\}$.

Regarding the notion of acceptability, for instance, argument c *is acceptable w.r.t. any set of arguments since it is unattacked. Also, arguments* d *and* f *are acceptable w.r.t. the set* $\{d\}$. *Then, since the set* $\{d,f\}$ *is conflict-free, it is also an admissible set of* AF. *In contrast, the set* $\{a\}$ *is not admissible because, even though it is conflict-free and it defends* a *against the attack from* b, *it does not defend* a *against the attack from* c.

As part of the definition of acceptability semantics for AF, [51] introduced the characteristic function, $\mathrm{F}_{AF} : 2^{Ar} \mapsto 2^{Ar}$, where $\mathrm{F}_{AF}(S) = \{a \mid a \text{ is acceptable w.r.t. } S\}$. Then, the complete, preferred, grounded and stable semantics for AFs are defined using the notion of extension as follows.

Definition 3 (Defs. 7, 13, 20 and 23 in [51])**.** *Given* $AF = \langle Ar, att\rangle$ *and* $S \subseteq Ar$:

- S *is a preferred extension of* AF *iff it is a maximal (w.r.t. set inclusion) admissible set of* AF.
- S *is a stable extension of* AF *iff it is conflict-free and* $\forall a \in Ar\backslash S$, $\exists b \in S$ *such that* $(b,a) \in att$.
- S *is the grounded extension of* AF *iff it is the least fixed point of* F_{AF}.
- S *is a complete extension of* AF *iff it is an admissible set and* $\forall a \in Ar$ *such that* a *is acceptable w.r.t.* S, $a \in S$.

A series of results surrounding the basic semantic notions and the characterization of the different semantics are formalized in [51], some of which establish a relationship between sets of extensions obtained under different semantics. On the one hand, Dung's *Fundamental Lemma* shows that given any two arguments a and a' which are acceptable w.r.t. an admissible set S, the set $S' = S \cup \{a\}$ is also admissible, and a' is acceptable w.r.t. S'. Then, it is also shown that the characteristic function of an AF is monotonic (w.r.t. set inclusion). Then, amongst the results over the different semantics, we can highlight the following:

- Each preferred extension is also a complete extension, but not vice-versa.
- Every stable extension is also a preferred extension but not vice-versa.
- A complete extension is a fixed point of the characteristic function of AF.
- The grounded extension is the least (w.r.t. set inclusion) complete extension.
- Every argumentation framework possesses a grounded extension and at least one preferred extension. This is not the case for stable extensions.

The different semantics proposed in [51], as well as some of their relationships, are illustrated below.

Example 2 (cont'd) *The grounded extension of AF is $\{b, c\}$, whereas the preferred (also, stable) extensions are $\{b, c, d, f\}$ and $\{b, c, e\}$.*

Finally, it is worth mentioning that many additional semantics for AFs have been proposed in the literature, as well as alternative characterizations in terms of *labellings* (see [9] for an overview). However, in this chapter we will focus on the complete, preferred, stable and grounded semantics (referred to as the *Dung semantics* or the *classical semantics*) since they are the ones covered by the approaches to higher-order interactions considered in this chapter.

3 The premises for higher-order interactions: higher- order argumentation frameworks

To our best knowledge, the first work in which the idea of higher-order interactions appears has been presented is [13]. In that article, generalized argumentation networks are presented considering the following points: nodes are arguments, arrows are interactions with two possible cases (attacks or supports), each element of these networks (nodes and arrows) are valued, and the interactions are used in order to propagate these values. Note that the notion of support used in that work is not clearly defined and seems not to correspond to any of the types of support presented in Section 2. In this context, higher-order interactions (from an argument to an interaction)[1] are introduced only in order to influence the value of the target interaction; such a propagation process is described through some examples. Nevertheless, no semantics (extension-based or labelling-based) is formally defined.

[1] Note that the possibility of having an attack as a source of an attack is also evoked in [13] but not really used.

Following this seminal work, many different approaches have been developed with, at least at the beginning, a focus on higher-order attacks (so without taking into account the support relation) and a strong link to the notion of "valuation" (in the most general sense, so values or preferences). This is for instance the case of the *Extended Argumentation Framework* (*EAF*) that is proposed in [67; 69].

3.1 The Extended Argumentation Framework (*EAF*)

The aim of this approach is to explicitly represent the impact of the preferences between arguments in the argumentation framework by the introduction of attacks that target other attacks. These "second-order attacks" are then used in the definition of the defeat relation (the attack relation refined by preferences between arguments), that is in turn used in the computation of semantics. The formal definition of an *EAF* issued from [69] is the following:

Definition 4 (Def. 4 in [69]). *An* Extended Argumentation Framework *(EAF) is a tuple* $\langle Ar, att, att2\rangle$ *such that:*

1. *Ar is a set of arguments,*
2. *$att \subseteq Ar \times Ar$ is a set of "simple attacks" (i.e. binary attacks between arguments),*
3. *$att2 \subseteq Ar \times att$ is a set of attacks targeting simple attacks,*
4. *if $(a,(b,c))$ and $(a',(c,b)) \in att2$ then (a,a') and $(a',a) \in att$.*

As in Dung's framework, an *EAF* can be represented using a directed graph in which nodes correspond to arguments, and edges to attacks (solid arrows for simple attacks – elements of *att* – and double-pointed arrows for attacks to attacks – elements of *att2* –).

This definition can be illustrated using an example also issued from [69]:

Example 3 (**Introduction example in [69]**). *Consider two people exchanging arguments about the weather forecast:*

Argument a**:** *Today will be dry in London since the BBC forecast sunshine.*

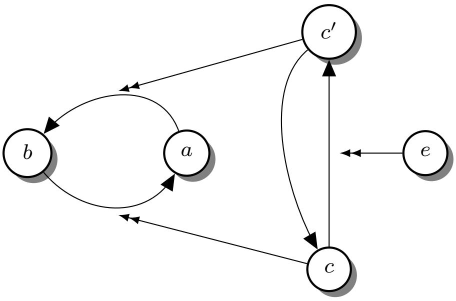
Argument b**:** *Today will be wet in London since CNN forecast rain.*

Argument c**:** *But the BBC are more trustworthy than CNN.*

Argument c'**:** *However, statistically CNN are more accurate forecasters than the BBC.*

Argument e**:** *And basing a comparison on statistics is more rigorous and rational than basing a comparison on your instincts about their relative trustworthiness.*

Here, c and c' do not attack a nor b. They attack the attacks between a and b: c by saying that a is preferred to b, and c' by saying that b is preferred to a. Moreover, the same behaviour occurs with e, which attacks the attack from c to c' (by saying that c' is preferred to c). This example can be represented by the following EAF:



Then, using the *att2* relation, the notion of conflict-freeness can be refined and the notion of defeat, related to a given set of arguments, can be introduced:

Definition 5 (Defs. 5 and 6 in [69])**.** *Let $\langle Ar, att, att2\rangle$ be an EAF and $S \subseteq Ar$.*

S is conflict-free *iff $\forall a, b \in S$, if $(a, b) \in att$, then $(b, a) \notin att$ and $\exists c \in S$ such that $(c, (a, b)) \in att2$.*

The argument a defeats the argument b w.r.t. *S (denoted by $a \rightarrow^S b$) iff $(a, b) \in att$ and there exists no argument $c \in S$ such that $(c, (a, b)) \in att2$.*

Note that each unattacked attack originates a defeat w.r.t. any set. Another interesting point is the fact that an argument and its attacker can belong to the same conflict-free set if the attack between them is not a symmetrical one and is attacked by an element of the set. Moreover, even if the notion of defeat is not directly used in the definition of conflict-free sets, both notions are related: a conflict-free set cannot contain elements involved in a defeat.

In order to refine the concept of acceptability, an additional notion is introduced in [69]: the reinstatement set (informally, the set of defeats that is able to reinstate a given defeat using a given set of arguments).

Definition 6 (Def. 7 in [69])**.** *Let $\langle Ar, att, att2\rangle$ be an EAF and $S \subseteq Ar$.*

Consider the set of defeats $R^S = \{a_1 \rightarrow^S b_1, \ldots, a_n \rightarrow^S b_n\}$. R^S is a reinstatement set for the defeat $c \rightarrow^S d$ iff :

1. $c \rightarrow^S d \in R^S$,

2. *for $i = 1 \ldots n$, $a_i \in S$,*

3. *$\forall a_i \rightarrow^S b_i \in R^S$, $\forall b'$ such that $(b', (a_i, b_i)) \in att2$, $\exists a' \rightarrow^S b' \in R^S$.*

Then, semantics for EAF are defined in much the same way as for Dung's framework but using the defeat relation in place of the attack relation and also the reinstatement set for defining the notion of acceptability.

Definition 7 (Defs. 8 and 9 in [69])**.** *Let $\langle Ar, att, att2\rangle$ be an EAF and $S \subseteq Ar$.*

An argument $a \in Ar$ is acceptable w.r.t. S iff $\forall b$ such that $b \rightarrow^S a$, $\exists c \in S$ such that $c \rightarrow^S b$ and there is a reinstatement set for $c \rightarrow^S b$.

Let S be a conflict-free set of arguments, then:

- *S is an admissible extension iff every argument in S is acceptable w.r.t. S.*
- *S is a preferred extension iff S is a $\subseteq$-maximal admissible extension.*
- *S is a complete extension iff each argument which is acceptable w.r.t. S is in S.*
- *S is a stable extension iff $\forall b \notin S$, $\exists a \in S$ such that $a \rightarrow^S b$.*

Example 3 (cont'd) *With this example, we can illustrate the previous definitions. Considering the notion of defeat, there are 4 possible defeats (one for each element of att):*

- *$b \rightarrow^S a$ with any S that does not contain c (i.e. if S contains c, then the attack from b to a is not a defeat w.r.t. that set).*
- *$a \rightarrow^S b$ with any S that does not contain c'.*
- *$c \rightarrow^S c'$ with any S that does not contain e.*
- *$c' \rightarrow^S c$ with any S, since the attack (c', c) is never attacked.*

Note that e is never defeated, since it is never attacked. Note also that the two-length cycle between c and c' has been "broken" by the attack issued from e: the attack from c' to c is always a defeat, whereas the attack from c to c' is a defeat only w.r.t. sets that do not contain e. The same thing occurs for the two-length cycle between a and b, since these attacks become defeats w.r.t. sets with different constraints.

Concerning the notion of conflict-freeness, some examples follow. The set $\{a, b\}$ (resp. $\{c, c'\}$) is not conflict-free since there is a symmetrical attack between these arguments; whereas the set $\{e, c, a\}$ is conflict-free since there is no attack between these arguments.

The notion of acceptability w.r.t. a given set can be illustrated using the set $S = \{e, c', b\}$: e (resp. c′) is acceptable w.r.t. S since it is unattacked (resp. since there is no defeat targeting c′ w.r.t. S which contains e); c (resp. a) is not acceptable w.r.t. S since c′ cannot be attacked by a defeat w.r.t. S (resp. since b cannot be attacked by a defeat w.r.t. S).

And finally, following Definition 7, one can conclude that the set $S = \{e, c', b\}$ is an admissible, preferred, complete and stable extension of the EAF.

The particular case of an *EAF* with an empty *att2* relation easily shows that *EAF*s are a conservative generalization of *AF*s. Indeed, if $att2 = \varnothing$, then the defeat and attack relations coincide and the reinstatement set can be reduced to a singleton (the attack used for defending the acceptability of the argument against a given attack).

Of course, when the *att2* relation is not empty, and even if *EAF*s can inherit some properties from *AF*s (for instance, the fact that preferred extensions are also complete but not vice-versa, see [81]), they also have some specifics in terms of semantics: the characteristic function of *EAF* is not, in general, monotonic and so the definition of the grounded extension differs.

Definition 8 (Defs. 10-11 in [69]). *Let $EAF = \langle Ar,\ att,\ att2\rangle$, $S \subseteq Ar$, and 2^{ArC} denote the set of all conflict-free subsets of Ar. The characteristic function* F_{EAF} *of EAF is defined as follows:*

$$\mathrm{F}_{EAF} :\ 2^{ArC} \mapsto 2^{Ar}$$
$$\mathrm{F}_{EAF}(S) = \{a \mid a \textit{ is acceptable w.r.t. } S\}$$

For any EAF $\langle Ar, att, att2\rangle$ the following sequence of subsets of Ar can be defined:

- $\mathrm{F}^0 = \varnothing$
- $\mathrm{F}^{i+1} = \mathrm{F}(\mathrm{F}^i)$

Then, the grounded extension of an *EAF* can be defined in terms of the sequence in the preceding definition as long as the *EAF* is finitary:

Definition 9 (Defs. 11-12 in [69]). *Let $EAF = \langle Ar,\ att,\ att2\rangle$. EAF is said to be finitary iff $\forall a \in Ar$ the set $\{b \mid (b, a) \in att\}$ is finite, and $\forall (a, b) \in att$ the set $\{c \mid (c, (a, b)) \in att2\}$ is finite.*

If EAF is finitary and $\mathrm{F}^0 = \varnothing$, $\mathrm{F}^{i+1} = \mathrm{F}(\mathrm{F}^i)$, then $\bigcup_{i=0}^{\infty}(\mathrm{F}^i)$ is the grounded extension of EAF.

Example 3 (cont'd) *The EAF corresponding to the weather example is clearly finitary. Then,* $\mathrm{F}^1 = \{e\}$*, since e is the only unattacked argument. Now, as shown previously,* $c' \rightarrow^S c$ *w.r.t. any set of arguments, in particular,* $\mathrm{F}^1 = \{e\}$*. In contrast, c does not defeat* c' *w.r.t.* $\mathrm{F}^1 = \{e\}$*, and* c' *has no other attackers. As a result,* c' *is acceptable w.r.t.* $\{e\}$ *and* $\mathrm{F}^2 = \{e, c'\}$*. Then, since a does not defeat b w.r.t.* $\mathrm{F}^2 = \{e, c'\}$*, and b has no other attackers, it holds that* $\mathrm{F}^3 = \{e, c', b\}$ *is the grounded extension of EAF.*

Note that some of the previous definitions were slightly improved since the publication of [69] in order to take into account some new constraints or to correct some undesired behaviours (see for instance the definition of conflict-freeness given in Def. 13 of [72], where the authors establish a link between structured argumentation systems and *EAF*s).

As shown in the literature, Dung's acceptability semantics can also be defined through labellings [9]. Briefly, a labelling assigns exactly one label to each argument: either `in`, `out`, or `undec`. The arguments labelled `in` constitute an extension E under a given semantics; `out` arguments are defeated by arguments in E, and arguments labelled `undec` are neither in the extension nor defeated by E. For an $EAF = \langle Ar, att, att2\rangle$, since attacks on attacks and the reinstatement of attacks may affect the acceptability of arguments, labels are also assigned to attacks in *att*, so that if $(x, y) \in att$ is `in` (resp. `out`), then this denotes that the attack (x, y) is successful (resp. unsuccessful). Also, analogously to the labellings for arguments, attacks can be labelled as `undec`. As a result, whereas the attacks at the argument level (*i.e.* those in the *att* relation) are labelled, second order attacks (*i.e.* those in the *att2* relation) are not. Formally:

Definition 10 (Def. 7 in [68])**.** *A labelling for an EAF* $\langle Ar, att, att2\rangle$ *is a pair of total functions* $(\mathcal{L}_{Ar}, \mathcal{L}_{att})$ *such that:*

1. $\mathcal{L}_{Ar} : Ar \mapsto \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$
2. $\mathcal{L}_{att} : att \mapsto \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$

For $\mathtt{S} \in \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$: $\mathtt{S}(\mathcal{L}_{Ar}) = \{x \in Ar \mid \mathcal{L}_{Ar}(x) = \mathtt{S}\}$; $\mathtt{S}(\mathcal{L}_{att}) = \{(x, y) \in att \mid \mathcal{L}_{att}((x, y)) = \mathtt{S}\}$

Definition 11 (Def. 8 in [68])**.** *Let* $\mathcal{L} = (\mathcal{L}_{Ar}, \mathcal{L}_{att})$ *be a labelling for an EAF* $\langle Ar, att, att2\rangle$*.* $\forall x \in Ar$:

1. $x \in \mathtt{out}(\mathcal{L}_{Ar})$ *is legally* `out` *iff* $\exists (y, x) \in att$ *such that* $\mathcal{L}_{Ar}(y) = \mathtt{in}$ *and* $\mathcal{L}_{att}((y, x)) = \mathtt{in}$.

2. $x \in \texttt{in}(\mathcal{L}_{Ar})$ *is legally* $\texttt{in}$ *iff* $\forall (y, x) \in att$*, either* $\mathcal{L}_{Ar}(y) = \texttt{out}$ *or* $\mathcal{L}_{att}((y, x)) = \texttt{out}$.

3. $x \in \texttt{undec}(\mathcal{L}_{Ar})$ *is legally* $\texttt{undec}$ *iff :*

   (a) $\nexists (y, x) \in att$ *such that* $\mathcal{L}_{Ar}(y) = \texttt{in}$ *and* $\mathcal{L}_{att}((y, x)) = \texttt{in}$*; and*

   (b) *it is not the case that:* $\forall y \in Ar$*,* $(y, x) \in att$ *implies* $\mathcal{L}_{Ar}(y) = \texttt{out}$ *or* $\mathcal{L}_{att}((y, x)) = \texttt{out}$.

$\forall (x, y) \in att :$

1. $(x, y) \in \texttt{out}(\mathcal{L}_{att})$ *is legally* $\texttt{out}$ *iff* $\exists (z, (x, y)) \in att2$ *such that* $\mathcal{L}_{Ar}(z) = \texttt{in}$.

2. $(x, y) \in \texttt{in}(\mathcal{L}_{att})$ *is legally* $\texttt{in}$ *iff* $\forall (z, (x, y)) \in att2$*,* $\mathcal{L}_{Ar}(z) = \texttt{out}$.

3. $(x, y) \in \texttt{undec}(\mathcal{L}_{att})$ *is legally* $\texttt{undec}$ *iff :*

   (a) $\nexists (z, (x, y)) \in att2$ *such that* $\mathcal{L}_{Ar}(z) = \texttt{in}$*; and*

   (b) *it is not the case that:* $\forall z \in Ar$*,* $(z, (x, y)) \in att2$ *implies* $\mathcal{L}_{Ar}(z) = \texttt{out}$.

For $\texttt{S} \in \{\texttt{in}, \texttt{out}, \texttt{undec}\}$:

- *An argument* $x \in Ar$ *is said to be illegally* $\texttt{S}$ *iff* $x \in \texttt{S}(\mathcal{L}_{Ar})$*, and it is not legally* $\texttt{S}$.

- *An attack* (y, x) *is said to be illegally* $\texttt{S}$ *iff* $(y, x) \in \texttt{S}(\mathcal{L}_{att})$*, and it is not legally* $\texttt{S}$.

Then, the admissible, preferred and stable *EAF* labellings are defined in [68] as follows:

Definition 12 (Def. 9 in [68])**.** *Let* $\mathcal{L} = (\mathcal{L}_{Ar}, \mathcal{L}_{att})$ *be a labelling for an EAF* $\langle Ar, att, att2 \rangle$.

- $\mathcal{L}$ *is admissible iff :*

  1. *no* $x \in Ar$ *is illegally* $\texttt{in}$ *or illegally* $\texttt{out}$*;*
  2. *no* $(y, x) \in att$ *is illegally* $\texttt{in}$ *or illegally* $\texttt{out}$*; and*
  3. $\forall x, y \in \texttt{in}(\mathcal{L}_{att})$*, it is not the case that* $(y, x) \in att$ *and* $(x, y) \in att$.

- $\mathcal{L}$ *is preferred iff it is admissible and there is no admissible labelling* $\mathcal{L}'$ *such that* $\texttt{in}(\mathcal{L}_{Ar}) \subset \texttt{in}(\mathcal{L}'_{Ar})$.

- $\mathcal{L}$ *is stable iff it is admissible,* $\texttt{undec}(\mathcal{L}_{Ar}) = \varnothing$ *and* $\texttt{undec}(\mathcal{L}_{att}) = \varnothing$.

Then, Modgil shows that the admissible, preferred and stable extensions of an EAF are in one-to-one correspondence with the corresponding labellings of their arguments. Specifically, for $E \in$ {admissible, preferred, stable}, E is a σ-extension of EAF iff there exists a σ-labelling $(\mathcal{L}_{Ar}, \mathcal{L}_{att})$ of EAF such that $\texttt{in}(\mathcal{L}_{Ar}) = E$.

Example 3 (cont'd) *The only preferred and stable labelling corresponding to the EAF of the weather example is $(\mathcal{L}_{Ar}, \mathcal{L}_{att})$, where $\texttt{in}(\mathcal{L}_{Ar}) = \{e, c', b\}$, $\texttt{out}(\mathcal{L}_{Ar}) = \{a, c\}$, $\texttt{in}(\mathcal{L}_{att}) = \{\beta, \eta\}$ and $\texttt{out}(\mathcal{L}_{att}) = \{\alpha, \epsilon\}$.*

The line of work on EAF was extended in different ways. For instance, in [70; 69], a specific class of EAF has been defined (the *hierarchical* EAF). This kind of framework is stratified so that attacks at some level i are only attacked by arguments that belong to the next level up. For instance, the EAF of Example 3 could be partitioned into 3 levels: level 1 corresponding to $(\{a, b\}, \{(a, b), (b, a)\}, \{(c, (b, a)), (c', (a, b))\})$, level 2 corresponding to $(\{c, c'\}, \{(c, c'), (c', c)\}, \{(e, (c, c'))\})$ and level 3 corresponding to $(\{e\}, \{\}, \{\})$. Note that there are always two kinds of attacks in these hierarchical EAFs, so second-order attacks exist.

In [70; 69], the Value-based Argumentation Frameworks introduced by Bench-Capon in [14] $(VAF)^2$ are translated into hierarchical EAFs.

Another version of hierarchical EAF which accounts for attacks originating in a set of arguments is also used in [72] in order to establish links with $ASPIC+$ [85]. These links allow the introduction of *structured EAFs* that satisfy the postulates proposed in [26].

Moreover, EAFs can be considered as meta-argumentation frameworks, *i.e.* frameworks able to argue about the argumentation process itself. Indeed, the relation *att2* given in EAF can be viewed as a "meta-element" expressing information about the argumentation process (how to take into account the attacks between two arguments when preferences exist). In [15], a study of meta-argumentation is presented with a methodology and some techniques, among them a flattening technique that transforms an EAF into a Dung argumentation framework introducing meta-arguments; in fact, this flattening gives good results in the case of a hierarchical EAF, see [82]. An application of this technique to the EAF is given in the following definition, which simplifies Def. 10 of [15] in order to avoid some irrelevant meta-arguments and attacks.

Definition 13 (Def. 10 in [15]). *Let $\langle Ar, att, att2 \rangle$ be an EAF. The flattened version of this EAF is the AF defined by:*

[2] Value-based Argumentation Frameworks have been introduced for persuasion situations. They take into account valued arguments and audiences.

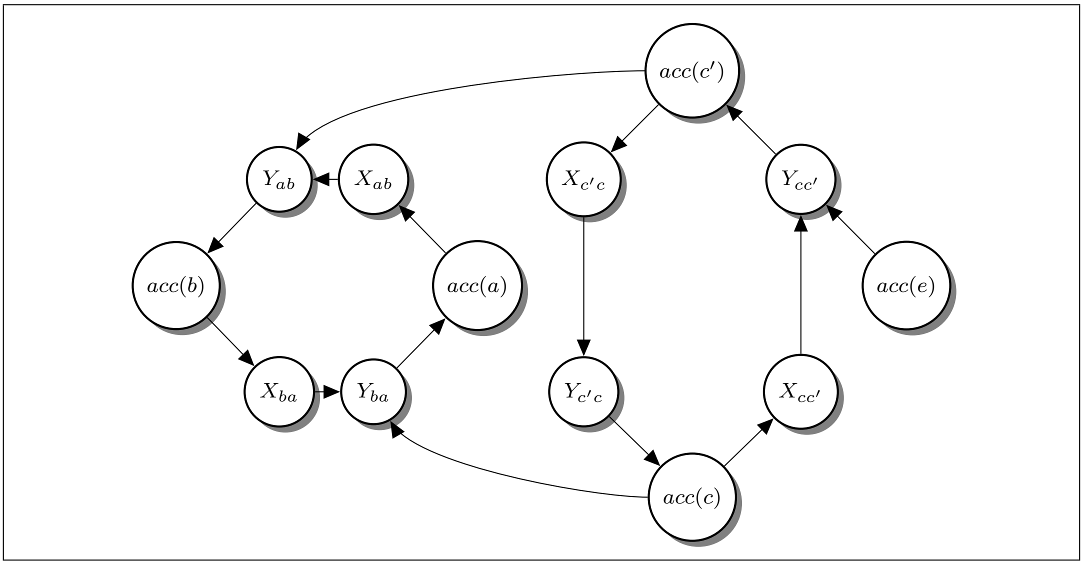


Figure 3: The flattened EAF for Example 3

- *the set of arguments* $= \{acc(a)|a \in Ar\} \cup \{X_{ab}, Y_{ab}|(a,b) \in att\}$
- *the binary attack relation* $=$
  $\{(X_{ab}, Y_{ab})|(a,b) \in att\} \cup$
  $\{(Y_{ab}, acc(b))|(a,b) \in att\} \cup$
  $\{(acc(a), X_{ab})|(a,b) \in att\} \cup$
  $\{(acc(c), Y_{ab}))|(c,(a,b)) \in att2\}$

Example 3 (cont'd) *See in Figure 3, the flattening of this EAF.*

Then with Dung semantics, the preferred (and also stable and complete) extension of this flattened EAF is the set that contains $acc(e)$, $acc(c')$, $acc(b)$ *and does not contain* $acc(a)$, $acc(c)$.

Note that several other flattening processes are proposed in literature:

- In [18], an *EAF* is proposed in order to argue about coalitions of agents. Even though the starting point used in that study is only semi-formal and so not completely abstract (the definition of coalitions is done using agents, goals, ...), the built *EAF* is abstract considering that arguments are coalitions, and attacks represent either attacks between coalitions (for instance because they have the same goal) or the impact of some preferences over these attacks. The flattening process proposed in order to take into account this *EAF* is very similar to the one defined in [15]: only names for the meta-arguments are different. Thus, the same resulting argumentation framework is produced.

- In [71], another flattening process for EAF is proposed. It is not similar to the previous ones defined in [18; 15] in the sense that the structure of the graph is not the same (more nodes and more edges). Nevertheless, it is shown in [97] that all of them correspond to the same "argumentation pattern", *i.e.* to the same behaviour of the second-order attacks.[3]

Note that, in [71, Section 5], some similarities are exhibited between EAF and other approaches, but no formal comparison is done. These approaches are the $AFRA$ (see Section 3.3) and Gabbay's approach (see Section 3.2).

3.2 The Higher-Level Argumentation Frame ($HLAF$)

Gabbay pursued his study of "higher-level networks" introduced in [13] through several papers [56; 57], using the idea of meta-argumentation. These networks are more general than EAFs since one can find attacks to attacks at any level (and not only second-order attacks); moreover, other kinds of attacks can be found in these networks. For instance:

- attacks whose source is either a set of elements (joint or conjunctive attacks), or another attack,
- attacks whose target is a set of elements (disjunctive attacks).

Gabbay's aim was to define a framework rich enough to generalize all the existing networks (including the use of a support relation, but supports are not accounted for in those papers). In the argumentation context, the following basic definition for Higher-Level Argumentation Frames ($HLAF$) considers only attacks from one argument to another argument or another attack:

Definition 14 (Def. 1.1 in [57]). *Let Ar be a set of arguments.* Level $(0, n)$ argumentation frames *are defined as follows:*

1. *A pair $(a, b) \in Ar \times Ar$ is called a level $(0, 0)$ attack.*

[3] In [97], an argumentation pattern is defined as a multi-labelling of a set of arguments associated to a propositional formula reflecting constraints about this labelling. For instance, the EAF defined by three arguments a, b and c with an attack from a to c and a second-order attack from b to (a, c) is characterized by the constraint:
$[(\mathcal{L}ab(c) = \texttt{in}) \to (\mathcal{L}ab(a) = \texttt{out} \vee \mathcal{L}ab(b) = \texttt{in})] \wedge$
$[(\mathcal{L}ab(a) = \texttt{in} \wedge \mathcal{L}ab(b) = \texttt{out}) \to (\mathcal{L}ab(c) = \texttt{out})]$.

2. *If $c \in Ar$ and α is a level $(0, n)$ attack then (c, α) is a level $(0, n+1)$ attack.*
3. *A level $(0, n)$ argumentation frame is the pair $\langle Ar, att \rangle$, where att contains level $(0, m)$ attacks for $0 \leq m \leq n$.*

It is obvious to see that an EAF can be viewed as a particular case of $HLAF$: it is a level $(0, 1)$ argumentation frame with a specific constraint about the sources of level $(0, 1)$ attacks (see Item 4 of Definition 4):

Example 3 (cont'd) *The level $(0, 1)$ argumentation frame corresponding to the weather example is:*

- $Ar = \{a, b, c, c', e\}$
- $att = \{(a, b), (b, a), (c, c'), (c', c), (c, (b, a)), (c', (a, b)), (e, (c, c'))\}$*, the 4 first attacks being level $(0, 0)$ attacks and the 3 last ones being level $(0, 1)$ attacks.*

Then, two kinds of approaches are proposed in order to take into account these networks: labelling-based semantics and flattening processes.[4]

For the first approach, Gabbay proposed the following labelling-based semantics as in Caminada's works [25; 27]:

Definition 15 (Def. 2.2 in [57])**.** *Consider $\langle Ar, att \rangle$ a level $(0, n)$ argumentation frame. Let $\mathcal{L}ab : Ar \cup att \to \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$. $\mathcal{L}ab$ is a complete labelling if, for every $\beta \in Ar \cup att$, it holds that:*

1. *$\mathcal{L}ab(\beta) = \mathtt{in}$ if there is no a such that $(a, \beta) \in att$.*
2. *$\mathcal{L}ab(\beta) = \mathtt{out}$ if there exists a such that $(a, \beta) \in att$, $\mathcal{L}ab(a) = \mathtt{in}$ and $\mathcal{L}ab((a, \beta)) = \mathtt{in}$.*
3. *$\mathcal{L}ab(\beta) = \mathtt{in}$ if for all a such that $(a, \beta) \in att$, $\mathcal{L}ab(a) = \mathtt{out}$ or $\mathcal{L}ab((a, \beta)) = \mathtt{out}$.*
4. *$\mathcal{L}ab(\beta) = \mathtt{undec}$ if for all a such that $(a, \beta) \in att$, either ($\mathcal{L}ab(a) = \mathtt{out}$ or $\mathcal{L}ab((a, \beta)) = \mathtt{out}$), or ($\mathcal{L}ab(a) = \mathtt{in}$ and $\mathcal{L}ab((a, \beta)) = \mathtt{undec}$), or ($\mathcal{L}ab(a) = \mathtt{undec}$ and $\mathcal{L}ab((a, \beta)) = \mathtt{in}$), or ($\mathcal{L}ab(a) = \mathcal{L}ab((a, \beta)) = \mathtt{undec}$). And moreover, for some a such that $(a, \beta) \in att$, either $\mathcal{L}ab(a) = \mathtt{undec}$ or $\mathcal{L}ab((a, \beta)) = \mathtt{undec}$.*

[4] A third approach is also evoked in Gabbay's works: the translation into logical formalisms (logic programming in [56] and intuitionistic logic in [58]). Nevertheless, this approach will not be developed here due lack of space.

Example 3 (cont'd) *The set of elements that must be labelled is:* $Ar \cup att = \{a, b, c, c', e, (a, b), (b, a), (c, c'), (c', c), (c, (b, a)), (c', (a, b)), (e, (c, c'))\}$.

And the corresponding complete labelling in the sense of Definition 15 is:

- *the elements labelled* `in`*:* e, c', b, $(e, (c, c'))$, $(c, (b, a))$, $(c', (a, b))$, (c', c),
- *the elements labelled* `out`*:* c, a, (c, c'), (a, b), (b, a),
- *the elements labelled* `undec`*: nothing.*

That gives the following "complete extension":

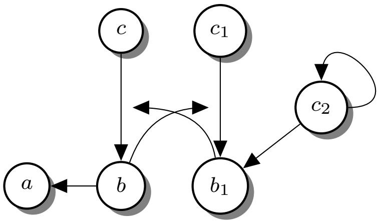
$$\{e, c', b, (e, (c, c')), (c, (b, a)), (c', (a, b)), (c', c)\}$$

It is interesting to note that attacks are also labelled and so can be viewed as belonging to the corresponding "extensions", in contrast to the semantics defined in the EAF approach (see Definition 7).

The translation approach for HLAFs has been already described in the previous section, consisting of a translation into a Dung's *AF*. Indeed, Definition 13 exactly corresponds to Gabbay's flattening process applied to the *EAF* case. However, Gabbay considers that the translation process described in Definition 13 is not enough in order to represent generalized Higher-Level Argumentation Frames and, in particular, attacks whose source is another attack. Indeed, in such a case, joint attacks must be used in order to capture the real meaning of those attacks (see the discussion in [56, Section 2]). Notwithstanding this, following the translation approach it is easy to see that Gabbay's *HLAF*s can be considered as a conservative generalization of *AF*s.

In [57] some comparisons are presented between HLAFs and *EAF* (see Section 3.1), inductive defense semantics proposed by [62] (see Section 3.4) and *AFRA* (see Section 3.3), but they remain only informal. Nevertheless, using some examples, Gabbay shows that all these approaches do not coincide. Example 4 illustrates this point by comparing *HLAF* and *EAF* (the comparison with [62]'s work can be found in Section 3.4, whereas the one with *AFRA* is given in Section 3.3).

Example 4 (See Figure 3 in [57])**.** *Consider for instance the following example.*



In this example, using the EAF semantics, the complete extension is $\{a, c, c_1\}$. With Gabbay's approach, at least two complete labellings exist, corresponding to the sets of arguments $\{a, c, c_1\}$ and $\{c, c_1\}$. Indeed, in the $HLAF$, argument b_1 can be labelled either `out` *or* `undec`*.*

Note that Gabbay's ideas can also be applied in structured argumentation. Indeed, a recent work [7] proposes a structured vision of argumentation by "blocks" (an argument being viewed as an argumentation). So, in that work, since each block is an argumentation graph and interactions exist between blocks, a kind of "recursivity" can be identified as it has been done in Gabbay's works.

3.3 Argumentation Frameworks with Recursive Attacks (*AFRA*)

In [10; 11] the authors proposed the Argumentation Framework with Recursive Attacks (*AFRA*) as a generalization of the *AF*, where attacks are allowed to target other attacks as well as arguments. The recursiveness of their approach relies on the fact that these attacks on attacks can appear at any level, thus allowing for higher-order attacks.

As argued by the authors, from a conceptual view, such a generalization supports a straightforward representation of reasoning situations which are not easily accommodated within Dung's framework. In particular, as part of their motivation and similarly to [69], the authors propose an example where higher-order attacks are partly used to encode preferences between conflicting arguments. However, as also stated by the authors in [11], further levels of recursive attacks can be considered in the area of modelling decision processes.

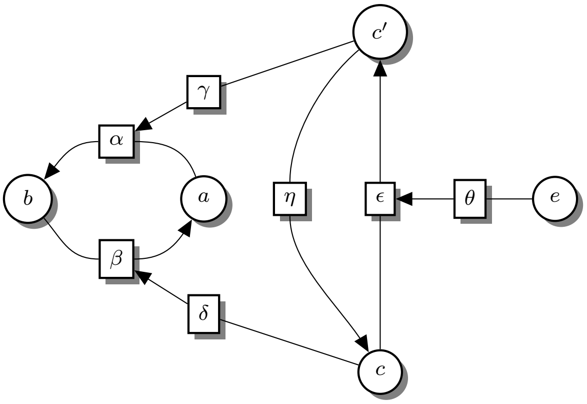
Definition 16 (Def. 3 in [11])**.**
An Argumentation Framework with Recursive Attacks *(AFRA) is a pair $\langle Ar, att\rangle$, where Ar is a set of arguments and $att \subseteq Ar \times (Ar \cup att)$ is an attack relation.*

Given an attack $\alpha = (a, X) \in att$, a is said to be the source of α, denoted as $\mathbf{s}(\alpha) = a$, and X is the target of α, denoted as $\mathbf{t}(\alpha) = X$. Moreover, the authors introduce an abbreviated notation for recursive attacks, avoiding to explicitly show all the recursive steps implied in their definition; for instance, an attack $(a, (b, c))$ can be expressed as (a, α), where $\alpha = (b, c)$. Then, as in Dung's framework, the authors introduce a graph-like notation for the *AFRA* where nodes correspond to arguments and edges represent attacks that are labelled with their associated Greek letters.

Example 3 (cont'd) *The $AFRA$ corresponding to the weather example can be defined by the sets:*

- *$Ar = \{a, b, c, c', e\}$, and*
- *$att = \{\alpha, \beta, \gamma, \delta, \epsilon, \eta, \theta\}$, where $\alpha = (a, b)$, $\beta = (b, a)$, $\gamma = (c', \alpha)$, $\delta = (c, \beta)$, $\epsilon = (c, c')$, $\eta = (c', c)$, $\theta = (e, \epsilon)$.*

The graphical representation for this AFRA is given below, where arguments are in circles and the Greek letters labelling attacks are within squares:



A key difference between the *AFRA* and the other approaches discussed in the previous subsections is that the authors of [10; 11] conceive an attack as an entity able to affect any other entity (be it an argument or an attack) rather than just a by-product of how arguments relate to each other. Consequently, all semantic notions for *AFRA* are defined following Dung's methodology, except for the fact that attacks are included as first-class elements in those definitions. As a result, similarly to Gabbay's approach where attacks are labelled (see Section 3.2), the extensions of an *AFRA* may not only include arguments, but also attacks.

As a starting point different types of defeat are defined, which regard attacks (rather than their source arguments) as the subjects able to defeat arguments or other attacks. This is also coherent with the fact that an attack can be made ineffective by attacking the attack itself. Moreover, according to the idea that an attack is strictly related to its source, a defeat over an attack also occurs in a situation where the source of the attack is itself defeated.

Definition 17 (Defs. 4, 5 and 6 in [11])**.** *Let $\langle Ar, att\rangle$ be an AFRA, $\alpha \in att$ and $X \in Ar \cup att$. α defeats X, denoted $\alpha \rightarrow^R X$, if $\mathbf{t}(\alpha) = X$ (direct defeat), or $X = \beta \in att$ and $\mathbf{t}(\alpha) = \mathbf{s}(\beta)$ (indirect defeat).*

Then, based on this notion of defeat, the notions of conflict-freeness, acceptability, admissibility and extensions under different semantics are introduced.

Definition 18 (Defs. 7, 8, 10 in [11]). *Let $\langle Ar,\ att\rangle$ be an AFRA, $S \subseteq Ar \cup att$:*

- *S is conflict-free iff $\nexists \alpha, X \in S$ such that $\alpha \to^R X$.*
- *$X \in S$ is acceptable w.r.t. S iff $\forall \alpha \in att$ such that $\alpha \to^R X$, $\exists \beta \in S$ such that $\beta \to^R \alpha$.*
- *S is admissible iff it is conflict-free and each element of S is acceptable w.r.t. S.*

Note that, whereas [10] just considered the preferred semantics, [11] extended the results to also cover the complete, grounded, stable, semi-stable and ideal semantics. Nonetheless, as mentioned before, in this chapter we will only focus on the four classical semantics since they are the ones covered by most approaches. For the purpose of defining the grounded semantics, [11] defines the characteristic function analogously to [51].

Definition 19 (Def. 9 in [11]). *The characteristic function of $AFRA = \langle Ar, att\rangle$ is defined as follows:*

$$\mathrm{F}_{AFRA}:\ 2^{Ar\cup att} \mapsto 2^{Ar\cup att}$$

$$\mathrm{F}_{AFRA}(S) = \{X \in Ar \cup att \mid X \textit{ is acceptable w.r.t. } S\}$$

Definition 20 (Defs. 11 to 14 in [11]). *Let $AFRA = \langle Ar, att\rangle$ and $S \subseteq Ar \cup att$:*

- *S is a complete extension of $AFRA$ iff S is admissible and every element of $Ar \cup att$ which is acceptable w.r.t. S belongs to S (i.e. $\mathrm{F}_{AFRA} \subseteq S$).*
- *S is the grounded extension of $AFRA$ iff it is the least fixed point of F_{AFRA}.*
- *S is a preferred extension of $AFRA$ iff it is a maximal (w.r.t. set inclusion) admissible set.*
- *S is a stable extension of $AFRA$ iff S is conflict-free and $\forall X \in Ar \cup att$, if $X \notin S$ then $\exists \alpha \in S$ such that $\alpha \to^R X$.*

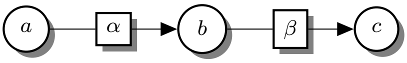
Example 3 (cont'd) *The preceding definitions can be illustrated on the weather example as follows. On the one hand, each attack in att originates a direct defeat on its target, namely, $\alpha \to^R b$, $\beta \to^R a$, $\gamma \to^R \alpha$, $\delta \to^R \beta$, $\epsilon \to^R c'$, $\eta \to^R c$ and $\theta \to^R \epsilon$. On the other hand, the indirect defeats are: $\alpha \to^R \beta$, $\beta \to^R \alpha$, $\epsilon \to^R \eta$ and $\eta \to^R \epsilon$. Then, for instance, the set $\{a, b\}$ is conflict-free even though a and b are the source and target of the attack α (also, the target and source of the attack β). As discussed before, this is because defeats can only be originated by attacks; hence, for instance, any set containing just arguments will be conflict-free in the AFRA.*

Note that, similarly to what occurs in the EAF, the defeat from θ to ϵ breaks the two-length attack cycle involving arguments c and c'. Hence, The two-length cycle involving arguments a and b is also broken. Consequently, e.g. , $\{e, \theta, c', \eta, \gamma, b, \beta\}$ is an admissible set of this $AFRA$. Moreover, this set is the only complete extension, thus being the grounded extension and the only preferred extension of the framework, which is also stable. Once again, note that this result aligns with the result obtained for the EAF, whose corresponding extension was $\{e, c', b\}$.

In contrast, we can highlight a difference between the result for the $AFRA$ and the one obtained for Gabbay's level $(0, n)$ argumentation frame: whereas the attack $(c, (b, a))$ was labelled as `in` *in the $HLAF$, δ (the corresponding attack in the $AFRA$) does not belong to the extension. This is because the attack η, which is defended against ϵ by the undefeated attack θ, directly defeats c and therefore, indirectly defeats δ. Consequently, b belongs to the $AFRA$ extension while it does not belong to the corresponding extension of the $HLAF$.*

Another simpler example allows to compare Gabbay's approach with $AFRA$.

Example 5. *Consider the following very simple framework.*



In this case, with the $AFRA$ semantics, the set $\{\alpha, c\}$ is admissible (the attack from b to c is made ineffective by α), whereas with Gabbay's approach c cannot be labelled `in` *without a being labelled* `in`.
Another difference appears for the complete semantics: in the $AFRA$, the only complete extension is the set $\{a, \alpha, c\}$ (since α defeats β), whereas in the $HLAF$ the complete labelling is the set $\{a, \alpha, c, \beta\}$ (there is no link between α and β).
In both cases, the difference is due to the notion of defeat adopted by the $AFRA$ (see Definition 17), which accounts for indirect defeats.

In addition to proposing several argumentation semantics, [11] shows that many properties satisfied by Dung's AF also hold for the $AFRA$. First, the characteristic function of the $AFRA$ is shown to be monotonic w.r.t. set inclusion (differently from the EAF's). Then, the authors prove that stable extensions of the $AFRA$ are also preferred extensions but not vice-versa. In addition, they include results showing that every preferred extension is a complete extension but not vice-versa, that the grounded extension is the least complete extension, and that every $AFRA$ possesses at least one preferred extension.

As another set of results, [11] formally shows that when an $AFRA$ coincides with an AF (when no higher-order attacks occur) the generalized notions for the $AFRA$

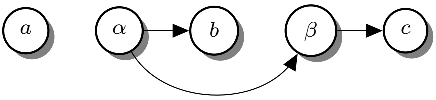
are compatible with the ones for the AF. There exists a correspondence at the level of acceptability semantics (*e.g.* grounded, preferred, stable, complete semantics) but there is no correspondence between more basic semantic notions. Consequently, this means that $AFRA$s are not a conservative generalization of AFs since, among other things, the notion of conflict-freeness does not coincide at the $AFRA$ and AF level (see for instance the fact that the set $\{a, b\}$ on Example 5 is conflict-free in an $AFRA$ but not in an AF).

Also, a flattening method is proposed to express an $AFRA$ as an AF, drawing the relevant correspondences concerning the different semantic notions and argumentation semantics. Then, following the flattening technique, the extensions of the $AFRA$ are the extensions of its associated AF.

Definition 21 (Def. 10 in [11]). *Let $AFRA = \langle Ar, att\rangle$, the corresponding argumentation framework is $AF = \langle Ar_{AF}, att_{AF}\rangle$, where:*

$$\begin{aligned} Ar_{AF} &= \quad Ar \cup att \\ att_{AF} &= \quad \{(\alpha, X) \mid \alpha \in att, X \in (Ar \cup att), \alpha \to^{R} X\} \end{aligned}$$

Example 5 (cont'd) *Applying the AFRA-AF flattening from Definition 21 we obtain the following AF:*



Here, we have that the only complete extension of the associated AF is $\{a, \alpha, c\}$ (in accordance with the result obtained by directly applying the acceptability semantics on the $AFRA$).

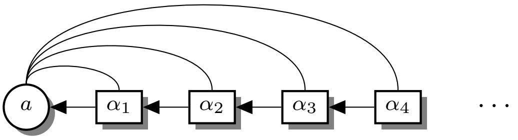
[11] formally shows that the two approaches for determining acceptability in the $AFRA$ (*i.e.* the direct computation approach and the flattening approach) are equivalent. As remarked by the authors, this kind of correspondence is very useful as it allows one to reuse or adapt, in the context of $AFRA$, the large corpus of results and implementations available for Dung's framework. In particular, as will be shown in Section 6, the flattening of an $AFRA$ into an AF is exploited for implementing a reduction-based approach to compute the $AFRA$ extensions.

Finally, [11] draws a detailed comparison between the $AFRA$ and Modgil's EAF, highlighting four points: the fact that EAF only allows for second-order attacks whereas the $AFRA$ allows for higher-order attacks at any level; the differences in the definition of conflict-freeness; the non-monotonicity of Modgil's characteristic function for the general case of EAFs versus the monotonicity of the $AFRA$ characteristic function; and, related to the previous point, the fact that the grounded extension of the EAF is not the least complete extension of an EAF in the general case (whereas this relationship does hold for the $AFRA$).

3.4 The inductive semantics for $HLAF$

In [62] the authors proposed a new inductive semantics for Gabbay's Higher-Level Argumentation Frames ($HLAF$)[5] introduced in Section 3.2. The authors argued that their semantics, based on an inductive defense relation, is sceptical and grounded towards the acceptability of attacks in a sense that an attack is "acceptable" w.r.t. a set of arguments S only if it is inductively defended by S, but could be credulous towards the acceptability of arguments. They motivated their semantics by stating that Gabbay's approach, as well as Modgil's approach, may yield counter-intuitive results in some cases, such as the one illustrated by the example below.

Example 6 (Introduction Ex. in [62])**.** *Consider a framework like the one depicted below, consisting of attacks* $\alpha_1 = (a, a)$ *and* $\alpha_{i+1} = (a, \alpha_i)$ *for* $i \geq 1$*:*



In this figure, each attack α_i *is represented by an "arrow" that goes from its source (a) to its target (a or* α_{i-1}*) "across the box"* α_i *that just gives the name of the attack.*

The authors state that they find it rather hard to imagine any practical interpretation of this framework. Then, they state that as a sceptical reasoner one would not want to draw any conclusion (as a result, not accept a). This is because an agent arguing for a has to rely on an infinite line of defense $\alpha_2, \alpha_4, \ldots$*. Then, they argue that the semantics for HLAF introduced in Section 3.2, as well as the corresponding AFRA semantics from Section 3.3 will yield a unique preferred extension* $\{a, \alpha_2, \alpha_4, \ldots\}$*, and they find this result counter-intuitive.*

In order to avoid undesired results like the one mentioned above, [62] proposes the inductive semantics of $HLAF$s which, in a situation like the one corresponding to Example 6, will yield the empty set as the only extension. For simplicity, the authors define their semantics for *bounded* $HLAF$s, but mention that their results could be easily generalized for the case of unbounded $HLAF$. Briefly, a $HLAF$

[5] In [62] the authors referred to Gabbay's formalism as the Extended Argumentation Framework (EAF); however, in order not to confuse it with Modgil's EAF (see Section 3.1) here we will keep Gabbay's naming for $HLAF$.

$\langle Ar, att\rangle$ is said to be bounded if each argument or attack in the framework has a finite number of attacks against it.

They start by defining the notion of *inductive defense*, which captures a sceptical attitude of rational agents towards the acceptance of attacks.

Definition 22 (Def. 3.1 in [62])**.** *Given $HLAF = \langle Ar,\ att\rangle$, $S \subseteq Ar$ and $\beta \in att$:*

- *S inductively defends (for short, i-defends) β within 0-steps iff there is no argument $c \in Ar$ such that $(c, \beta) \in att$.*

- *S i-defends β within $(k+1)$-steps iff either: S i-defends β within k-steps; or for each $c \in Ar$, if $(c, \beta) \in att$, then there exists $d \in S$ such that:*

    - *$(d, c) \in att$ and S i-defends (d, c) within k-steps, or*
    - *$(d, (c, \beta)) \in att$ and S i-defends $(d, (c, \beta))$ within k-steps.*

Example 3 (cont'd) *Given the HLAF corresponding to the weather example, it holds that γ, δ, η and θ are i-defended by any set of arguments within 0-steps (thus, within k-steps for $k \geq 1$) since they are not attacked by any argument in the framework.*

Then, it holds that β is i-defended by the set $S = \{c'\}$ within 1-steps because for the only argument c such that $(c, \beta) = \delta \in att$, there exists $c' \in S$ such that $(c', c) = \eta \in att$ and S i-defends η within 0 steps.

In contrast, α and ϵ are not i-defended by any set; moreover, they are respectively attacked by γ and θ, which are i-defended by any set within 0-steps.

Then, accounting for this notion of inductive defense, they characterize the new acceptability semantics of $HLAF$ as follows.

Definition 23 (Defs. 3.2 to 3.5 in [62])**.** *Given $HLAF = \langle Ar, att\rangle$ and $S \subseteq Ar$:*

- *S is i-conflict-free iff $\nexists a, b \in S$ such that $(a, b) \in att$ and S i-defends (a, b) (within any number of steps).*

- *An argument $a \in Ar$ is i-acceptable w.r.t. S iff for each $b \in Ar$ such that $(b, a) \in att$, there exists $c \in S$ such that:*

    - *$(c, b) \in att$ and S i-defends (c, b); or*
    - *$(c, (b, a)) \in att$ and S i-defends $(c, (b, a))$.*

- *S is i-admissible iff it is i-conflict-free and every argument in S is i-acceptable w.r.t. S.*

- *The characteristic function based on i-defense is defined as follows:*

  $\mathrm{F}_I : 2^{Ar} \mapsto 2^{Ar}$

  $\mathrm{F}_I(S) = \{a \in Ar \mid a \text{ is i-acceptable w.r.t. } S\}$

- *S is an i-preferred extension iff it is a maximally (w.r.t. set inclusion) i-admissible set .*

- *S is an i-complete extension iff it is an i-admissible set and each argument that is i-acceptable w.r.t. S belongs to S.*

- *S is the grounded i-extension iff it is the least fixed point of* F_I.

The semantic notions of *HLAF* based on i-defense can be illustrated on the weather example.

Example 3 (cont'd) *Given that e is an unattacked argument, it holds that* $\mathrm{F}_I(\varnothing) = \{e\}$. *Then,* $\mathrm{F}_I(\{e\}) = \{e, c'\}$ *since, as shown before, there exists $c' \in Ar$ such that $(c', c) = \epsilon \in att$ but $(e, \epsilon) = \theta \in att$, where θ is i-defended by $\{e\}$. Finally,* $\mathrm{F}_I(\{e, c'\}) = \{e, c', b\}$ *since the attack $\alpha = (a, b)$ is itself attacked by $\gamma = (c', \alpha)$ and γ is i-defended by $\{e, c'\}$. Moreover, $\{e, c', b\}$ is the least fixed point of the characteristic function and a maximal i-admissible set; thus, it corresponds to both the i-grounded extension and the only i-preferred extension of HLAF. As a result, the outcome in this case coincides with that obtained for Modgil's EAF. In addition, the outcome aligns with that obtained for the AFRA (the extension obtained here is contained in the extension obtained for the AFRA), which was shown to differ from the one obtained with Gabbay's semantics for HLAF.*

Another example illustrates the differences between Gabbay's approach and inductive defense semantics.

Example 4 (cont'd) *Recall that, with Gabbay's approach, at least two complete labellings are possible corresponding to the sets $\{a, c, c_1\}$ and $\{c, c_1\}$. Let us now consider the i-defense semantics for HLAF. Given that c and c_1 are the only unattacked arguments, it holds that* $\mathrm{F}_I(\varnothing) = \{c, c_1\}$. *Then, we have* $\mathrm{F}_I(\{c, c_1\}) = \{c, c_1\}$. *Note that $a \notin$* $\mathrm{F}_I(\{c, c_1\})$ *because, even though c attacks b (the only attacker of a), a is not acceptable w.r.t. $\{c, c_1\}$ since the attack (c, b) is not i-defended by $\{c, c_1\}$ (within any number of steps); the only attacks i-defended by $\{c, c_1\}$ are (c_2, c_2), (c_2, b_1), $(b, (c_1, b_1))$, $(b_1, (c, b))$ and (b, a) (all of which are, in particular, i-defended within 0 steps). Consequently, $\{a, c, c_1\}$ cannot be an extension using the inductive semantics of [62].*

In [62] the authors formally showed that their inductive semantics preserves the key properties of well-established semantics for abstract argumentation, such as the Fundamental Lemma and the monotonicity of the characteristic function. Furthermore, it is obvious to see that, in the case of a $HLAF$ with no higher-order attacks, all attacks in the framework will be i-defended; thus, i-conflict-freeness turns into Dung's conflict freeness, and the same holds for acceptability, admissibility, etc. And so this approach is a conservative generalisation of Dung's approach.

Moreover, in [62], some links were also established with other higher-order approaches. It was shown that any extension obtained with Modgil's EAF semantics, Gabbay's $HLAF$ semantics or the $AFRA$ semantics contains a sceptical part corresponding to an extension obtained under the i-defense semantics, in addition to a credulous part resulting from the credulousness towards the acceptance of attacks. Formally, that corresponds to: let S be an extension obtained with Modgil's EAF semantics, Gabbay's $HLAF$ semantics or the $AFRA$ semantics, S contains a greatest (w.r.t. set-inclusion) i-extension T (for the homonym semantics), *i.e.* $T \subseteq S$ and $\forall U$ being an i-extension (for the same semantics), if $U \subseteq S$ then $U \subseteq T$. This result is derived differently following the other higher-order approaches that are studied in [62].

- First, the authors stated that inductive defense semantics could be viewed as a sceptical approach to the semantics of Gabbay; in that way, for instance, the g-grounded[6] extension corresponds to the union of the i-grounded extension and the set of attacks i-defended by it. Then, they state that the truly sceptical part of any g-complete extension can be characterized by an i-complete extension; here, they again highlight that the difference in the complete extensions results from the credulousness of Gabbay's approach w.r.t. the acceptance of attacks.

- Then, regarding the relationship between [62]'s semantics and the $AFRA$ semantics, the authors state that they differ in the conditions imposed over acceptable attacks, and is related to the existence of indirect defeats in the $AFRA$ (see Definition 17). In that way, an attack will be acceptable in the $AFRA$ only if both the attack and its source argument are defensible. Nevertheless, despite this difference, the authors establish a correspondence between i-complete extensions and bcgg-complete[7] extensions: a bcgg-complete extension is equal to the union of an i-complete extension and the set of attacks coming from arguments in the i-complete extension that are i-defended by it.

- And finally, as to the relationship between [62]'s semantics and Modgil's se-

[6] The grounded extension according to Gabbay's semantics. In the remainder of this section, we will refer to the extensions obtained under Gabbay's σ semantics as the g-σ extensions.

[7] Similarly to the notation for Gabbay's approach, these denote the complete extensions obtained by the $AFRA$ semantics.

mantics for the EAF, the authors remark the following: whereas Modgil's EAF could be viewed as a special case of general extended frameworks such as the $HLAF$, its semantics are based on the underlying intuition that attacks against attacks represent preferences between conflicting arguments (thus motivating the last clause of Definition 4). Therefore, this insight suggests that different intuitions and applications could lead to different classes and different semantics for extended argumentation frameworks (as also evidenced by the non-monotonicity of Modgil's characteristic function for the general case of EAF). In spite of these differences, the authors in [62] argue that their i-grounded semantics captures the most sceptical part of Modgil's grounded semantics since, as shown in [69], the characteristic function of hierarchical and preference symmetric frameworks is monotonic (where in the former case, the hierarchical restriction essentially ensures that any defense of an attack is inductive). Then, they enforce i-admissible sets by requiring m-conflict-freeness,[8] characterizing the notion of mi-admissibility.[9] As a result, they show that mi-admissibility could be viewed as a sceptical part in the credulous semantics of Modgil. In other words, that any m-preferred[10] extension contains a maximal (w.r.t. set inclusion) mi-admissible set.

To end this section it is worth mentioning that the line of work on inductive defense semantics started by [62] was recently continued in [65]. There, the authors defined a new semantics for $HLAF$ accounting for *infinite* inductive defense, since the notion of i-defense characterized in [62] is only inductively defined for finite steps. For that purpose, they defined a notion of *renovation sets* to recognize "valid attacks", similarly to the "i-defense of an attack" in [62]. Then, they formally showed the relationship between the notion of i-defense and their renovation sets: an attack α is i-defended by a set of arguments S within k-steps iff there exists a finite renovation set of α w.r.t. S, which renovates α within k-steps. In that way, they state that the semantics of [62] can also be expressed with finite renovation sets.

3.5 The Recursive Argumentation Framework (RAF)

In [29], another framework that allows representing both simple and higher-order attacks (*i.e.* attacks from an argument to either another argument or another attack)

[8] That is, conflict-freeness as defined in Definition 5 by Modgil.

[9] "mi-admissibility" means that this notion is defined mixing notions given in Modgil's work and [62].

[10] Like before, m-preferred extension is used to denote a preferred extension obtained with Modgil's semantics for EAF.

is considered.

Definition 24 (Def. 4 in [29])**.** *A* Recursive Argumentation Framework *(RAF) is a tuple* $\langle Ar, att, \mathbf{s}, \mathbf{t}\rangle$ *where* Ar *is a finite and non-empty set of arguments,* att *is a finite set disjunct from* Ar *representing attack names,* $\mathbf{s}$ *is a function from* att *to* Ar *mapping each interaction to its source, and* $\mathbf{t}$ *is a function from* att *to* $(Ar \cup att)$ *mapping each interaction to its target.*

Note that a RAF can be graphically represented in the same way as an $AFRA$ (see Section 3.3).

Acceptability semantics for argumentation frameworks with higher-order attacks have been defined in a direct way in [29]. The idea is to specify the conditions under which the arguments are considered as accepted directly on the extended framework, without translating the original framework into an AF. Moreover, due to the defeasible nature of attacks (attacks may be affected by other attacks), conditions under which the attacks are accepted must also be specified. Indeed, some attacks may not be "valid", in the sense that they cannot defeat the argument or attack they are targeting. So, acceptability conditions for arguments should be given with respect to valid attacks and, conversely, attacks should be declared valid with respect to other arguments or attacks. For instance, the fact that two arguments may be conflicting depends on the validity of the attack between them. Hence, the traditional notion of extension defined in terms of a set of arguments is replaced by a pair of a set of arguments and a set of attacks, called a "structure".

Definition 25 (Def. 5 in [29])**.** *Consider* $RAF = \langle Ar, att, \mathbf{s}, \mathbf{t}\rangle$*. A* structure *of* RAF *is a pair* (S, Γ) *with* $S \subseteq Ar$ *and* $\Gamma \subseteq att$*.*

Intuitively, given a structure $U = (S, \Gamma)$, S contains the arguments that are accepted "owing to" U and Γ contains the attacks which are valid "owing to" U (the meaning of "owing to" depending on the considered semantics).

In the following, we recall the acceptability conditions for structures, and the definitions of the semantics that are given in [29]. The key notion is the fact that a set of arguments (resp. attacks) can be "defeated" (resp. "inhibited") w.r.t. a given structure.

Definition 26 (Equations (1) – (4) in [29])**.** *Consider* $RAF = (Ar,\ att,\ \mathbf{s}, \mathbf{t})$*. Let* $U = (S, \Gamma)$ *be a structure of* RAF*,* $a \in Ar$ *and* $\alpha \in att$*.*

- a *is* defeated *w.r.t.* U *iff there is* $\beta \in \Gamma$ *with* $\mathbf{s}(\beta) \in S$ *and* $\mathbf{t}(\beta) = a$*,*
- α *is* inhibited *w.r.t.* U *iff there is* $\beta \in \Gamma$ *with* $\mathbf{s}(\beta) \in S$ *and* $\mathbf{t}(\beta) = \alpha$*.*

$Def(U)$ (resp. $Inh(U)$) denotes the set of arguments (resp. attacks) that are defeated (resp. inhibited) w.r.t. U.

Then semantics for RAF are defined as follows:

Definition 27 (Defs. 6, 7 in [29])**.** *Consider $RAF = \langle Ar, att, \mathbf{s}, \mathbf{t}\rangle$. Let $U = (S, \Gamma)$ be a structure of RAF.*

- *U is* conflict-free *iff $S \cap Def(U) = \varnothing$ and $\Gamma \cap Inh(U) = \varnothing$.*
- *Let $a \in Ar$ and $\alpha \in att$. a (resp. α) is* acceptable *w.r.t. U iff for each $\beta \in att$ with $\mathbf{t}(\beta) = a$ (resp. $\mathbf{t}(\beta) = \alpha$), either $\beta \in Inh(U)$ or $\mathbf{s}(\beta) \in Def(U)$. $Acc(U)$ denotes the set of all arguments and attacks that are acceptable w.r.t. U.*
- *U is* admissible *iff it is conflict-free and for each $x \in (S \cup \Gamma)$, x is acceptable w.r.t. U.*
- *U is* complete *iff it is conflict-free and $Acc(U) = S \cup \Gamma$.*
- *U is* stable *iff it is conflict-free and satisfies $Ar \setminus S \subseteq Def(U)$ and $att \setminus \Gamma \subseteq Inh(U)$.*
- *U is* preferred *iff it is a $\subseteq$-maximal admissible structure.*
- *U is* grounded *iff it is the $\subseteq$-minimal conflict-free structure $U = (S, \Gamma)$ satisfying $Acc(U) \subseteq S \cup \Gamma$.*[11]

Example 3 (cont'd) *The structure $(\{b, c', e\}, \{\beta, \delta, \gamma, \eta, \theta\})$ is the grounded, complete, preferred and stable structure of the RAF corresponding to the weather example. At this point we can remark an important difference with $AFRA$: whereas η defeats δ (because it defeats its source c) in $AFRA$, η does not inhibit δ w.r.t. this structure in RAF. Hence, we obtain different results for the grounded, complete, preferred and stable semantics, where δ is left out of the $AFRA$ extension, but is included in the corresponding RAF structure.*

The notion of structure has been strengthened in order to obtain a conservative generalization of Dung's frameworks for the conflict-free, admissible, complete, stable and preferred semantics. It is worth noting that in an AF, each attack is considered as valid, in the sense that it may affect its target. The next definition strengthens the notion of structure by adding a condition on attacks that will force every acceptable attack to be valid.

[11] The definition for the grounded structure was given in [33], which is an extended version of [29].

Definition 28 (Defs. 11 to 13 in [29])**.** *Consider $RAF = \langle Ar, att, \mathbf{s}, \mathbf{t} \rangle$.*

1. *A* d-structure on *RAF is a structure $U = (S, \Gamma)$ such that $(Acc(U) \cap att) \subseteq \Gamma$.*
2. *A* conflict-free (resp. admissible, complete, preferred, stable) d-structure *is a conflict-free (resp. admissible, complete, preferred, stable) structure which is also a d-structure.*

This result has also been extended to the grounded semantics in [33]. The conservative generalization proved in [29; 33] relies upon a correspondence between a Dung's framework (and its extensions) and a "non-recursive" *RAF* (and its d-structures), where a non-recursive *RAF* is a *RAF* in which no attack targets another attack.

Another one-to-one correspondence has been proved in [29]. Indeed the *RAF* and the *AFRA* approaches give similar results for complete, preferred and stable semantics but, once again, it is not the case when we consider conflict-freeness and admissibility (see Propositions 2 to 5 in [29]). Moreover this correspondence needs to apply some constraints on the semantics results (it is not a direct one). Example 5, already used for comparing Gabbay's approach with *AFRA*, can also be used for illustrating these points.

Example 5 (cont'd) *First, the set $\{\alpha, \beta\}$ cannot be conflict-free in AFRA (since α defeats β), whereas the structure $(\varnothing, \{\alpha, \beta\})$ is conflict-free in RAF.*

Moreover, recall that $\{c, \alpha\}$ is an admissible set of the AFRA, whereas the structure $(\{c\}, \{\alpha\})$ is not admissible with the RAF approach. Indeed, in AFRA, α defeats β (or b) despite the absence of its source while, in RAF, an attack whose source is not accepted cannot defeat other arguments or attacks.

Consider now the semantics level, for instance for the preferred semantics. With the RAF approach, the preferred structure is $(\{a, c\}, \{\alpha, \beta\})$ whereas with the AFRA approach, the preferred extension is $\{a, c, \alpha\}$. In that case, if we want to obtain a RAF structure from an AFRA extension, we need to add to the structure all those attacks whose only reason for being defeated, according to AFRA, is because of the attacks towards their source (here β). Conversely, the AFRA extension is obtained from the RAF structure by the removal of attacks whose source is not in the structure (here β, too).

Note that, on Example 5, *RAF* produces results similar to Gabbay's approach. This is also the case when we consider Example 4.

Example 4 (cont'd) *In this example, considering the complete labellings obtained with Gabbay's approach and the structures of the RAF approach, the same results are obtained: first a, c, c_1, and all attacks are labelled* `in` *and are in the same* *structure; second c, c_1, and all attacks except (c, b) and (c_1, b_1) are labelled* `in` *and are in the same structure.*

These two examples show some correspondences between RAF and Gabbay's higher-level argumentation frames. Nevertheless, these correspondences remain to be proven, particularly because between labellings and structures a main difference exists: the `undec` value.

3.6 Comparison between Higher-order approaches: a first and succinct summary

Throughout Section 3 we highlighted many differences and similarities in order to compare the five approaches introduced in this section (EAF–Section 3.1–, $HLAF$–Section 3.2–, $AFRA$ –Section 3.3–, i-semantics for $HLAF$–Section 3.4–and RAF–Section 3.5). These comparison points were introduced when pertinent (depending on the definitions and examples that were discussed at that point in the text). So, in order to facilitate the reading and the understanding of this chapter, we just recall here the main comparison points between all these approaches.

- First of all, these approaches have been compared with Dung's framework and generally they are a conservative generalization of the latter when no higher-order attacks are present. Nevertheless, this result does not hold for the $AFRA$ when we consider some basic notions such as conflict-freeness (see Example 5 in Section 3.3).

- An EAF can be viewed as a particular case of $HLAF$ (a level $(0, 1)$ argumentation frame with a specific constraint), but EAF and $HLAF$ do not coincide from a semantics point of view (see Example 4 in Section 3.2).

- Examples 3 and 5 in Section 3.3 illustrate the same results between $HLAF$ and $AFRA$: $HLAF$ and $AFRA$ do not coincide from a semantics point of view.

- Moreover, a detailed comparison between $AFRA$ and EAF can be found in [11] highlighting the fact that EAF and $AFRA$ do not coincide from a semantics point of view.

- Another comparison is available concerning $HLAF$, EAF and i-semantics, yielding once again the same result: $HLAF$, EAF and i-semantics do not coincide (see Examples 3 and 4 in Section 3.4 and the text given at the end of Section 3.4).

- No comparison exists between RAF and the other higher-order approaches, except for $AFRA$. In this case, Example 5 in Section 3.5 can be used for illustrating the fact that RAF and $AFRA$ do not coincide from a semantics perspective.

Note that a more complete analysis and comparison of these approaches can be found in Section 7.

4 Different variants of first-order bipolar argumentation frameworks

In this section we will present some bipolar argumentation frameworks, which are amongst the most-widely used in the literature and inspired the approaches from Section 5. Then we will end this section by briefly discussing the links between the developments on bipolar argumentation frameworks and works in structured argumentation that also account for a notion of support, as well as mentioning other works that contemplate the existence of support relations for performing legal reasoning, for mining arguments and relations from debates, and for identifying arguments and their relations in an empirical study.

Bipolar Argumentation Frameworks (BAFs) were firstly introduced in [63; 95; 5] and further developed in [34], where the authors discuss the use of bipolarity in argumentation, analyzing how it appears under different forms in each step of the argumentation process. Briefly, a BAF extends Dung's AF by considering two independent interactions between arguments, with diametrically opposed nature: an attack relation and a support relation. Over the years, different *interpretations* for the notion of support were proposed in the literature, leading to the formalization of variants of BAFs.

In this section, we will start by introducing the characterization of BAF given in [34], where a *general* notion of support is considered (*i.e.* a support relation that does not impose constraints on the arguments it relates, other than expressing a positive relationship between them). Then, we will introduce the Argumentation Framework with Necessities (AFN) originally proposed in [78], whose support relation is interpreted as *necessity*, meaning that if an argument a supports another argument b, then the acceptance of a is required to get the acceptance of b. Finally, we will present the approach of [80], where an *evidential* interpretation of support is considered to capture a particular notion: an argument cannot be accepted unless it is supported by evidence. For each of these approaches, we will only provide the basic definitions of the framework and the types of attack they consider, without entering into details about the different methods they propose for determining the accepted arguments of the framework.

As mentioned before, other interpretations for the notion of support such as *deductive* [16] or *backing* [43] have been considered in the literature (we refer the reader to [39; 46] for a full account of support in abstract argumentation). Note that the deductive approach will be introduced in Section 5.1 since it accounts for higher-order interactions.

4.1 The General Bipolar Argumentation Framework

As briefly mentioned at the beginning of this section, a Bipolar Argumentation Framework (BAF) extends Dung's AF by incorporating a support relation that is defined independently from the attack relation. Formally:

Definition 29 (Def. 1 in [34]).
A Bipolar Argumentation Framework (BAF) $\langle Ar, att, sup \rangle$ *consists of a set Ar of arguments, a binary relation att called an attack relation, and another binary relation sup called a support relation.*

Similarly to Dung's AF, a BAF can also be represented by a directed graph, with two kinds of edges: solid arrows for the attack relation and double arrows for the support relation. The notion of BAF and its graphical representation are illustrated by the following example, taken from [46] (in turn, inspired on [87; 37]):

Example 7 (Introduction example in [46]). *Consider the following arguments exchanged during the meeting of the editorial board of a newspaper:*

Argument i**:** *Information I concerning person P should be published.*

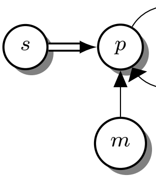
Argument p**:** *Information I is private, so P denies publication.*

Argument s**:** *I is an important information concerning P's son.*

Argument m**:** *P is the new prime minister, so everything related to P is public.*

It is clear that some conflicts appear during the discussion. That is the case of the conflict between arguments p and i, and between arguments m and p. On the other hand, there is a relation between arguments p and s, which is clearly not a conflict. Moreover, s provides a new piece of information enforcing argument p.

This discussion can be represented by a BAF as the one depicted below:



Given the coexistence of the support and attack relations in a BAF, [34] introduce the notions of *supported* and *secondary*[12] attack, which combine a sequence of supports with a direct attack.

Definition 30 (Def. 3 in [34])**.** *Given a* $BAF = \langle Ar, att, sup \rangle$ *and* $a, b \in Ar$*. A supported attack from* a *to* b *exists iff there exists a sequence of arguments* $a = a_1, \ldots, a_n = b$ $(n \geq 3)$ *such that* $(a_i, a_{i+1}) \in sup$*, with* $(1 \leq i \leq n-2)$*, and* $(a_{n-1}, a_n) \in att$*.*

A secondary attack from a *to* b *exists iff there exists a sequence of arguments* $a = a_1, \ldots, a_n = b$ $(n \geq 3)$ *such that* $(a_1, a_2) \in att$ *and* $(a_i, a_{i+1}) \in sup$*, with* $(2 \leq i \leq n-1)$*.*

By extension, the authors in [34] state that a sequence of two arguments a, b such that $(a, b) \in att$ (*i.e.* a direct attack) is also considered to be a supported attack.

Example 7 (cont'd) *Given the BAF from this example, we have the direct attacks specified by the attack relation (which are also considered to be supported attacks), and a supported attack from* s *to* i*. On the other hand, no secondary attacks exist.*

Having established the conflicts that arise from the coexistence of the attack and support relations, the authors of [34] turn to establish the conditions under which the acceptable arguments of a BAF can be identified. For that, several alternatives were proposed in [34; 35; 38], ranging from the direct characterization of the classical semantics for BAF (in particular, considering a wider range of admissible sets and preferred extensions, by imposing additional constraints related to the support relation), to the characterization of a Dung-like AF associated with BAF, called the Coalition Argumentation Framework (CAF), where arguments correspond to *coalitions* of arguments from the BAF that are linked by the support relation. Since the aim of this section is just to introduce the basic formalization of the BAF, focusing on the interpretation of support it adopts, we will not go into further details about these approaches and refer the interested reader to [39; 46]; this also applies to the approaches to be introduced in the following subsections.

[12] In [34] the authors use the terminology *indirect* attack; however, in later works they adopted the terminology *secondary*.

4.2 The Argumentation Framework with Necessities

In [78] the authors firstly introduced the Argumentation Framework with Necessities (AFN), an extension of Dung's AF that incorporates a specialized kind of support relation between arguments: the necessity relation. Briefly, the necessity relation establishes that if an argument a supports another argument b, then a is necessary to obtain b. In that way, "if b is accepted then a is also accepted" and, conversely, "if a is not accepted then b cannot be accepted" either. The authors continued their work on AFNs in [79; 20; 77]; in particular, in [79] they proposed a generalization of their framework to account for sets of supporting arguments. In this section we will introduce the basic notions surrounding the formalization of AFN as proposed in [79] and [20], since these are the ones that inspired some of the developments presented later in Section 5.

Definition 31 (Def. 4 in [20]). *An Argumentation Framework with Necessities (AFN) is defined by $\langle Ar, att, sup\rangle$, where Ar is a set of arguments, $att \subseteq Ar \times Ar$ is a binary attack relation and $sup \subseteq Ar \times Ar$ is a binary irreflexive and transitive relation, called the necessity relation.*

The authors in [20] state that the irreflexive and transitive nature of *sup* excludes any risk to have a cycle of necessities. In particular, they state that such cycles are undesirable because they correspond to a kind of fallacy (begging the question).

Given the intended meaning of the support relation in AFN, which specializes the general support relation in BAF, positive relationships like the one illustrated on Example 7 might not be well captured by the necessity relation. That is, given arguments s and p such that s supports p, it is neither the case that s is necessary for p, nor that p is necessary for s. Hence, this relation cannot be accommodated within the AFN support relation. The necessity relation of the AFN is illustrated by the following example, partly taken from [96].[13]

Example 8 (Ex. from [96]). *Consider the following (partial) argument exchange during a degree committee meeting:*

Argument a (Prof$_1$): *Student X cannot apply for a PhD on May*

Argument b (Student X): *I will graduate on March*

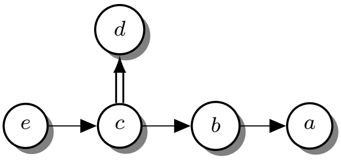
Argument c (Prof$_2$): *X is missing a grade in the logics course*

Argument d (Prof$_3$): *On the academic transcript, there is no grade in the logics course*

[13] The complete example will be introduced later in Section 5.1.

Argument e **(Student** X**):** *The professor of the logics course said I passed the exam*

This informal exchange could be represented by the AFN depicted below, where attacks are depicted by single arrows and supports are depicted by double arrows:



Here, among other relationships, we can highlight the fact that argument c is necessary for argument d.

In [79] the authors argued that, unlike a general support relation like the one introduced in Section 4.1, the necessity relation has the advantage to ensure that its interaction with the attack relation generates new attacks having exactly the same nature as the direct ones. These *extended attacks* are defined by combining a sequence of supports with a direct attack.

Definition 32 (Def. 2 in [79]). *Let* $\langle Ar, att, sup\rangle$ *be an AFN and* $a, b \in Ar$*. There is an extended attack from a to b iff there exists* $c \in Ar$ *such that either:* $(a, c) \in att$ *and* $(c, b) \in sup$*, or* $(c, b) \in att$ *and* $(c, a) \in sup$*. The direct attack* $(a, b) \in att$ *is considered to be a particular case of extended attack.*

Example 8 (cont'd) *Here, in addition to the direct attacks expressed in the attack relation att, there exists an extended attack from e to d, and an extended attack from d to b.*

It should be noted that the first kind of extended attack presented in Definition 32 coincides with the secondary attacks from the BAF (see Definition 30). This kind of extended attack is meant to enforce the acceptability constraint derived from the necessity interpretation of support; specifically, an extended attack of the first kind, where a attacks c and c supports b, is meant to enforce the constraint that if c is not accepted (in particular, in a case where a is accepted), then b should not be accepted either.

On the other hand, the second kind of extended attack is somewhat irrelevant. To illustrate this, let us consider the situation on Example 8. There, there is an extended attack of the second kind from d to b, expressing that if d is accepted, then b must not be accepted. In particular, given the constraint imposed by the necessary support relation, if d is accepted, then c must also be accepted (since c is necessary for d). Then, because of the attack from c to b, in such a case b will not be accepted. Hence, the extended attack from d to b seems a little bit useless. Nevertheless, it can be noted that the second kind of extended attack was introduced in [79] in order to highlight the duality between the necessary interpretation of support and the deductive interpretation of support originally proposed in [16][14]. For a full account of the duality between necessary support and deductive support we refer the reader to [39; 46].

Regarding the acceptability calculus in AFN, different approaches were proposed in the literature, similarly to the case of the BAF. On the one hand, the authors provided a direct characterization of the classical semantics for AFN. On the other hand, they introduced an alternative approach for obtaining the extensions of an AFN by characterizing an associated AF, obtained by considering the arguments of the original AFN and the extended attacks among them (hence, including also the direct attacks).

Finally, [79] also proposed an extension of the AFN in which the necessity relation can express the fact that a given argument requires at least one element among a set of arguments. The resulting framework is called Generalized Argumentation Framework with Necessities ($GAFN$), introduced below.

Definition 33 (Def. 8 in [79])**.** *A GAFN is defined by a tuple* $\langle Ar, att, sup\rangle$ *where* Ar *is a set of arguments,* $att \subseteq Ar \times Ar$ *is an attack relation and* $sup \subseteq ((2^{Ar} \backslash \varnothing) \times Ar)$ *is a necessity relation.*

In particular, the support relation in a $GAFN$ encodes the following constraint: given $S \subseteq Ar$ and $a \in Ar$, $(S, a) \in sup$ means that the acceptance of a requires the acceptance of at least one of the arguments in S; in other words, "if a is accepted, then there exists $b \in S$ such that b is also accepted". This generalization of the AFN was then considered in [77], where a characterization of additional semantics directly on the $GAFN$ were given following the extension-based approach, in addition to introducing labelling-based semantics for the framework. In particular, as will be shown in Section 5.2, the formalization of the AFN and the $GAFN$ inspired the characterization of different argumentation frameworks with recursive attacks and necessary supports.

4.3 The Evidential Bipolar Argumentation Framework

In argumentation theory it is usually assumed that the premises (thus, the arguments they belong to) always hold since argumentation frameworks represent a snapshot

[14] Recall that the approach to deductive support will be introduced later in Section 5 (specifically, in Section 5.1) since it also accounts for higher-order interactions.

of the arguments and relations involved on the reasoning process. However, alternative approaches like [80] consider that arguments should be backed up by evidence. Evidential reasoning involves determining which arguments are applicable based on some evidence. In that way, the approach to evidential support proposed in [80] intends to capture a particular notion: an argument cannot be accepted unless it is supported by evidence.

The *Evidential Argumentation System* was firstly introduced in [80], extending Dung's AF by incorporating a specialized support relation to capture the notion of *evidential support*; this line of work was later continued in [84]. Despite the original naming of their system, for uniformity purposes with other approaches to bipolar abstract argumentation, from hereon we will refer to this system as the *Evidential Bipolar Argumentation Framework* ($EBAF$).

The support relation in the $EBAF$ enables to distinguish between *prima-facie* and *standard* arguments. On the one hand, *prima-facie* arguments do not require support from other arguments to stand, whereas standard arguments must be linked to at least one *prima-facie* argument through a chain of supports. Given the evidential interpretation of support, an argument in the $EBAF$ will be accepted only if it is supported through a chain of arguments, each of them being itself supported. At the beginning of this chain of supporting arguments there is a special argument η that represents support from the environment (*i.e.* the existence of supporting evidence).

Definition 34 (Def. 3.1 in [84])**.** *An $EBAF$ is a tuple $\langle Ar, att, sup\rangle$, where Ar is a set of arguments, $att \subseteq (2^{Ar} \backslash \varnothing) \times Ar$ is the attack relation, and $sup \subseteq (2^{Ar} \backslash \varnothing) \times Ar$ is the support relation. A special argument $\eta \in Ar$ is distinguished, such that $\nexists (X, y) \in att$ where $\eta \in X$; and $\nexists X$ where $(X, \eta) \in att$ or $(X, \eta) \in sup$.*

Since the environment requires no support, η cannot appear as the second element of a member of *sup*; moreover, it cannot be attacked by any set of arguments. In addition, since any argument attacked by the environment will be unconditionally defeated it makes no sense to include such arguments, therefore prohibiting the environment from appearing in a set originating an attack. Also note that, differently from the previous approaches, and inspired on [74], the attack relation is not binary. Given $X \subseteq Ar$ and $a \in Ar$, $(X, a) \in att$ reads as follows: "if all the arguments in X are accepted, then a cannot be accepted". Then, for the evidential support relation, $(X, a) \in sup$ reads as: "the acceptance of a requires the acceptance of all the arguments in X"; furthermore, accepted arguments need to trace back to the special argument η.

Since the core idea of the $EBAF$ is that valid arguments (in particular, attackers) need to trace back to the environment, the authors define the notion of evidence supported attack (e-supported attack) as follows.

Definition 35 (Def. 3.2 and 3.4 in [84]). *Let $\langle Ar, att, sup\rangle$ be an EBAF, $a \in Ar$ and $S \subseteq Ar$:*

- *a has evidential support (e-support) from S iff $a = \eta$ or there is a non-empty $S' \subseteq S$ such that $(S', a) \in sup$ and $\forall b \in S'$, b has e-support from $S \backslash \{a\}$.*
- *a has minimal e-support from S if there is no $S' \subset S$ such that a has e-support from S'.*
- *S carries out an evidence-supported attack (e-attack) on a iff $(S', a) \in att$ where $S' \subseteq S$, and for all $s \in S'$, s has e-support from S.*
- *an e-supported attack by S on a is minimal iff there is no $S' \subset S$ that carries out an e-supported attack on a.*

Finally, semantics for *EBAF* have been characterized in [80] and then reformulated in [84]. In addition, [84] formally established a correspondence between *EBAF* and *GAFN* and identified correspondences between the properties of both of these systems to the properties obtained in Dung's argumentation framework. Briefly, the translation is such that unsupported arguments in the *GAFN* will correspond to arguments supported by η in *EBAF*. Then, each attack from a to b in *GAFN* will be translated into an attack from $\{a\}$ to b in *EBAF*. Then, the generalized support relation of *GAFN* is translated in a way such that all sets of supporting arguments for a given argument a in *GAFN* are combined into different sets of supporting arguments for a in *EBAF*. Formally:

Definition 36 (Transl. 1 in [84]). *Let $\langle Ar,\ att,\ sup\rangle$ be a GAFN. The corresponding EBAF $\langle Ar', att', sup'\rangle$ is created as follows:*

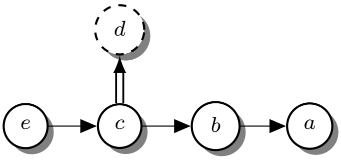
- *$Ar' = Ar \cup \{\eta\}$.*
- *For every two arguments $a, b \in Ar$ such that $(a, b) \in att$, put $(\{a\}, b)$ in att'.*
- *Let $a \in Ar$ and $Z = \{Z_1, \ldots, Z_n\}$ be a collection of all sets Z_i such that $(Z_i, a) \in sup$. If Z is empty, then put $(\{\eta\}, a)$ in sup'; otherwise, for all $Z' \in (Z_1 \times \ldots \times Z_n)$, add (Z'_S, a) to sup', where Z'_S is the set of all elements in Z'.*

Following the preceding translation we can model the *AFN* from Example 8 as an *EBAF*.

Example 8 (cont'd) *The arguments and interactions exchanged during the degree committee meeting can be represented by the EBAF $\langle Ar, att, sup\rangle$, where:*

- $Ar = \{a, b, c, d, \eta\}$
- $att = \{(\{e\}, c), (\{c\}, b), (\{b\}, a)\}$
- $sup = \{(\{\eta\}, e), (\{\eta\}, c), (\{\eta\}, b), (\{\eta\}, a), (\{c\}, d)\}$

This EBAF is depicted below where, for simplicity, the special argument η is omitted. Instead, prima-facie arguments (i.e. arguments supported by η) are represented using solid outlines whereas standard arguments are represented with dashed outlines. So, the only standard argument in this example is d. Also, since every attack and support in the EBAF originates from a singleton set, the solid arrows (resp. the double arrows) directly depart from the argument originating the attack (resp. the support).



Here, every attack in att corresponds to a minimal e-supported attack.

4.4 Links with Support in Structured Argumentation and Others

Most research on bipolar argumentation systems has been carried out at the abstract level. Notwithstanding this, there exist other works tackling the issue of dealing with the notion of support in other contexts. In this section we will briefly comment on some of them, divided into two groups.

The first group of works addresses the notion of support in three of the major structured argumentation systems: *ASPIC+* [73], Assumption-Based Argumentation (*ABA*) [48] and Defeasible Logic Programming (*DeLP*) [60]. On the one hand, [86] and [47] studied different forms of support in *ASPIC+* and analyzed whether they correspond to any of the existing interpretations of support at the abstract level, showing that *ASPIC+* sub-argument relation is a special case of necessary support, and can be considered as a special case of evidential support. On the other hand, [49] studied necessary support, deductive support, and the coalitions approach for *BAF* in the context of *ABA*. In particular, they proved that the aforementioned interpretations of support in *BAF*s correspond, under the (respective) admissible and preferred semantics (where defined), to the admissible and preferred semantics of a restricted kind of *ABA* frameworks, called *bipolar*. Finally, [42] extended

DeLP by incorporating a new kind of rules and arguments, corresponding to Toulmin's notion of backing [93]. Then, the authors showed that this extended version of *DeLP* can be used to instantiate the bipolar abstract argumentation framework that adopts the *backing* interpretation of support originally proposed in [43].

The second group of works we consider contemplates the existence of support relations in different ways: using *BAF*s to perform legal reasoning, mining arguments and support relations, and providing an empirical study showing the usefulness of support relations. [64] proposes a transformation from PROLEG [89] to a *BAF* where the support relation is originated in a set of arguments, and gives a semantics for that *BAF* in a way that guarantees that a PROLEG answer set coincides with the set of accepted arguments in the *BAF*; as stated by the authors, their aim is that the meaning of legal reasoning is preserved by their proposed semantics. The work by Cabrio and Villata [24] discusses and evaluates, on a sample of natural language arguments extracted from Debatepedia, the support and attack relations among arguments in *BAF*s adopting different interpretations of support (general, necessary, deductive) with respect to the more specific notions of textual entailment and contradiction. They investigated the distribution of those attacks in the debates, showing that all these interpretations of support (and the corresponding attacks) are verified in human debates, though with different frequency. Finally, [83] describes the results of an experiment in which participants were asked to judge dialogues in terms of agreement and structure. Among other findings, the data they collected supports the use of *BAF*s, since the notion of defence does not necessarily account for all of the positive relations between the statements viewed by the participants.

The works accounted for in this subsection serve to establish a connection between the developments on bipolar argumentation at the abstract and structured levels, as well as providing an empirical justification for using *BAF*s, their application in the legal domain, and the mining of support relations. Nevertheless, it should be noted that none of the approaches discussed above accounts for the existence of higher-order or recursive interactions (neither attack nor support). Consequently, in principle, they would not be suitable to instantiate the approaches to higher-order interactions that will be addressed in this chapter. A deeper study on how to accommodate these structured approaches to fit the existing literature about higher-order interactions in abstract argumentation is certainly of interest. However, since this chapter is meant to focus on recapping the state of the art on higher-order interactions in abstract argumentation, such study is out of scope and will be addressed on future works.

5 Different supports, so different higher-order bipolar approaches

The works presented in this section respect the typology of the support relation: deductive, necessary and finally evidential support.

5.1 Higher-order deductive supports

In [16; 96], the authors pursued their previous work presented in [15] by the introduction of supports in the same meta-argumentation framework. They only considered deductive supports: "a deductively supports b" means that "if a is accepted then b is also accepted". In fact, this is also the first work in which this notion of deductive support is formally defined and used; so, we can consider that Bipolar Argumentation Frameworks with Deductive Support ($BAFD$s) are introduced in [16]. Moreover, in order to take into account "defeasible supports" (supports that can be attacked), [16; 96] use second-order interactions with different constraints following the nature of the interaction:

- the attack relation *att* and the support relation *sup* are binary relations over the set of arguments (they are called simple attacks and supports);
- a second attack relation *att2* targets either a simple attack, or a simple support (second-order attack);
- the source of a second-order attack is either an argument or a simple attack.

This "second-order bipolar argumentation framework" can be flattened using Def. 9 in [16]. Note that, since the second-order attacks cannot be attacked, this original definition can be simplified as follows:[15]

Definition 37. *Let $\langle Ar, att, sup, att2\rangle$ be a second-order argumentation framework defined with:*

- *Ar being the set of arguments,*
- *$att : Ar \times Ar$ being the set of simple attacks,*
- *$sup : Ar \times Ar$ being the set of simple deductive supports,*
- *$att2 : (Ar \cup att) \times (att \cup sup)$ being the set of second-order attacks.*

[15] Note that Definition 37 extends Definition 13 given in Section 3.1.

The flattened version of this framework is the Dung argumentation framework defined by:

- *the set of arguments =*
  $\{acc(a)|a \in Ar\} \cup \{X_{ab}, Y_{ab}|(a, b) \in att\} \cup \{Z_{ab}|(a, b) \in sup\}$
- *the binary attack relation =*
  $\{(X_{ab}, Y_{ab})|(a, b) \in att\} \cup$
  $\{(Y_{ab}, acc(b))|(a, b) \in att\} \cup$
  $\{(acc(a), X_{ab})|(a, b) \in att\} \cup$
  $\{(acc(c), Y_{ab}))|(c, (a, b)) \in att2 \text{ and } (a, b) \in att\} \cup$
  $\{(Z_{ab}, acc(a))|(a, b) \in sup\} \cup$
  $\{(acc(b), Z_{ab})|(a, b) \in sup\} \cup$
  $\{(acc(c), Z_{ab}))|(c, (a, b)) \in att2 \text{ and } (a, b) \in sup\} \cup$
  $\{(Y_{cd}, Y_{ab}))|((c, d), (a, b)) \in att2 \text{ and } (a, b) \in att\}$[16]

As argued by the authors in [16], the coexistence of attacks and supports towards arguments in their framework leads to the existence of new attacks, which reinforce the acceptability constraints imposed by the deductive support relation. Specifically, they consider *supported attacks* (like in Definition 30 for the *BAF*) and *mediated attacks*, defined as follows.

Definition 38 (Def. 7 in [16]). *Let $\langle Ar, att, sup, att2\rangle$ be a second-order argumentation framework and $a, b \in Ar$. A mediated attack from a to b exists if there is a sequence of arguments $b = a_1, \ldots, a_n$ $(1 < n)$ such that for all $1 \leq i < n$, $(a_i, a_{i+1}) \in sup$ and $(a, a_n) \in att$.*

In [96], this approach has also been extended taking into account prioritized supports and has been applied to structured argumentation and to the Abstract Dialectical Framework (*ADF*) developed by Brewka and Woltran (see [23] for an overview).

The following example illustrates Definition 37.

Example 8 (cont'd) *Consider the following additional arguments exchanged during the degree committee meeting:*

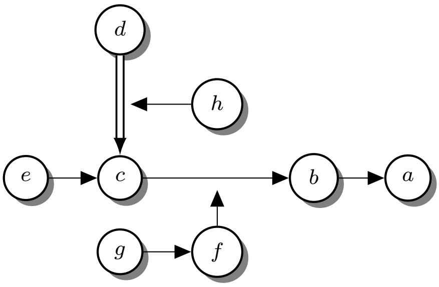
Argument f (Student X): *I was in hospital in the date of the logics exam*

Argument g (Prof$_1$): *There is no record of your stay in the hospital*

[16] In [16], the authors consider that the source of an attack that targets a support must always be an argument. Nevertheless, this constraint does not appear in Def. 9 given in [16].

Argument h **(Student** X**):** *The professor of logics was ill and could not register my exam*

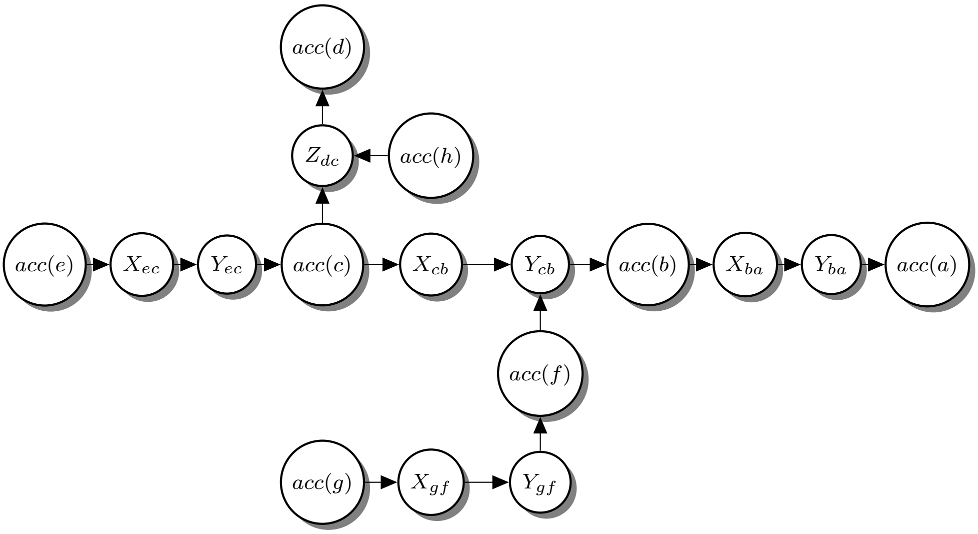
The exchange accounting for every argument and interaction can be represented by the following directed graph in which one can find, among other things, that d supports c (under the deductive interpretation of support, the direction of the arrow previously representing the necessary support from c to d is now reversed), h attacks the support (d, c)*, and* f *attacks the attack* (c, b)*:*



(attacks are represented with solid arrows and supports with double arrows)

Here, there exists a supported attack from d to b and there exists a mediated attack from e to d.

This framework can be flattened into a simple AF (only arguments and simple attacks) as follows:



The arguments b, d, e, g and h are acceptable in this AF and belong to any classical extension (grounded, preferred, stable, ...). Moreover, if we consider that the meta-arguments Y (resp. Z) represent the attacks (resp. the support), we can also conclude that the attacks (e, c), (g, f), (b, a) and of course, since they cannot be attacked, $(f, (c, b))$ and $(h, (d, c))$ are acceptable in this AF and belong to any classical extension. In contrast, the attack (c, b) and the support (d, c) are not acceptable in this AF.

Note that d is acceptable since the support (d, c) is invalidated by the attack coming from h. Otherwise, in the case the argument h is not considered, the existence of this support and the fact that c is not acceptable would imply that d would also not be acceptable.

Considering the used flattening process and the fact that the semantics in these second-order deductive bipolar frameworks are defined as in AF, it is obvious to see that these frameworks are a conservative generalization of AF, of $BAFD$[17] and of EAF. And so considering the differences between EAF and the other approaches (Gabbay's approach, $AFRA$ and RAF), it is also obvious to see that there is no one-to-one correspondence between second-order deductive bipolar frameworks and these approaches.

5.2 Higher-order necessary supports

Throughout this section, recall that "a necessary supports b" means that "if b is accepted then a is also accepted" (duality between necessary and deductive supports).

5.2.1 $ASAF$ approach

In [44] the authors firstly proposed the *Attack-Support Argumentation Framework* ($ASAF$) taking its basis from the $AFRA$ and the AFN (see Sections 3.3 and 4.2). Specifically, the $ASAF$ features a necessary support relation and an attack relation allowing for attacks and supports between arguments, as well as attacks and supports from an argument to the attack and support relations, at any level. This line of work was further pursued in [45] and [61], where the latter consolidates the previous works showing different (and equivalent) alternatives for addressing the acceptability calculus in the $ASAF$, and showing the relationship w.r.t. the frameworks it is inspired on.

As stated in [61], the intuition behind the existence of a higher-order support in the $ASAF$ (*i.e.* a support targeting an attack/support) is that the supporting

[17] Indeed, $BAFD$ corresponds to these second-order deductive bipolar frameworks without any second-order attacks.

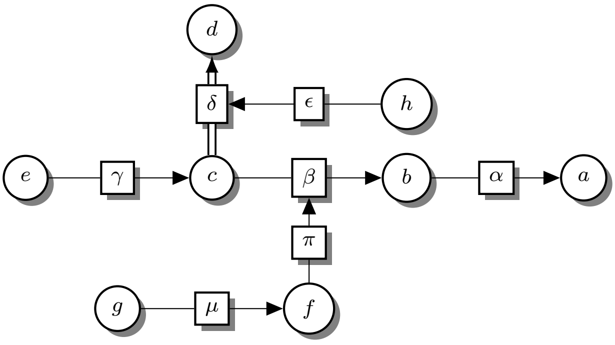
argument provides the context under which the targeted interaction holds. Hence, for instance, given a support β from an argument a to an attack or a support X, argument a should be accepted in order for the interaction X to hold. Similarly, extending the intuition behind the existence of a recursive attack relation (*e.g.* as in the EAF to model preferences), higher-order attacks in an $ASAF$ (*i.e.* attacks targeting an attack/support) capture the intuition that the attacking argument provides a context under which the targeted interaction should not hold.

Definition 39 (Def. 11 in [61]). *An* Attack-Support Argumentation Framework *(ASAF) is a tuple* $\langle Ar, att, sup\rangle$ *where* Ar *is a set of arguments,* $att \subseteq Ar \times (Ar \cup att \cup sup)$ *is an attack relation and* $sup \subseteq Ar \times (Ar \cup att \cup sup)$ *is a necessary support relation. It is assumed that* sup *is acyclic and* $att \cap sup = \varnothing$*.*

Since in $ASAF$ attacks and supports can be attacked or supported, abbreviated notations for the interactions are proposed (similarly to what is done in the $AFRA$, see Section 3.3, or in the RAF, see Section 3.5), making use of $\mathbf{s}(\cdot)$ and $\mathbf{t}(\cdot)$ for identifying the source and target of an interaction. Then, for instance, an attack from an argument a to a support from b to X (with X being an argument, an attack or a support) will be represented by a pair $\alpha = (a, \beta)$ in the attack relation att of the $ASAF$, where $\beta = (b, X)$ is a pair belonging to the support relation sup of the $ASAF$; in this case, it holds that $\mathbf{s}(\alpha) = a$, $\mathbf{t}(\alpha) = \beta$, $\mathbf{s}(\beta) = b$ and $\mathbf{t}(\beta) = X$.

Given the duality between the necessary and deductive interpretations of support discussed in Section 4.2, we can represent the discussion held during the degree committee meeting with an $ASAF$, since the only support involves arguments c and d. Nonetheless it should be noted that, for deductive supports targeting an interaction, a necessary support cannot be obtained directly by reversing the support since that would imply that the resulting necessary support originates in an interaction (and this is not allowed in the $ASAF$).

Example 8 (cont'd) *The complete exchange of arguments can be represented by the following* $ASAF$*. Similarly to before, arguments are depicted in circles, attacks are depicted using solid arrows, supports are depicted using double arrows, and attacks/supports are labelled with Greek letters in squares:*



In [61] the authors provided a characterization of the *ASAF* semantics directly on the framework. In order to do that, the authors followed the same methodology applied for the *AFRA* (see Section 3.3) which consists on first identifying the different kinds of defeat that can occur in the *ASAF* and then define some basic semantic notions to finally characterize the complete, preferred, stable and grounded semantics of the framework.

Definition 40 (Defs. 12 - 18 in [61])**.** *Let* $ASAF = \langle Ar, att, sup\rangle$*,* $\alpha \in att$*,* $X \in (Ar \cup att \cup sup)$ *and* $S \subseteq sup$*.*

- α *unconditionally defeats* X*, denoted* α udef X *iff either* $(\alpha, X) \in att$*, or* $X \in att$ *and* $(\alpha, \mathbf{s}(X)) \in att$*.*
- α *conditionally defeats* X *given the set* S*, denoted* α cdef X given S *iff there exists a sequence of arguments* $[a_1, \ldots, a_n]$ $(n \geq 2)$ *such that for every* a_i $(1 \leq i < n)$*,* $(a_i, a_{i+1}) \in sup$*, and it holds that* $\mathbf{t}(\alpha) = a_1$ *and either:* $a_n = X$*, or* $a_n = \mathbf{s}(X)$ *and* $X \in att$*; the set* S *is the union of the supports* $(a_i, a_{i+1}) \in sup$*.*

Note that the preceding definition allows arguments, attacks or supports to be defeated. In the first bullet, if $(\alpha, X) \in att$, a defeat reminiscing the direct defeat of the *AFRA* would occur; on the other hand, if $(\alpha, \mathbf{s}(X)) \in att$ with $X \in att$, a defeat akin to the indirect defeat of the *AFRA* takes place. Then, in the second bullet, if $a_n = X$ (and $\mathbf{t}(\alpha) = a_1$), a defeat corresponding to the first kind of extended attack of the *AFN* occurs; on the other hand, if $a_n = \mathbf{s}(X)$ with $X \in att$ (and again, $\mathbf{t}(\alpha) = a_1$) we have a new kind of defeat, which combines the behavior of the first kind of extended attack from the *AFN* and the indirect defeat from the *AFRA*.

Then, based on these defeats, the notions of conflict-freeness, acceptability and admissibility for ASAF are defined as follows:

Definition 41 (Defs. 19–21 in [61])**.** *Let* $ASAF = \langle Ar, att, sup\rangle$ *and* $S \subseteq (Ar \cup att \cup sup)$*.*

- *S is conflict-free iff $\nexists \alpha, X, \in S$, $\nexists S' \subseteq S$ such that either α* udef *X or α* cdef *X* given *S'.*

- *$X \in (Ar \cup att \cup sup)$ is acceptable w.r.t. S iff it holds that:*

  1. *$\forall \alpha \in att$ such that α* udef *X: $\exists \beta \in S$, $\exists S' \subseteq S$ such that β* udef *α or β* cdef *α* given *S'.*
  2. *$\forall \alpha \in att$, $\forall T \subseteq sup$ such that α* cdef *X* given *T: $\exists \beta \in S$, $\exists S' \subseteq S$, $\exists \gamma \in \{\alpha\} \cup T$ such that β* udef *γ or β* cdef *γ* given *S'.*

From the semantic notions defined in Definition 41, the complete, preferred, stable, and grounded extensions of the *ASAF* can be defined.

Definition 42 (Def. 22 in [61])**.** *Let $ASAF = \langle Ar,\ att,\ sup\rangle$ and $S \subseteq (Ar \cup att \cup sup)$.*

- *S is a complete extension of $ASAF$ iff it is admissible and $\forall X \in (Ar \cup att \cup sup)$, if X is acceptable w.r.t. S, then $X \in S$.*

- *S is a preferred extension of $ASAF$ iff it is a maximal (w.r.t. $\subseteq$) admissible set of $ASAF$.*

- *S is a stable extension of $ASAF$ iff it is conflict-free and $\forall X \in (Ar \cup att \cup sup)\backslash S$, $\exists \alpha \in S$, $\exists S' \subseteq S$ such that α* udef *X or α* cdef *X* given *S'.*

- *S is the grounded extension of $ASAF$ iff it is the smallest (w.r.t. $\subseteq$) complete extension of $ASAF$.*

Example 8 (cont'd) *The only complete, preferred and stable extension of this $ASAF$, which is also its grounded extension, is $\{e, \gamma, d, \epsilon, h, g, \mu, b, \alpha\}$. In particular note that, even though γ* cdef *d* given *$\{\delta\}$, it holds that ϵ* udef *δ. Consequently, d is acceptable w.r.t. the set $\{\epsilon\}$; moreover, note that the set $\{\gamma, d\}$ is conflict-free because it does not contain δ (the support required for the existence of the conditional defeat of γ on d). More generally, every set of arguments, attacks and supports from the $ASAF$ that does not include all the necessary elements for the existence of a defeat (either unconditional or conditional) is conflict-free; again, this characteristic is inherited from the $AFRA$.*

Recently, [1] proposed labelling-based semantics for the *ASAF*.[18] Briefly, a *labelling* for an *ASAF* $\langle Ar, att, sup\rangle$ is a total function $\mathcal{L} : (Ar \cup att \cup sup) \mapsto \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$.

[18] Note that [1] provides all the corresponding definitions in inline text; thus, we maintain inline definitions in this chapter.

Given a labelling $\mathcal{L}$, we define $\texttt{in}(\mathcal{L}) = \{X \mid \mathcal{L}(X) = \texttt{in}\}$, $\texttt{out}(\mathcal{L}) = \{X \mid \mathcal{L}(X) = \texttt{out}\}$, and $\texttt{undec}(\mathcal{L}) = \{X \mid \mathcal{L}(X) = \texttt{undec}\}$. Also, when convenient, a labelling $\mathcal{L}$ can be represented by the triple $(\texttt{in}(\mathcal{L}), \texttt{out}(\mathcal{L}), \texttt{undec}(\mathcal{L}))$.

Then, the *complete labellings* are defined in [1] as follows. $\mathcal{L}$ is a *complete labelling* of an $ASAF$ $\langle Ar, att, sup\rangle$ iff for every $X \in (Ar \cup att \cup sup)$ it holds that: (1) $\mathcal{L}(X) = \texttt{in}$ iff $\forall\alpha \in att, \forall S \subseteq sup$ such that α cdef X given S, $\exists Y \in (\{\alpha\} \cup S)$ such that $\mathcal{L}(Y) = \texttt{out}$; and (2) $\mathcal{L}(X) = \texttt{out}$ iff $\exists\alpha \in att, \exists S \subseteq sup$ such that α cdef X given S and $\forall Y \in (\{\alpha\} \cup S)$, $\mathcal{L}(Y) = \texttt{in}$.

In other words, for X to be labelled as $\texttt{in}$ by a complete labelling of an $ASAF$ the following conditions must be satisfied: for every set of elements originating a defeat on X, one of the elements in the set is labelled as $\texttt{out}$ (*i.e.* either the attack or one of the supports, if they exist). Analogously, for X to be labelled as $\texttt{out}$, it must be the case that there exists a set of elements originating a defeat on X where every element in the set (*i.e.* the attack and every support) is labelled as $\texttt{in}$. Finally, if X is neither labelled as $\texttt{in}$ nor as $\texttt{out}$, it is labelled as $\texttt{undec}$.

[1] mentions that there exists a one-to-one correspondence between complete extensions and complete labellings of an $ASAF$. Specifically, they state that each complete extension E is in one-to-one correspondence with a complete labelling $\mathcal{L} = (E, E^+, (Ar \cup att \cup sup) \setminus (E \cup E^+))$, where $E^+ = \{X \in (Ar \cup att \cup sup) \mid \exists\alpha \in E, \exists S \subseteq E$ such that α cdef X given $S\}$. That is, the complete labelling $\mathcal{L}$ *corresponding* to a complete extension E of an $ASAF$ is given by the triple $(\texttt{in}(\mathcal{L}), \texttt{out}(\mathcal{L}), \texttt{undec}(\mathcal{L}))$, where $\texttt{in}(\mathcal{L}) = E$, $\texttt{out}(\mathcal{L}) = E^+$, and $\texttt{undec}(\mathcal{L}) = (Ar \cup att \cup sup) \setminus (E \cup E^+))$.

Then, as argued by the authors in [1], the preferred, stable and grounded labellings of an $ASAF$ can be defined in terms of the complete labellings of the framework: $\mathcal{L}$ is a *preferred* (resp. *stable*, *grounded*) *labelling* of $ASAF$ iff it is a complete labelling such that $\texttt{in}(\mathcal{L})$ is a preferred (resp. *stable*, *grounded*) extension of $ASAF$.

Example 8 (cont'd) *The only complete labelling of the ASAF (also, its grounded labelling and its only preferred and stable labelling) is* $(\{e, \gamma, d, \epsilon, h, g, \mu, b, \alpha\}, \{c, \delta, f, \pi, \beta, a\}, \varnothing)$.

Finally, in [44] the authors proposed to translate an $ASAF$ into an AF in order to be able to determine the extensions of the framework. In that way, they first translated the $ASAF$ into its associated AFN and finally, translated the AFN into an AF. The translation given in [44] was later refined in [61] and is shown below.

Definition 43 (Defs. 23, 24, 9 and 10 in [61]). *Let* $ASAF = \langle Ar, att, sup\rangle$.

The AFN *associated with* $ASAF$ *is* $\langle Ar_{AFN}, att_{AFN}, sup_{AFN}\rangle$, *where:*

$$\begin{array}{rl} Ar_{AFN} = & Ar \cup att \cup sup \cup \{\beta^+, \beta^- \mid \beta \in sup\} \\ att_{AFN} = & \{(\alpha, X) \mid \alpha \in att,\ \mathbf{t}(\alpha) = X\} \cup \\ & \{(b, \beta^-), (\beta^-, Y) \mid \beta \in sup,\ \mathbf{s}(\beta) = b,\ \mathbf{t}(\beta) = Y\} \\ sup_{AFN} = & \{(a, \alpha) \mid \alpha \in att,\ \mathbf{s}(\alpha) = a\} \cup \\ & \{(\beta, \beta^+), (\beta, \beta^-), (b, \beta^+) \mid \beta \in sup,\ \mathbf{s}(\beta) = b\} \end{array}$$

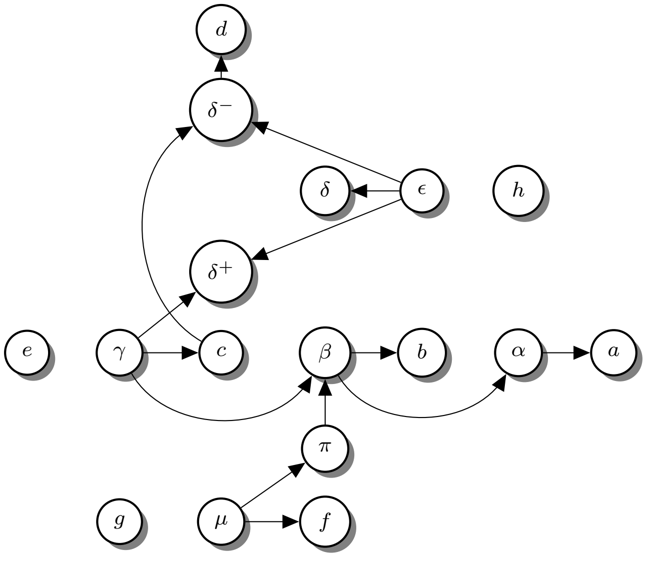
The AF associated with ASAF is $\langle Ar_{AF}, att_{AF}\rangle$, where:

$$\begin{array}{rl} Ar_{AF} = & Ar_{AFN} \\ att_{AF} = & att_{AFN} \cup \\ & \{(a, b) \mid \exists c \in Ar_{AFN} \textit{ with } (a, c) \in att_{AFN}, (c, b) \in sup_{AFN}\} \end{array}$$

Note that the second set of attacks added to att_{AF} in Definition 43 exactly corresponds to the first kind of extended attack in the AFN, as described in Def. 32. Then, as mentioned before, extensions of an $ASAF$ can be obtained from extensions of its AF as follows:

Definition 44 (Defs. 25 - 27 in [61])**.** *Let $ASAF = \langle Ar, att, sup\rangle$ and $AF = \langle Ar_{AF}, att_{AF}\rangle$ be its associated argumentation framework. If S is an extension of AF under the complete, preferred, stable or grounded semantics, then $S' = S \backslash \{\beta^+, \beta^- \mid \beta \in sup\}$ is an extension of $ASAF$ under the same semantics.*

Example 8 (cont'd) *The AF associated with the ASAF, obtained with Definition 43, is depicted below:*



Here, the only complete extension (also, the grounded extension and the only preferred and stable extension) of the associated AF is $\{e, \gamma, g, \mu, b, \alpha, h, \epsilon, d\}$; furtheremore, this is also the only complete, grounded, preferred and stable extension of the ASAF. Note that the resulting extension differs from the one obtained in the deductive case: the attack π (corresponding to the attack $(f, (c, b))$ in the deductive approach) does not belong to the ASAF extension because of the indirect defeat coming from μ; this is due to the fact that the ASAF approach takes AFRA as basis.

In [28] an alternative translation of an $ASAF$ into an AF was proposed with the aim of addressing the acceptability calculus of the framework. This alternative translation also accounts for an intermediate translation into an AFN and is driven by three features that can be identified in interactions involved in a recursion: *groundness*, *validity* and *activation*; following this translation, interactions have to be active in order to be included in the extensions of an $ASAF$. Specifically, as proposed in [28], an interaction is considered to be grounded if its source is accepted. The validity of an interaction is determined by looking at the interactions that may affect it, that is, interactions attacking and supporting it. Finally, an interaction is considered to be active if it is both grounded and valid; then, for instance, an interaction that is attacked by another interaction that is active will not be considered as valid. The translation of [28] follows:

Definition 45 (Defs. 4 and 8 in [28])**.**
Let $ASAF = \langle Ar,\ att,\ sup\rangle$. The AFN associated with ASAF is $\langle Ar_{AFN}$, att_{AFN}, $sup_{AFN}\rangle$, where:

$$\begin{array}{rl}
Ar_{AFN} = & Ar\ \cup \{\alpha \mid \alpha = (a, X) \in att\}\ \cup \{\beta \mid \beta = (b, Y) \in sup\} \\
att_{AFN} = & \{(\alpha, X) \mid \alpha \in att,\ \mathbf{t}(\alpha) = X\} \\
sup_{AFN} = & \{(a, \alpha) \mid \alpha \in att \cup sup,\ \mathbf{s}(\alpha) = a\} \cup \\
 & \{(\alpha, X) \mid \alpha \in sup,\ \mathbf{t}(\alpha) = X\}
\end{array}$$

The AF associated with ASAF is $\langle Ar_{AF}, att_{AF}\rangle$, where:

$$\begin{array}{rl}
Ar_{AF} = & Ar_{AFN} \cup\ \{N_{XY} \mid (X, Y) \in sup_{AFN}\} \\
att_{AF} = & \{(\alpha, X) \mid (\alpha, X) \in att_{AFN}\} \cup \\
 & \{(\alpha, N_{XY}) \mid (\alpha, X) \in att_{AFN},\ \alpha \in att,\ X \in sup,\ \mathbf{t}(X) = Y\} \cup \\
 & \{(X, N_{XY}), (N_{XY}, Y) \mid (X, Y) \in sup_{AFN}\} \cup \\
 & \{(N_{XY}, N_{YZ}) \mid (X, Y) \in sup_{AFN},\ X \in sup,\ Y \in sup,\ \mathbf{t}(Y) = Z\}
\end{array}$$

As proposed in [28], extensions of the $ASAF$ can be obtained from extensions of its AF obtained through Definition 45 by just filtering out the N_{XY} arguments.[19]

[19] Note that [28] provides no formal definition as to how to obtain the correspondence between extensions of the $ASAF$ and extensions of its associated AF.

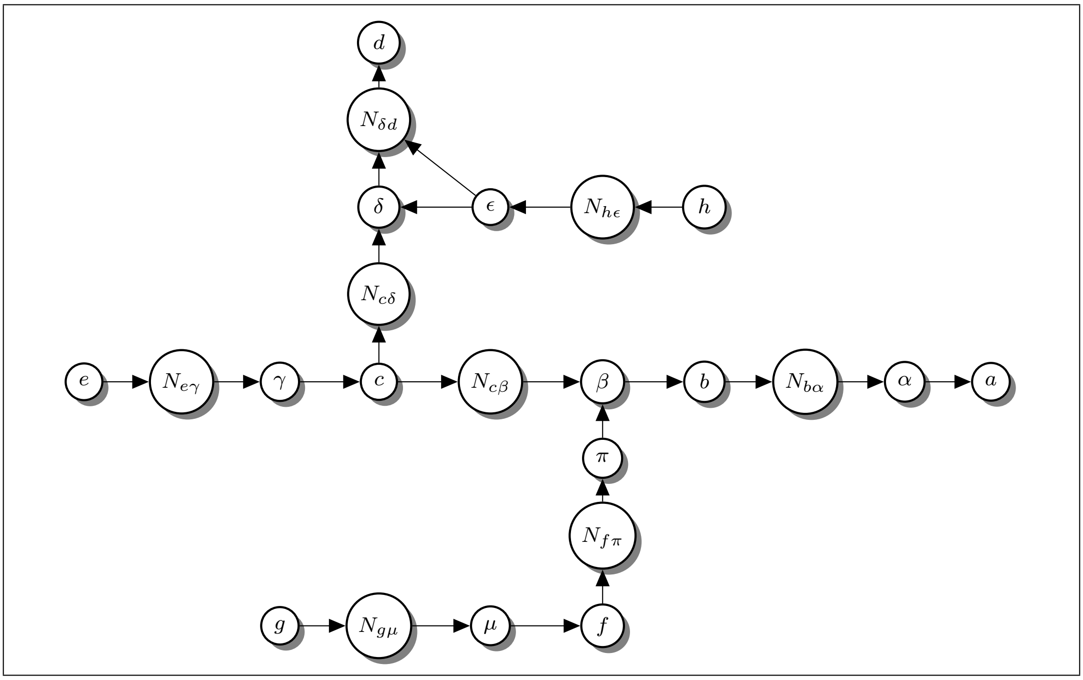
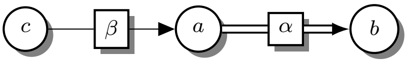


Figure 4: *AF* associated with the *ASAF* of Example 8, following Definition 45

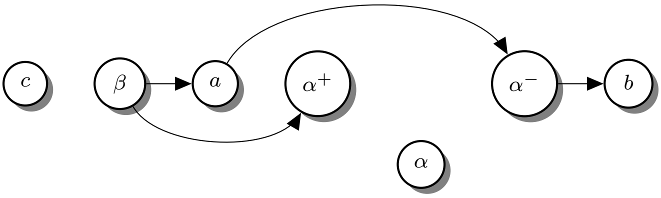
Example 8 (cont'd) *The AF associated with the ASAF, obtained with Definition 45, is depicted in Figure 4. The only complete extension (also, the grounded extension and the only preferred and stable extension) of the associated AF is* $\{e, \gamma, N_{c\beta}, N_{c\delta}, g, \mu, N_{f\pi}, b, \alpha, h, \epsilon, d\}$*. As a result, by filtering out the N-arguments, the only complete, grounded, preferred and stable extension of the ASAF is* $\{e, \gamma, g, \mu, b, \alpha, h, \epsilon, d\}$*.*

It should be noted that, although the same outcome was obtained for Example 8 when considering the translations of Definition 43 and Definition 45, this does not hold for the general case. The reason for this difference relies on the fact that, differently from [61], for a support to be accepted in [28] it must be the case that its source is also accepted. This difference is illustrated by the following example.

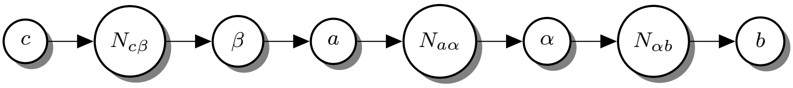
Example 9. *Consider the ASAF depicted below:*



With the translation of Definition 43 we obtain the following associated AF:



On the other hand, with the translation of Definition 45 we obtain the associated AF depicted below:



In the former case, the only complete, grounded, preferred and stable extension of the associated AF is $\{c, \beta, \alpha^-, \alpha\}$ *and thus, the only complete, grounded, preferred and stable extension of the ASAF would be* $\{c, \beta, \alpha\}$. *In the latter case, the only complete, grounded, preferred and stable extension of the associated AF is* $\{c, \beta, N_{a\alpha}, N_{\alpha b}\}$*; consequently, the only complete, grounded, preferred and stable extension of the ASAF would be* $\{c, \beta\}$.

Note that a one-to-one correspondence exists between $ASAF$ *without support* and RAF (indeed RAF and $AFRA$ approaches give similar results for semantics level, and $ASAF$ are a conservative generalization of $AFRA$). Nevertheless, it is not the case when we consider $ASAF$ with support (so $ASAF$ that are not only $AFRA$).

5.2.2 $RAFN$ approach

In [31; 32], the authors pursued their works about RAF, presented in [29] (see Section 3.5), by the definition and the study of an extension, called Recursive Argumentation Framework with Necessity ($RAFN$), that is able to take into account higher-order necessary supports. The approach presented in [31; 32] is similar to the one used in [29]: formalization of $RAFN$ and direct definition of semantics.

Note that, differently from the $ASAF$ approach, the source of a necessary support in $RAFN$ can be a set of arguments; on the other hand, like in the $ASAF$, this is not the case for an attack.

Definition 46 (Def. 17 in [31])**.** *A* Recursive Argumentation Framework with Necessity ($RAFN$) *is a tuple* $\langle Ar, att, sup, \mathbf{s}, \mathbf{t}\rangle$*, where* Ar*,* att *and* sup *are three pairwise disjunct sets respectively representing arguments, attacks and supports names,*

$\mathbf{s}$ *is a function from* $(att \cup sup)$ *to* $(2^{Ar} \setminus \varnothing)$ *mapping each interaction to its source, and* $\mathbf{t}$ *is a function from* $(att \cup sup)$ *to* $(Ar \cup att \cup sup)$ *mapping each interaction to its target. It is assumed that* $\forall \alpha \in att,\ \mathbf{s}(\alpha)$ *is a singleton.*

RAFN semantics are defined using the extension of the notion of "structure" for *RAF* (see Definition 25 in Section 3.5): A structure of the *RAFN* is a triple $U = (S, \Gamma, \Delta)$ such that $S \subseteq Ar$, $\Gamma \subseteq att$ and $\Delta \subseteq sup$. Intuitively, the set S represents the set of "acceptable" arguments w.r.t. the structure U, while Γ and Δ respectively represent the set of "valid attacks" and "valid necessary supports" w.r.t. U.

In order to define the structures corresponding to each semantics, some additional notions are introduced. Intuitively, an element x (argument, attack or support) can be defeated w.r.t. U iff there is a "valid attack" w.r.t. U that targets x and whose source is "acceptable" w.r.t. U. Concerning the notion of *supported elements* w.r.t. a structure, elements (arguments, attacks, supports) which receive no necessary support do not require any support, so they are supported w.r.t. any structure; and an element x is supported w.r.t. a given structure U if *for each* support α (which can be regarded as supported), the source of α contains *at least one* argument of U that can be regarded as supported. An element of a *RAFN* is considered as being still supportable as long as *for each* non-defeated support, *there exists at least one* argument in its source, which is non-defeated and regarded as supportable. And finally, elements that are defeated or that are unsupportable are said to be *unacceptable* (they cannot be accepted). Then an attack $\alpha \in att$ is *unactivable*[20] (such an attack cannot be "activated" in order to defeat the element that it is targeting) iff it is either unacceptable or its source is unacceptable. The following notation is used in the next definitions: let $E \subseteq (Ar \cup att \cup sup)$, $\overline{E} = (Ar \cup att \cup sup) \setminus E$.

Definition 47 (Defs. 18 to 20 in [31]). *Let* $RAFN = (Ar, att, sup, \mathbf{s}, \mathbf{t})$*. Given a structure* $U = (S, \Gamma, \Delta)$*:*

1. *For* $X \in \{Ar, att, sup\}$, $Def_X(U) = \{x \in X | \exists \alpha \in \Gamma, \mathbf{s}(\alpha) \in S$ *and* $\mathbf{t}(\alpha) = x\}$. $Def(U) = Def_{Ar}(U) \cup Def_{att}(U) \cup Def_{sup}(U)$ *denotes the set of all defeated elements w.r.t.* U*.*

2. $Supp(U) = \{x | \forall \alpha \in \Delta$ *such that* $\mathbf{t}(\alpha) = x$*, if* $\alpha \in Supp(U_{-x})$ *then* $\mathbf{s}(\alpha) \cap (S \cap Supp(U_{-x})) \neq \varnothing\}$ *with* $U_{-x} = U \setminus \{x\}$. U *is* self-supporting *iff* $(S \cup \Gamma \cup \Delta) \subseteq Supp(U)$*.*

3. $UnSupp(U) = \overline{Supp(U')}$ *denotes the set of* unsupportable *elements w.r.t.* U*.*

[20] This is the word used in [31] and a neologism. It expresses the impossibility of activating an attack.

4. *$UnAcc(U) = Def(U) \cup UnSupp(U)$ denotes the set of* unacceptable *elements w.r.t. U.*

5. *$UnAct(U) = \{\alpha \in att | \alpha \in UnAcc(U)$ or $\mathbf{s}(\alpha) \subseteq UnAcc(U)\}$ denotes the set of* unactivable *attacks w.r.t. U.*

Note that the set of elements supported by a structure are defined using a self-reference. Indeed one wants to avoid the situation in which an element x would be supported only because x is supported.

Then semantics can be defined as follows.

Definition 48 (Defs. 21 and 22 in [31]). *Let $RAFN = (Ar, att, sup, \mathbf{s}, \mathbf{t})$. Given a structure $U = (S, \Gamma, \Delta)$:*

- *$x \in Ar \cup att \cup sup$ is acceptable w.r.t. U iff (i) $x \in Supp(U)$ and (ii) for each attack $\alpha \in att$ with $\mathbf{t}(\alpha) = x$, $\alpha \in UnAct(U)$.*
  $Acc(U)$ denotes the set of all elements that are acceptable w.r.t. U.

- *U is* conflict-free *iff $S \cap Def_{Ar}(U) = \varnothing$, $\Gamma \cap Def_{att}(U) = \varnothing$ and $\Delta \cap Def_{sup}(U) = \varnothing$.*

- *U is* admissible *iff it is conflict-free and $(S \cup \Gamma \cup \Delta) \subseteq Acc(U)$.*

- *U is* complete *iff it is conflict-free and $(S \cup \Gamma \cup \Delta) = Acc(U)$.*

- *U is* preferred *iff it is a $\subseteq$-maximal complete structure,*

- *U is* stable *iff it is complete and $\overline{(S \cup \Gamma \cup \Delta)} = UnAcc(U)$.*

- *U is* grounded *iff it is a $\subseteq$-minimal complete structure.*

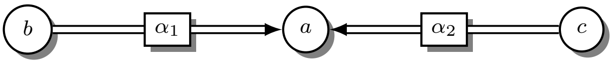
All the definitions can be illustrated using Example 8.

Example 8 (cont'd) *The graphical representation for the RAFN corresponding to this example is the same as the one given for the ASAF in Section 5.2.1.*

In this example, using the previous definitions, there is only one structure that is grounded, preferred and stable: $(\{b, d, e, g, h\}, att, \varnothing)$. Here the only interaction that is not acceptable is the support (d, c) (its attacker ϵ being acceptable). This is one difference between this approach and the approaches presented in Sections 5.1 and 5.2.1. Here, the attack β (i.e. (c, b)) is acceptable since its attacker (π) is unactivable (even if it is acceptable), the source of π being unacceptable.

The next examples illustrate further differences between the *ASAF* and the *RAFN* approach. Indeed several differences can be outlined (even if we exclude cycles of necessary supports, and assume that interactions are binary ones). First, in $ASAF$, attacks and supports are combined to obtain extended (direct or indirect) defeats and these defeats are used in the definition of conflict-freeness. In contrast, in $RAFN$, the notions of support and attack are dealt with separately.

Example 10 (Ex. 16 in [31]). *Consider the simple argumentation framework with only 2 necessary supports (so without any higher-order interaction).*

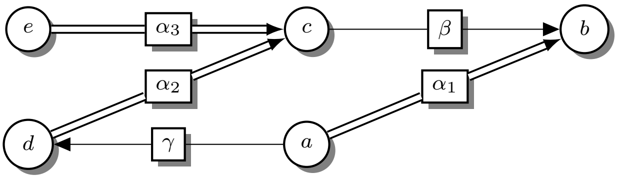


As for acceptability, following the ASAF semantics defined directly over the framework (see Section 5.2.1), an element is acceptable w.r.t. a set of elements whenever it can be defended against each defeat. So, in the particular case when there is no attack, each element of the framework would be acceptable w.r.t. any set, and the sets $\{a, \alpha_1, \alpha_2\}$, $\{a, b, \alpha_1, \alpha_2\}$, $\{a, c, \alpha_1, \alpha_2\}$ *(among others) are admissible.*

In contrast, RAFN acceptability explicitly requires a support. So, the structures $(\{a\}, \varnothing, \{\alpha_1, \alpha_2\})$, $(\{a, b\}, \varnothing, \{\alpha_1, \alpha_2\})$ *and* $(\{a, c\}, \varnothing, \{\alpha_1, \alpha_2\})$ *are not admissible with RAFN semantics.*

Another difference was already pointed out in [29], where correspondences have been provided between a RAF and an $ASAF$ without support. Indeed, in an $ASAF$, an attack is not acceptable whenever its source is not acceptable.

Example 11 (Ex. 15 in [31]). *Let RAFN be the following argumentation framework:*



With RAFN semantics, β is not attacked and not supported so β must belong to each complete structure.

With ASAF semantics, if β is acceptable w.r.t. a set S, then c must also be acceptable w.r.t. S. If S is a complete extension, S contains a, γ, α_1, α_2 and α_3. As c is defeated by γ given $\{\alpha_2\}$, it cannot be the case that c is acceptable w.r.t. S. So β cannot belong to any complete extension.

Note also that the $RAFN$ is a conservative generalization of the $GAFN$ (see Section 4.2 in [31]).

Moreover, since $RAFN$ are obviously a conservative generalization of RAF, they inherit a one-to-one correspondence with $AFRA$ in the case of the complete, preferred and stable semantics but only when there is *no support* (so when $RAFN$ are reduced to RAF).

5.3 Higher-order evidential supports

In [30], the RAF is extended with the introduction of evidential supports. Recall that, as presented in Section 4.3, the evidential understanding of the support relation introduced in [80] allows to distinguish between two different kinds of arguments: *prima-facie* and *standard arguments. Prima-facie* arguments were already present in [95] as those that are justified whenever they are not defeated. On the other hand, *standard arguments* are not directly assumed to be justified and must inherit support from prima-facie arguments through a chain of supports.

This extension of RAF, called Recursive Evidence-Based Argumentation Framework ($REBAF$), can be defined as follows:

Definition 49 (Def. 13 in [30]). *A* recursive evidence-based argumentation framework *(REBAF) is a sextuple $\langle Ar, att, sup, \mathbf{s}, \mathbf{t}, PF\rangle$ where Ar, att and sup are three (possible infinite) pairwise disjunct sets respectively representing arguments, attacks and supports names; $PF \subseteq Ar \cup att \cup sup$ is a set representing the* prima-facie *elements that do not need to be supported; functions $\mathbf{s} : (att \cup sup) \longrightarrow 2^{Ar} \setminus \varnothing$ and $\mathbf{t} : (att \cup sup) \longrightarrow (Ar \cup att \cup sup)$ respectively map each attack and support to its source and its target.*

Then the definition of $REBAF$ semantics uses similar notions and techniques to the ones used in [31] for the $RAFN$. For instance, the notion of structure in $REBAF$ and, given a structure U, the sets $Def(U)$ and $Def_X(U)$ exactly correspond to the equivalent notions in $RAFN$. The other notions are of course adapted to account for the constraints emerging from the evidential interpretation of the support relation:

Definition 50 (Sec. 3.2 in [30]). *Let $\langle Ar, att, sup, \mathbf{s}, \mathbf{t}, PF\rangle$ be a $REBAF$. Let $U = (S, \Gamma, \Delta)$ be a structure of $REBAF$.*

- $Supp(U) = PF \cup \{\mathbf{t}(\alpha) | \exists \alpha \in \Delta \cap Supp(U_{-\mathbf{t}(\alpha)}), \mathbf{s}(\alpha) \subseteq S \cap Supp(U_{-\mathbf{t}(\alpha)})\}$ *with*[21] $U_{-\mathbf{t}(\alpha)} = U \backslash \{\mathbf{t}(\alpha)\}$.
- $UnAcc(U) = Def(U) \cup \overline{Supp(U')}$ *with* $U' = (\overline{Def_{Ar}(U)}, att, \overline{Def_{sup}(U)})$.
- $UnAct(U) = \{\alpha \in att | \alpha \in UnAcc(U)$ *or* $\mathbf{s}(\alpha) \cap UnAcc(U) \neq \varnothing\}$.

[21] By abuse of notation, we write $U \backslash X$ instead of $(S \backslash X, \Gamma \backslash X, \Delta \backslash X)$ with $X \subseteq (Ar \cup att \cup sup)$.

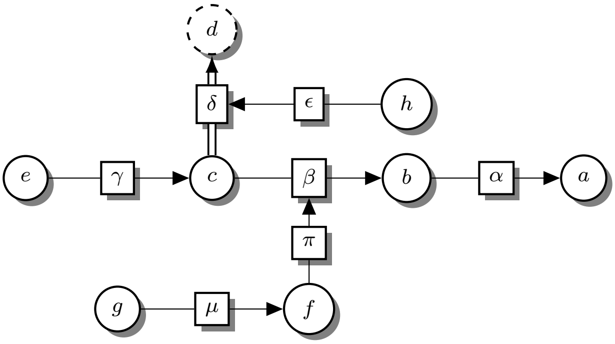
Note that the notion of self-supporting structure in $REBAF$ is the same as the one given for $RAFN$. Then using these notions, the definitions for acceptability, admissibility, conflict-freeness and also for the complete semantics given for $RAFN$ (see Definition 48) can be reused. Some differences appear for the preferred and stable semantics; furthermore, no definition is given in [30] for the grounded semantics, but a definition is proposed in [41]:

Definition 51. (Defs. 16 in [30] and 2.14 in [41]) *Let* $REBAF = \langle Ar, att, sup, \mathbf{s}, \mathbf{t}, PF\rangle$. *Let* $U = (S, \Gamma, \Delta)$ *be a structure of* $REBAF$.

- U *is* preferred *iff it is a* $\subseteq$*-maximal admissible structure,*
- U *is* stable *iff* $(S \cup \Gamma \cup \Delta) = \overline{UnAcc(U)}$.[22]
- U *is* grounded *iff it is a* $\subseteq$*-minimal complete structure.*

f be illustrated on Example 8.

Example 8 (cont'd) *First of all, we must choose the set of prima-facie elements. Indeed, without prima-facie elements, most semantics will yield an empty set. Like for EBAF, elements that are not the target of a support are assumed to be prima-facie (the prima-facie elements are represented using solid outlines whereas standard elements are represented with dashed outlines). So, the only standard element is the argument d.*



Note that the structure $(\{b, e, g, h\}, att, \varnothing)$ *is the only complete, grounded, preferred and stable structure, as in the* $RAFN$ *case. Here, d is not acceptable since it is not supported (its support being attacked: both* δ *and its source c are attacked).*

In [30], several links with other approaches have been proven:

[22] Note that this already implies conflict-freeness.

- In Section 4 in [30]): $REBAF$ are a conservative generalization of RAF, and so inherit a one-to-one correspondence with $AFRA$ in the case of the complete, preferred and stable semantics but only when there is *no support* and when *each element is prima-facie.*

- In Section 5 in [30]): a one-to-one correspondence between $REBAF$ and finite $EBAF$; this correspondence does not work when we consider non-finite $EBAF$.

- In Section 4 in [30]): since the type of support used in $ASAF$ (necessary support) is different from the one used in $REBAF$ (evidential support), no correspondence can be established. And the same result occurs with $BAFD$ (deductive support).

- In Section 6 in [30]): $REBAF$ are a conservative generalization of AF considering the notion of d-structure (see Definition 28).

5.4 Comparison between Higher-order bipolar approaches: a first and succinct summary

Throughout Section 5, many differences and similarities were highlighted in order to compare the four approaches introduced in this section (Higher-order deductive framework–Section 5.1–, $ASAF$ –Section 5.2.1–, $RAFN$–Section 5.2.2– and $REBAF$–Section 5.3); moreover some links with the higher-order approaches from Section 3 were also given. All these comparison points have been introduced when it was pertinent (depending on the definitions and examples discussed at that point in the text). So in order to facilitate the reading and the understanding of this chapter, the main comparison points are recalled here.

- First of all, as for the higher-order approaches from Section 3, most of the presented higher-order bipolar approaches are a conservative generalization of Dung's framework when neither higher-order nor bipolar interactions exist. The only (partial) exception is the $ASAF$ since it is inspired on the $AFRA$ and so it inherits the same problem: the generalization holds only at the semantics level but not for the basic semantic notions (such as, for instance, conflict-freeness).

- Second, since each presented higher-order bipolar approach is built upon a specific higher-order approach, the bipolar version is a conservative generalization of the framework it is based on when no supports exist. So this link exists between the higher-order deductive framework and the EAF, between the $ASAF$ and the $AFRA$, and between the $RAFN$ or the $REBAF$ and the RAF. Of course, the same result holds (with some nuances) when we compare a higher-order bipolar approach and the bipolar framework it is based on, when no higher-order interactions exist.

- Third, because of the three types of support they consider, it is difficult to establish links between all higher-order bipolar approaches. So higher-order deductive frameworks are not comparable with $ASAF$, $RAFN$ or $REBAF$; $ASAF$ or $RAFN$ are not comparable with the higher-order deductive framework or $REBAF$; and the $REBAF$ is not comparable with the three other frameworks.

- And finally, in [31], a comparison between $ASAF$ and $RAFN$ has been carried out, yielding the same results as the ones between $AFRA$ and RAF: these two approaches do not coincide even if there exists a one-to-one correspondence.

Another point of comparison, not addressed by any of the higher-order bipolar approaches, regards the way in which they treat support cycles: whether they prevent them in the definition of the framework, whether they allow them but reject them in the definition of the semantics, etc.

As it can be noted in Definition 39, the $ASAF$ requires the support relation to be acyclic. As argued by the authors in [61], this restriction is inspired on the restrictions placed on the support relation of the AFN (see Def. 31), in which the support relation is required to be irreflexive and transitive. On the one hand, by being acyclic, the support relation of the $ASAF$ is also irreflexive; on the other hand, the transitive nature of necessary support is captured in the $ASAF$ by explicitly considering a sequence of supports in the definition of the conditional defeats. In contrast, we can note that neither the $BAFD$ with second-order attacks, the $RAFN$ nor the $REBAF$ impose restrictions on the support relation of the framework.

Given the deductive interpretation of support adopted by the $BAFD$, we can note that the existence of support cycles would be resolved by the corresponding Dung semantics in the translated AF. If we take the simplest odd-length support cycle, we can consider a self-supporting argument a; in such a case, the translated AF would be such that an even-length attack cycle between a and Z_{aa}. Similarly, a two-length support cycle between two arguments a and b would yield an even-length attack cycle in the translated AF, namely: $b \to Z_{ab}$, $Z_{ab} \to a$, $a \to Z_{ba}$ and $Z_{ba} \to b$. In both cases (odd-length and even-length support cycles), the resulting attack cycle in the AF would be of even length. Consequently, unless the cycle is broken, the arguments involved in the support cycle would be rejected by the grounded semantics, and possibly accepted by the complete, preferred or stable semantics.

At last, the treatment of support cycles in $RAFN$ and $REBAF$ is analogous. Both frameworks allow support relations originating in a set of arguments. Then, they define the set of supported elements by a structure, in which they prevent an element from being supported by itself (by considering U_{-x} in Definition 47 and $U_{-\mathbf{t}(\alpha)}$ in Definition 50). Consequently, since the acceptable elements w.r.t. a structure have to be supported by the structure, this prevents the semantics from accepting an argument that is just supported by itself (either directly or indirectly).

A more complete analysis of the four higher-order bipolar approaches is given in Section 7.

6 Computational issues and some applications

This section starts by introducing computational approaches that implement alternative semantics for some of the frameworks discussed in Sections 3 and 5. Then, we briefly discuss some applications of these frameworks or their underlying ideas to solve problems such as finding solutions to the *liar paradox* [55] and the construction of deductive mathematical proofs.

6.1 Computational issues

Several works concern the semantics computation for higher-order frameworks. They describe either logical approaches, or the use of dialectical proofs, or some more direct algorithms.

6.1.1 ASP Encodings for EAF and $AFRA$

As discussed in [53], and also evidenced by the different editions of the *International Competition on Computational Models of Argumentation (ICCMA)*,[23] reduction-based approaches for the implementation of argumentation related problems have become very popular. Among others, reductions to Answer Set Programming (ASP) [66; 75] and propositional logic became suitable for the relevant reasoning problems [92; 59].

In [53] the authors proposed an ASP reduction-based approach to compute acceptability in Modgil's EAF. For that purpose, they proposed an alternative (but equivalent) characterization for the acceptance of arguments in an EAF, which allowed them to design succinct ASP encodings for all standard semantics of the EAF.

[23] http://argumentationcompetition.org

Briefly, the new characterization of acceptability for EAF given in [53] relies on the consideration of a single reinstatement set for the defeats. As shown by the authors, since the union of two reinstatement sets for the same set of arguments S is also a reinstatement set, there exists a unique maximal reinstatement set.

Based on the new definitions, they proposed ASP encodings for EAF. Briefly, the answer-sets of the combination of an encoding for a semantics σ with an ASP representation of an EAF are in one-to-one correspondence to the set of σ-extensions of this EAF. The encoding is partitioned into several modules, and they begin with an input database for a given $EAF = \langle Ar, att, att2 \rangle$. Next we introduce the facts encoding an EAF; for further details and a full description of the encodings, including the definition of modules for each semantics, we refer the reader to [53]:

$$\begin{aligned}\widehat{EAF} := \quad & \{\mathbf{arg}(x).\mid x \in Ar\} \cup \\ & \{\mathbf{att}(x,y).\mid (x,y) \in att\} \cup \\ & \{\mathbf{d}(x,y,z).\mid (x,(y,z)) \in att2\}\end{aligned}$$

It is worth mentioning that their proposed encodings were incorporated within *ASPARTIX - Answer Set Programming Argumentation Reasoning Tool*,[24] an ASP-based argumentation system for representing and evaluating Dung's AF semantics and some of its extended frameworks, such as Modgil's EAF. In particular, the evaluation of semantics over an EAF is provided in the web-interface *GERD - Genteel Extended argumentation Reasoning Device*.[25] Figure 5 illustrates the use of the *GERD* tool on the EAF of Example 3, where argument c' is denoted as *cp*.

As mentioned before, different ASP encodings for Dung's framework exist (see *e.g.* [54]). Then, based on the encodings for Dung's framework and its semantics, *ASPARTIX* also offers the possibility to evaluate the $AFRA$ semantics. In order to be able to use those, an $AFRA$ is encoded by an ASP encoding similar to the one provided for the EAF, with the addition of some predicates allowing to translate the $AFRA$ into an AF (following the translation described at the end of Section 3.3). The corresponding encoding provided in the *ASPARTIX* website[26] is shown below.

An $AFRA$ is encoded by a sequence of statements where each statement either encodes an argument, or an attack between arguments, or an attack towards another attack. The facts representing $AFRA = \langle Ar, att \rangle$ are:

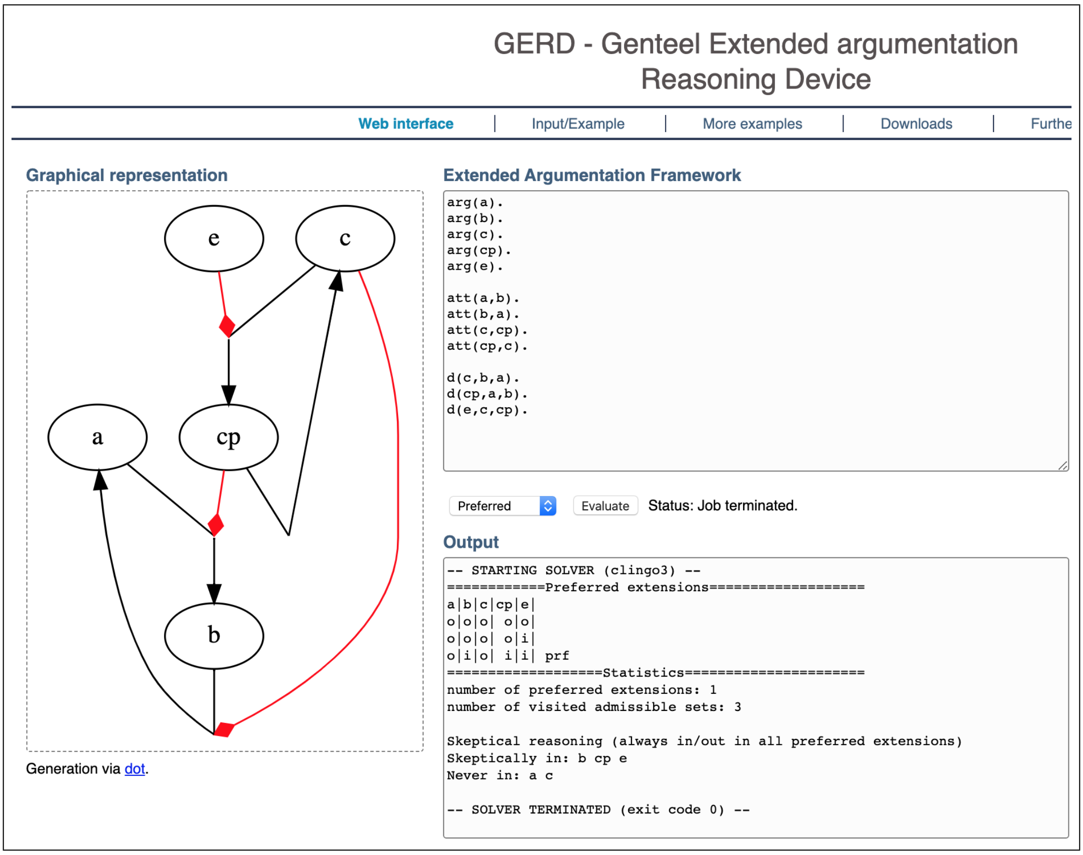
$$\widehat{AFRA} := \{\mathbf{afraA}(x).\mid x \in Ar\} \cup \{\mathbf{afraR}(\alpha,x,y).\mid \alpha = (x,y) \in att\}$$

Finally, the ASP implementation of the translation from an $AFRA$ into an AF is shown in Figure 6.

[24] http://www.dbai.tuwien.ac.at/research/argumentation/aspartix

[25] http://gerd.dbai.tuwien.ac.at

[26] https://www.dbai.tuwien.ac.at/research/argumentation/aspartix/afra.html


GERD - Genteel Extended argumentation Reasoning Device
Web interface | Input/Example | More examples | Downloads | Furthe
Graphical representation
e
c
a
cp
b
Generation via dot.
Extended Argumentation Framework
Preferred
Evaluate
Output

```
arg(a).
arg(b).
arg(c).
arg(cp).
arg(e).

att(a,b).
att(b,a).
att(c,cp).
att(cp,c).

d(c,b,a).
d(cp,a,b).
d(e,c,cp).
```

Status: Job terminated.

```
-- STARTING SOLVER (clingo3) --
============Preferred extensions===================
a|b|c|cp|e|
o|o|o| o|o|
o|o|o| o|i|
o|i|o| i|i| prf
==================Statistics=====================
number of preferred extensions: 1
number of visited admissible sets: 3

Skeptical reasoning (always in/out in all preferred extensions)
Skeptically in: b cp e
Never in: a c

-- SOLVER TERMINATED (exit code 0) --
```

Figure 5: Screenshot of *GERD* illustrating the *EAF* of Example 3 and its preferred extension $\{e, c', b\}$.

$\%$ *arguments*

$\mathbf{arg}(X) \leftarrow \mathbf{afraA}(X).$

$\mathbf{arg}(R) \leftarrow \mathbf{afraR}(R, X, Y),\ \mathbf{afraA}(X).$

$\%$ *direct defeat*

$\mathbf{att}(V, W) \leftarrow \mathbf{afraR}(V, X, W),\ \mathbf{arg}(W),\ \mathbf{afraA}(X).$

$\%$ *indirect defeat*

$\mathbf{att}(V, A) \leftarrow \mathbf{att}(V, W),\ \mathbf{afraR}(A, W, X),\ \mathbf{afraA}(W).$

Figure 6: ASP encoding to translate an *AFRA* into an *AF*.

6.1.2 Logical encoding of *REBAF* and *RAF*

Another logical approach is presented in [40; 41]. In these works, the authors use a three-sorted logic with equality in order to encode several variants of argumentation frameworks (*AF*, *RAF* and *REBAF*). With that work, the authors want to characterize in a logical way the meaning of each type of interaction, to encode the acceptance condition for arguments and interactions and then provide a computational issue for the semantics of these argumentation frameworks.

In this logic, the three sorts are: `arg` a sort for arguments, `att` a sort for attacks and `esup` a sort for evidential supports. Two function symbols s and t can be applied to objects of the sort `att` or `esup` to capture source and target of these interactions. The target can be either of sort `arg` or of sort `att` or of sort `esup` and the source can only be of sort `arg`. Note that this encoding takes into account only the case of interaction sources that are singletons.

Different unary predicates are also used for encoding each element of the argumentation framework: For a node a of the argumentation graph, $Acc(a)$ expresses the status of being accepted, whereas $Nacc(a)$ expresses that a cannot be accepted (implicitly: w.r.t. a given semantics); in other words, the meaning of $Nacc(a)$ is stronger than "a is not accepted". The language also admits atoms of the form $Val(\alpha)$ for attack or support names (intuitively, $Val(\alpha)$ means that the interaction named α is valid w.r.t. a given argumentation semantics). There is also the predicate symbol $PrimaFacie$ for denoting prima-facie elements (so for arguments and interactions).

Since one purpose is to obtain a logical characterization of structures, and so of acceptability, some additional unary predicate symbols are given: *Supp* for denoting supported elements (arguments, attacks or supports), *UnSupp* for denoting unsupportable elements and *eAcc* (resp. $eVal$) for denoting acceptability for arguments (resp. for interactions, *i.e.* attacks or supports). Note that $eAcc(x)$ ("x is e-accepted") can be understood as "x is accepted and supported" and similarly $eVal(\alpha)$ ("α is e-valid") can be understood as "α is valid and supported".

Using this vocabulary, the formulae describing a given argumentation framework, for instance a $REBAF$, can be partitioned in two sets:

- The first set contains the formulae describing the general behaviour of each interaction, possibly recursive, *i.e.* how an interaction interacts with arguments and other interactions related to it.

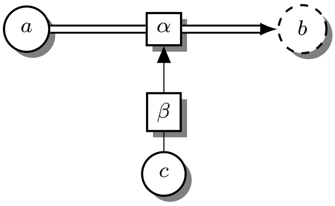
- The second set contains the formulae encoding the specificities of the current framework (enumeration of the arguments and interactions that belong to this framework).

Then, several formulae are introduced for encoding the different principles that govern argumentation semantics. There are formulae for capturing the defence principle, the reinstatement principle and the stability principle.[27] Then extensions under a given semantics (admissible, complete, preferred, grounded, or stable) can be characterized by models of logical theories obtained by combining some of these formulae.

Note that, if we consider finite argumentation frameworks, all the previous formulae can be rewritten in propositional logic and a SAT solver is enough for computing the structures resulting from $REBAF$ semantics.

Next we provide a very simple example in order to illustrate these ideas and notions, and present its complete encoding.

Example 12 (Ex. 1.2 in [41]).
Consider the following $REBAF$.



The set of formulae describing this $REBAF$ is the following:

$\Sigma(REBAF) = \{(eVal(\beta) \wedge eAcc(c)) \to \neg Val(\alpha),$
$Supp(a),$
$Supp(c),$
$Supp(\alpha),$
$Supp(\beta),$
$(eAcc(a) \wedge eVal(\alpha)) \to Supp(b),$
$(Supp(a) \wedge Acc(a)) \leftrightarrow eAcc(a),$
$(Supp(b) \wedge Acc(b)) \leftrightarrow eAcc(b),$
$(Supp(c) \wedge Acc(c)) \leftrightarrow eAcc(c),$
$(Supp(\alpha) \wedge Val(\alpha)) \leftrightarrow eVal(\alpha),$
$(Supp(\beta) \wedge Val(\beta)) \leftrightarrow eVal(\beta)\ \}$

The following ideas are used for obtaining $\Sigma(REBAF)$: first, a prima-facie element is supported. Second, an element is e-accepted if and only if it is accepted and supported. Third, if an attack and its source are e-accepted, then its target cannot be accepted (resp. valid). And finally, if a support and its source are e-accepted, then its target is supported.

[27] Note that the first set describes the conflict-freeness principle.

Then, the set of formulae $\Sigma_{ss}(REBAF)$ describing the notion of supported/unsupportable element is obtained from $\Sigma(REBAF)$ by adding formulae among which:

$Supp(b) \rightarrow (eAcc(a) \wedge eVal(\alpha))$
$\neg UnSupp(a)$
$\neg UnSupp(c)$
$\neg UnSupp(\alpha)$
$\neg UnSupp(\beta)$
$Unsupp(b) \leftrightarrow \Big(\ (eVal(\beta) \wedge eAcc(c)) \vee\ UnSupp(a) \vee\ UnSupp(\alpha)\ \Big)$

The first formula in $\Sigma_{ss}(REBAF)$ expresses the fact that if the target of a support is supported, then this support and its source are e-accepted. The other formulae correspond to the unsupported status: first, a prima facie element is not unsupported; second, if the target of a support is unsupported, then this support is not valid, or this support or its source are unsupported.

The set of formulae $\Sigma_d(REBAF)$ describing the principle of defence is obtained from $\Sigma_{ss}(REBAF)$ by adding formulae among which:

$Val(\alpha) \rightarrow (UnSupp(\beta) \vee UnSupp(c))$

The previous formula describes the defense of α: if α is defended (so valid) then its attacker β or the source of β are unsupported (here, this is the only way to invalidate the attack on α since neither β nor its source are in turn attacked).

The principle of reinstatement is expressed using the set of formulae $\Sigma_r(REBAF)$ obtained from $\Sigma_{ss}(REBAF)$ by adding the formulae:

$Acc(a)$
$Acc(b)$
$Acc(c)$
$Val(\beta)$
$(UnSupp(c) \vee UnSupp(\beta)) \rightarrow Val(\alpha)$

The four first formulae correspond to the case of an unattacked element: it does not need a defense for being accepted or valid. The last formula gives the condition for the reinstatement of an element that is the target of an attack (the reverse condition of the one given for the defense).

And finally, $\Sigma_s(REBAF)$ describing the stability principle is obtained from $\Sigma_{ss}(REBAF)$ by adding the formulae:

$Acc(a)$
$Acc(b)$
$Acc(c)$
$Val(\beta)$
$\neg Val(\alpha) \rightarrow eVal(\beta) \wedge eAcc(c)$
$\neg Supp(x) \rightarrow UnSupp(x)$ *for* $x \in \{a, c, \alpha, \beta\}$

The formulae in $\Sigma_s(REBAF)$ give the impact of either non-accepted/non-valid elements, or non-supported elements. For instance, if α is not valid, then its attacker β and the source of β are in the extension (so resp. e-valid and e-accepted).

From $\Sigma_d(REBAF)$ it can be deduced that $\neg Val(\alpha)$ then $\neg eVal(\alpha)$, $\neg Supp(b)$ and $\neg eAcc(b)$. That corresponds to the fact that no admissible structure contains b (resp. α, though being supported).
Moreover, there is a model of $\Sigma_d(REBAF)$ satisfying $eAcc(a)$, $eAcc(c)$ and $eVal(\beta)$. That corresponds to the fact that $(\{a, c\}, \varnothing, \{\beta\})$ is an admissible structure. This is also a $\subseteq$-maximal model. That corresponds to the fact that $(\{a, c\}, \{\beta\}, \varnothing)$ is a preferred structure; this is also a complete structure (since it corresponds to a model of $\Sigma_d(REBAF) \cup \Sigma_r(REBAF)$) and a stable structure (since it corresponds to a model of $\Sigma_s(REBAF)$).

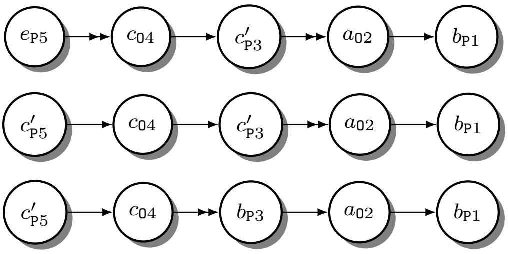
6.1.3 Dialectical proof procedure for Modgil's *EAF*

In addition to characterizing labellings for the EAF, in [68] the author defined a dialectical framework for EAF game proof theories, allowing to establish the justified status of an argument to be tested, and providing a basis for algorithmic development of EAF semantics. Analogously to dialectical proof procedures for Dung's AF, such theories consider a dialogue between two players: `P` (proponent) and `O` (opponent), each of which are referred to as the other's counterpart. A game begins with `P` moving an initial argument x to be tested. Then, `O` and `P` take turns in moving arguments that attack their counterpart's last move, where attacks can be either on an argument or an attack posed by their counterpart; alternatively, the players can also backtrack to a counterpart's previous move and initiate a new dialogue. In particular, Modgil's approach assumes the use of a finite EAF containing a finite number of arguments (thus, a finite number of attacks).

Then, a legal move function ϕ_{PC} is defined, which places restrictions on the players' moves, for the preferred credulous game (*i.e.* for determining whether an argument belongs to some preferred extension of the corresponding EAF). As argued by the author, since every admissible set of an EAF is a subset of a preferred extension, it suffices to show membership to an admissible set in order to show membership to a preferred extension. Briefly, the ϕ_{PC} game is a tree of ϕ_{PC}-dialogues whose root is `P`'s initial move of an argument. Also, the ϕ_{PC} function is such that it prevents `O` from moving arguments and attacks that have already been attacked by `P` in a dialogue d, since `P` will have already fulfilled its burden of defense with respect to these arguments/attacks. In addition, `P` can only move an argument x in d if: 1) x does not attack itself; 2) no argument y, and no attack (y, x) or (x, y) has been moved by `P`; and 3) x does not symmetrically attack some y moved by `P`.

Next, we illustrate Modgil's approach on the weather example:

Example 3 (cont'd) *Given the EAF representing the weather example, three ϕ_{PC} winning strategies for b are depicted below. The different moves in each strategy are identified by the argument put forward, the player introducing the argument, and a number indicating the order in which they are played. Also, the notation $a \rightarrow$ means that argument a attacks the previous argument, and the notation $c \twoheadrightarrow$ means that argument c attacks the attack between the two previous arguments:*



It should be noted that each winning strategy for b consists of a single dialogue. This is because the opponent O *has no alternatives to counter-attack the arguments/attacks put forward by the opponent* P*. Also, each strategy corresponds to an admissible set of EAF, from top to bottom: $\{e, c', b\}$ and $\{b, c'\}$ (the admissible set for the last two strategies coincides). Hence, b is (credulously) accepted w.r.t. the only preferred extension $\{e, c', b\}$ of EAF.*

6.1.4 Dialectical proof procedure for i-defense semantics of *HLAF*

In [62] the authors introduced a dialectical proof procedure for their inductive defense semantics of *HLAF* (see Section 3.4) based on [52; 91], where two unified frameworks of dialectical proof procedures were proposed.

Similarly to Modgil's approach discussed in the previous section, [62] proposes to evaluate the acceptability of arguments by resolving disputes between two players identified as proponent and opponent. They propose to represent disputes through *dispute derivations*, in which tuples $t_i = \langle P_i, O_i, SP_i, SO_i \rangle$ summarizing the history of the dispute up to step i are successively constructed by expanding the previous one. Given $HLAF = \langle Ar, att \rangle$, the set $P_i \subseteq Ar \cup att$ in each tuple represents the set of arguments and attacks put forward by the proponent (up to step i) that have not been defended by the proponent and hence are open to attacks by the opponent. Also, $SP_i \subseteq Ar$ is the set of all arguments presented by the proponent (up to step i). Consequently, the proponent does not need to re-defend arguments in $SP_i \backslash P_i$. On the other hand, $O_i \subseteq att$ is a set of attacks of the opponent against arguments presented by the proponent in previous steps that are not yet counter-attacked by the proponent. Thus, an attack $\alpha = (a, b) \in O_i$ needs to be counter-attacked by the proponent on either a or α. In addition, $SO_i \subseteq att$ is the set containing attacks by the opponent (up to step i) that have been counter-attacked by the proponent.

Thus, a dispute derivation for an argument a is a sequence of the tuples described above, satisfying the following conditions:

- $P_i \subseteq Ar \cup att$; $SP_i \subseteq Ar$; and $O_i, SO_i \subseteq att$.
- $P_0 = SP_0 = \{a\}$, and $O_0 = SO_0 = P_n = O_n = \varnothing$.
- At step i, an element X is selected from either P_i (*i.e.* an argument or attack put forward by the proponent that has to be defended) or from O_i (*i.e.* an attack from the opponent that has to be counter-attacked). The sets corresponding to the next tuple $(i + 1)$ are obtained as follows:
    - if $X \in P_i$: $P_{i+1} = P_i \backslash \{X\}$, $O_{i+1} = O_i \cup \{\alpha \mid \alpha = (Y, X) \in att\}$, $SP_{i+1} = SP_i$ and $SO_{i+1} = SO_i$; or
    - if $X \in O_i$: $O_{i+1} = O_i \backslash \{X\}$, $SO_{i+1} = O_i \cup \{X\}$, P_{i+1} augments P_i with an attack α targeting X and with the source of α (as long as the latter does not already belong to SP_i), and SP_{i+1} augments SP_i with the source of attack α (if not already present).

It should be noted that, since at each step the selection can be made from P_i or O_i, the sequence of steps does not necessarily correspond to alternating moves by the different players. Consecutive selections from P_i would correspond to consecutive plays by the opponent (searching to attack the selected proponent's argument or attack), whereas consecutive selections from O_i would correspond to consecutive plays by the proponent (searching to counter-attack the opponent's selected attack).

Then, the authors showed that if $\langle P_0, O_0, SP_0, SO_0 \rangle \ldots \langle P_n, O_n, SP_n, SO_n \rangle$ is a dispute derivation for an argument a, then SP_n is an i-admissible set that contains a. Let us now illustrate the construction of a dispute derivation on the weather example.

Example 3 (cont'd) *The construction of a dispute derivation for b is depicted in Figure 7, where the notation $\underline{X}$ means that X is selected in the corresponding step, and $Attack_X = \{\alpha \in att \mid \alpha = (a, X)\}$. The sequence $\langle P_0, O_0, SP_0, SO_0 \rangle \ldots \langle P_9, O_9, SP_9, SO_9 \rangle$ is a dispute derivation showing that b is acceptable w.r.t. its constructed i-admissible set $SP_9 = \{b, c', e\}$ which, in particular, is the only i-preferred extension of the HLAF.*

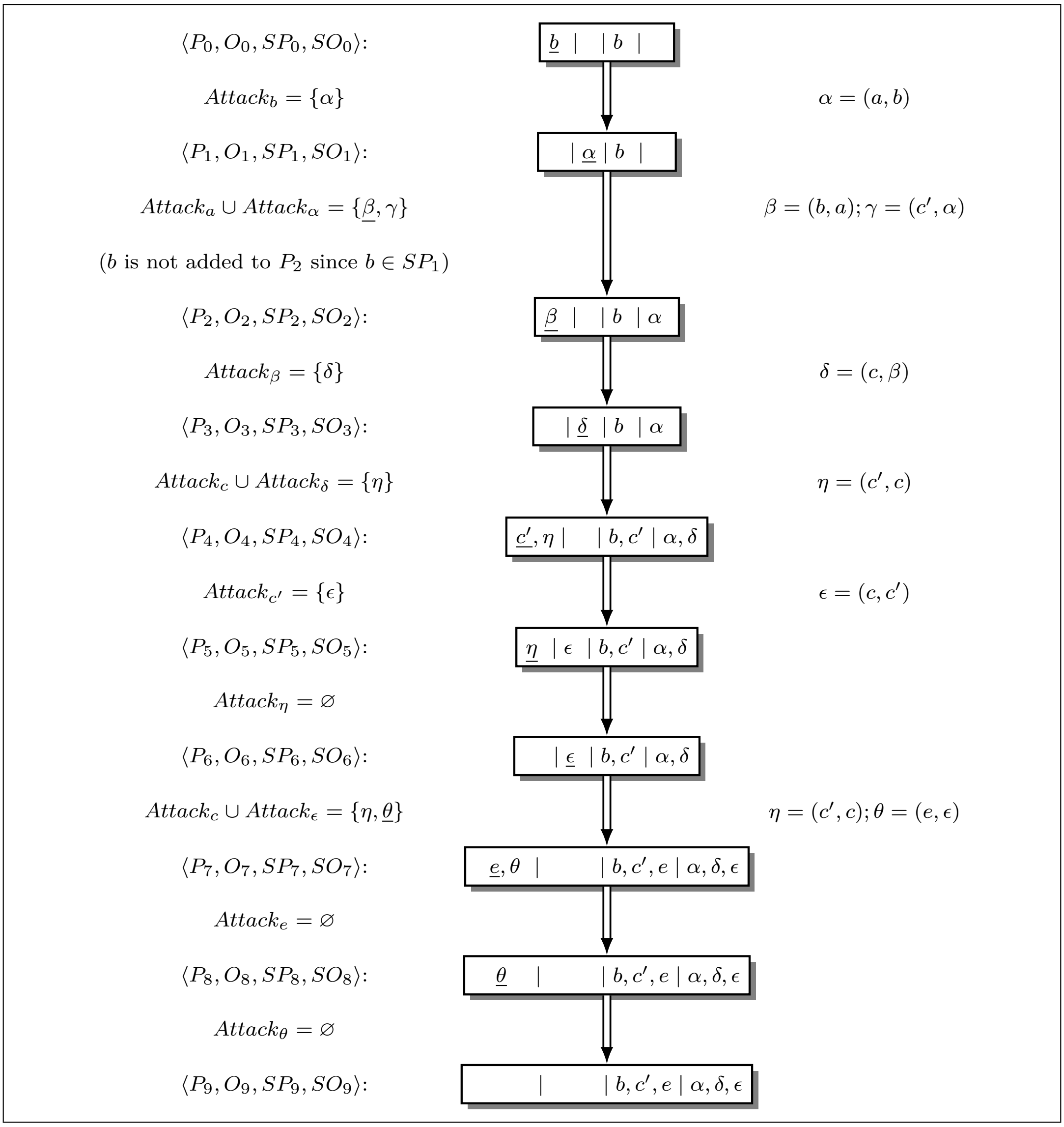

$\langle P_0, O_0, SP_0, SO_0 \rangle$:
$\underline{b} \mid \quad \mid b \mid$
$Attack_b = \{\alpha\}$
$\alpha = (a, b)$
$\langle P_1, O_1, SP_1, SO_1 \rangle$:
$\mid \underline{\alpha} \mid b \mid$
$Attack_a \cup Attack_\alpha = \{\underline{\beta}, \gamma\}$
$\beta = (b, a); \gamma = (c', \alpha)$
(b is not added to P_2 since $b \in SP_1$)
$\langle P_2, O_2, SP_2, SO_2 \rangle$:
$\underline{\beta} \mid \quad \mid b \mid \alpha$
$Attack_\beta = \{\delta\}$
$\delta = (c, \beta)$
$\langle P_3, O_3, SP_3, SO_3 \rangle$:
$\mid \underline{\delta} \mid b \mid \alpha$
$Attack_c \cup Attack_\delta = \{\eta\}$
$\eta = (c', c)$
$\langle P_4, O_4, SP_4, SO_4 \rangle$:
$\underline{c'}, \eta \mid \quad \mid b, c' \mid \alpha, \delta$
$Attack_{c'} = \{\epsilon\}$
$\epsilon = (c, c')$
$\langle P_5, O_5, SP_5, SO_5 \rangle$:
$\underline{\eta} \mid \epsilon \mid b, c' \mid \alpha, \delta$
$Attack_\eta = \varnothing$
$\langle P_6, O_6, SP_6, SO_6 \rangle$:
$\mid \underline{\epsilon} \mid b, c' \mid \alpha, \delta$
$Attack_c \cup Attack_\epsilon = \{\eta, \underline{\theta}\}$
$\eta = (c', c); \theta = (e, \epsilon)$
$\langle P_7, O_7, SP_7, SO_7 \rangle$:
$\underline{e}, \theta \mid \quad \mid b, c', e \mid \alpha, \delta, \epsilon$
$Attack_e = \varnothing$
$\langle P_8, O_8, SP_8, SO_8 \rangle$:
$\underline{\theta} \mid \quad \mid b, c', e \mid \alpha, \delta, \epsilon$
$Attack_\theta = \varnothing$
$\langle P_9, O_9, SP_9, SO_9 \rangle$:
$\mid \quad \mid b, c', e \mid \alpha, \delta, \epsilon$

Figure 7: Construction of a dispute derivation for argument b corresponding to the $HLAF$ of Example 3 (arrows represent transitions between steps)

Finally, the authors in [62] stated that a proof procedure for i-defense semantics can be reduced to a procedure searching for dispute derivations, which could be directly implemented by means of, for instance, base derivations defined in [91].

6.1.5 Algorithmic approaches for computing extensions of argumentation frameworks with higher-order interactions

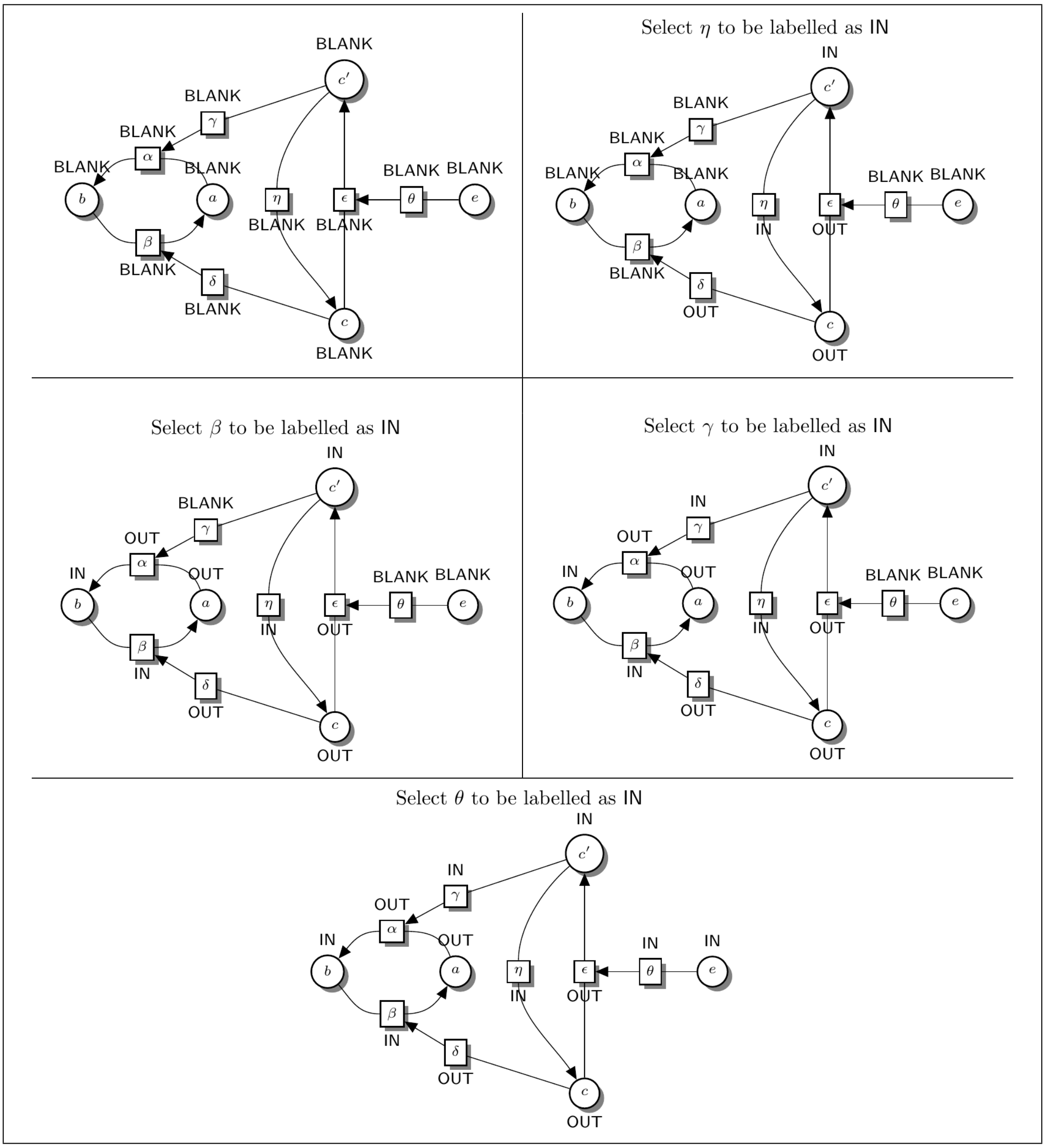
In this section we will briefly discuss different approaches proposed in the literature for computing the extensions of argumentation frameworks that include higher-order interactions.

- In [76] the authors proposed a series of algorithms allowing to enumerate the extensions of frameworks with higher-order attacks, such as the ones discussed in Section 3. In particular, they take the $AFRA$ as a case-study and propose algorithms for enumerating the preferred, stable, complete stage, semi-stable, ideal and grounded semantics of the framework. For illustration purposes, we will next describe the algorithm for obtaining the preferred extensions of an $AFRA = \langle Ar, att\rangle$, and show its application on Example 3.

  Briefly, the algorithm considers five labels: IN, OUT, MUST_OUT, BLANK and UNDEC. The BLANK label is the initial label for all arguments and attacks. In each iteration, a BLANK attack $\alpha \in att$ is labelled IN to indicate that α might be in a preferred extension. As a selection rule, attacks whose target is the source of the larger number of attacks are chosen first. Every time an attack is labelled IN, the labels of some attacks and arguments might change accordingly. An argument $a \in Ar$ is labelled OUT iff there is $\alpha \in att$ with the label IN such that $\mathbf{t}(\alpha) = a$. An attack $\beta \in att$ is labelled OUT iff there is $\alpha \in att$ with the label IN such that $\mathbf{t}(\alpha) \in \{\beta, \mathbf{s}(\beta)\}$. A BLANK argument a is labelled IN, implying that a might be in a preferred extension, iff there is $\alpha \in att$ with the label IN such that $\mathbf{s}(\alpha) = a$, or for each $\beta \in att$ such that $\mathbf{t}(\beta) = a$, the label of β is OUT. Then, each attack $\beta \in att$ with the label BLANK or UNDEC is labelled MUST_OUT iff there is $\alpha \in att$ with the label IN such that $\mathbf{t}(\beta) \in \{\alpha, \mathbf{s}(\alpha)\}$; finally, if some problem arises at this point (inconsistency between the labels assigned), the chosen attack α is labelled UNDEC to try to find a preferred extension excluding it.

  Figure 8 exemplifies the algorithm to enumerate the preferred extensions on the $AFRA$ from Example 3, where attacks are selected to be labelled as IN in the following order: η, β, γ, θ.

- In [3] the authors proposed an algorithm for efficiently recomputing the extensions of $BAFD$s with or without second-order attacks (see Section 5.1)



Select η to be labelled as IN

Select β to be labelled as IN

Select γ to be labelled as IN

Select θ to be labelled as IN

Figure 8: Application of the algorithm to enumerate the preferred extensions of the $AFRA$ corresponding to Example 3. The final labelling corresponds to the only preferred extension $\{e, \theta, c', \eta, \gamma, b, \beta\}$

after an update on the framework has been performed. Briefly, an update consists of the addition or removal of an argument, an attack or a support; however, as highlighted in [3], updates concerning an argument can be easily performed without requiring to recompute an extension. Their algorithm builds on the incremental approach proposed for Dung's AF in [2] and, given an initial $BAFD$, a semantics, an initial extension for it under the chosen semantics and an update, it computes an extension of the updated $BAFD$. This is achieved by introducing a meta-argumentation translation (analogous to the one proposed in Definition 37) according to which an initial $BAFD$, as well as its extension and an update, are transformed into a Dung's AF with a suitable initial extension and update.

In addition, the authors identify different conditions under which an update over a $BAFD$ is *irrelevant*, in the sense that the original input extension is still an extension of the updated framework; for this, only the stable and preferred semantics are considered. In other words, irrelevant updates are still applied on the input framework, yielding an updated framework; what occurs in those cases is that the extension of the updated framework does not need to be recomputed. Whereas the conditions characterizing the irrelevant updates are defined in [3] in terms of labellings for the $BAFD$, no formal definition of labellings for $BAFD$ is given; instead, the extensions-labellings correspondence proposed for Dung's AF (see [27]) is exploited.

Finally note that, even though the algorithm of [3] was envisioned for computing an extension of an updated $BAFD$, it could also be iteratively used for computing an extension of a static $BAFD$ in the following way: start with the $BAFD$ containing all arguments and no attacks nor supports as initial framework, and the set of all arguments as the initial extension; then, add the attacks and supports one-by-one by considering them as updates, with the restriction that the second-order attacks have to be added after adding the interactions they target.

- In line with the work discussed in the previous item, [1] proposed an incremental approach for efficiently computing extensions of an $ASAF$ after performing an update, considering the complete, preferred, stable and grounded semantics. Differently from the previous approach for $BAFD$, labellings for the $ASAF$ were formally characterized in [1] (see Section 5.2.1) and accounted for in the developed algorithm.

  The approach of [1] also relies on a transformation of an $ASAF$ into a Dung's AF which, as argued by the authors, improves the one proposed in Definition 43 from two standpoints: *i*) it is direct, meaning that it does not require the two-step process of [61] which first obtains an AFN and then an AF; and *ii*) the size of the resulting AF is smaller than that of the one obtained by applying Definition 43. Notwithstanding this, as shown in [1], the translation they proposed yields equivalent extensions to those of the corresponding $ASAF$ under the considered semantics.

In addition, the authors formally characterized the irrelevant updates for an $ASAF$, for which an extension E of an updated $ASAF$ can be directly obtained without requiring its overall computation. Note that, like in the case of $BAFD$, irrelevant updates are still applied on the input $ASAF$, yielding an updated $ASAF$. However, differently from the case of $BAFD$, an $ASAF$ extension may also contain attacks and supports; therefore, in the presence of irrelevant updates, the updated extension will not necessarily coincide with the original extension but could easily be obtained without requiring its overall recomputation. On the one hand, for an irrelevant update deleting an attack or a support, an extension of the updated $ASAF$ can be simply obtained by deleting the corresponding interaction from the original extension. On the other hand, for an irrelevant update corresponding to an addition of an attack or a support, the situation depends on the nature of the interaction: whereas a support will always be added to the extension of the updated $ASAF$, an attack will only be added to the extension in cases where its source argument belonged to the original extension.

Finally note that, like in the case of the $BAFD$, the incremental algorithm for the $ASAF$ could be used for obtaining an extension of the framework in the static case. Furthermore, as argued by the authors in [1], their proposed translation from an $ASAF$ into an AF could be used for obtaining $ASAF$'s extensions even in the static case, where updates are not considered (and the same would hold for their translation of a $BAFD$ into an AF).

6.1.6 The Grafix tool

Several tools have been developed by the argumentation community, each of them having its specificities (see for instance, the web-interface *GERD* evoked in Section 6.1.1). Among them, the Grafix tool has been proposed for creating and handling enriched abstract argumentation graphs, in particular those with higher-order interactions (RAF, $REBAF$ and $RAFN$), following some of the approaches described in this chapter [29; 31; 32; 30; 40].

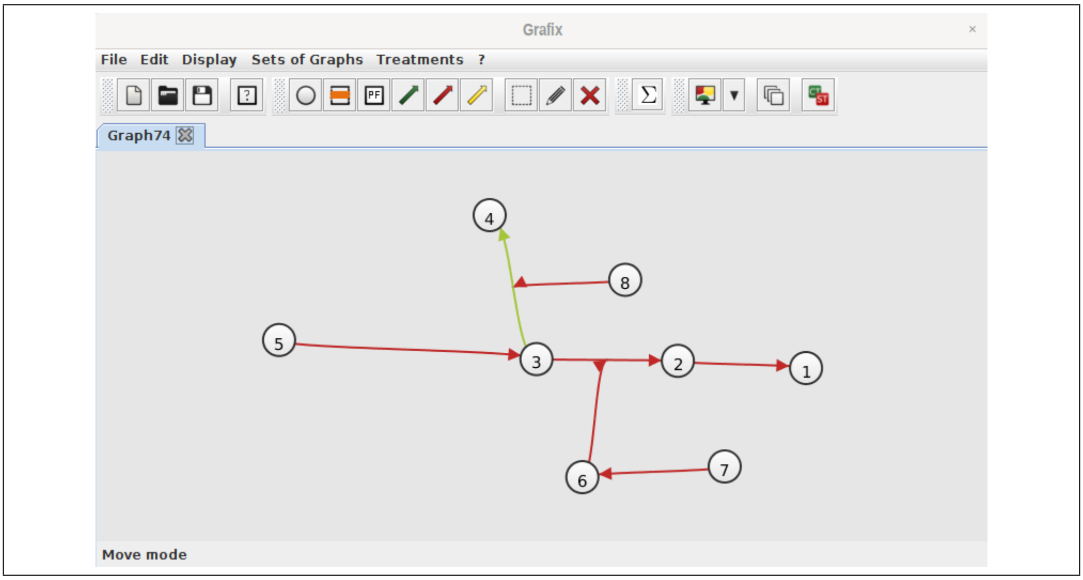


Figure 9: Vizualization of the *REBAF* version of Example 8 with Grafix Arguments are numbered as follows: 1 for a, . . . , 8 for h. Attacks (resp. supports) are represented with red (resp. green) arrows.

Grafix is a graphical tool[28] encoded in Java language (see [36]). It allows for the definition and the visualization of many kinds of argumentation graphs and the execution of some treatments on these graphs. Among these treatments, there is the computation of the well-known acceptability semantics. Another example of treatment is the translation of argumentation graphs into logical bases and the use of these bases for computing some acceptability semantics.

Figure 9 is a screenshot corresponding to the creation of Example 8 with this tool.

Then, Figure 10 shows the corresponding preferred structure computed with the Grafix tool when we consider that this framework is a *REBAF* and that all elements except from argument d are prima-facie.

6.2 Applications

In the literature, higher-order frameworks are used for representing and solving different problems. Here, we present two examples of such applications.

- [50] proposed the *Extended Explanatory Argumentation Framework* (*EEAF*),

[28] The visualization part of the tool is realized thanks to the `GraphStream` library (see [94]).

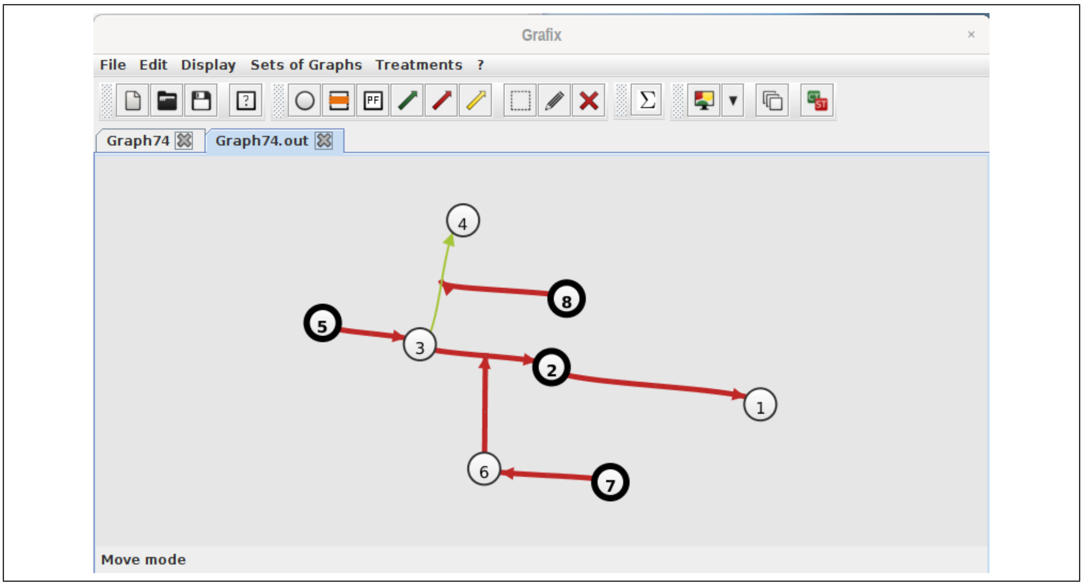


Figure 10: The preferred structure of the *REBAF* version of Example 8 with Grafix (all elements except *d*, numbered 4 in the figure, are prima-facie). Elements that belong to the preferred structure are in bold.

which extends the *Explanatory Argumentation Framework* of [90] by incorporating recursive attacks, joint attacks and a support relation. As argued by the authors, they apply the meta-argumentation methodology in order to incorporate these elements. The key feature of these frameworks is the existence of a set of *explananda* (scientific phenomenons of which, unlike arguments, the acceptability is not being questioned) and an explanatory relation relating arguments to other arguments or to *explanandum*, suitable for modelling the interaction between explanation and argumentation in scientific debates.

Definition 52 (Def. 18 in [50])**.** *An* Extended Explanatory Argumentation Framework *(EEAF) is a tuple* $\langle Ar, X, att, exp, inc, sup\rangle$*, where* Ar *is a set of arguments,* X *is a set of explananda,* $att \subseteq (2^{Ar} \cup exp \cup att) \times (Ar \cup exp \cup att \cup sup)$ *is a higher-order attack relation,* $exp \subseteq (Ar \times Ar) \cup (Ar \times X)$ *is an explanatory relation,* $inc \subseteq Ar \times Ar$ *is an incompatibility relation, and* $sup \subseteq Ar \times Ar$ *is a support relation.*

Note that the attack relation *att* not only allows for joint attacks and higher-order attacks, but also for attacks originating in other attacks. On the other hand, the incompatibility relation is used to identify opposing theories, as scientists usually do not accept multiple explanations of a given phenomenon at the same time.

Then, as argued by the authors in [50], the semantics of their $EEAF$ are defined by flattening their framework into an Explanatory Argumentation Framework. An Explanatory Argumentation Framework is a tuple $\langle Ar', att', X', exp', inc' \rangle$ (*i.e.* it has the same structure as the $EEAF$ minus the support relation), with the restriction that the attack relation att' is defined over pairs of arguments.

This translation is such that the set of arguments of the flattened Explanatory Argumentation Framework is comprised of: meta-arguments $acc(a)$ and $rej(a)$ for each argument in the $EEAF$, meta-arguments $X_{a,b}$ and $Y_{a,b}$ for each attack $(a, b) \in att$, meta-arguments $P_{a,b}$ and $Q_{a,b}$ for each pair $(a, b) \in exp$, a meta-argument $e(S)$ for each joint-attack having S as its set of originating arguments, and a meta-argument $Z_{a,b}$ for each pair of arguments $a, b \in Ar$. Also, the set of *explananda* in the flattened Explanatory Argumentation Framework is the same as the set of the corresponding $EEAF$. Then, the different relations of the $EEAF$ are mapped into the relations of its corresponding Explanatory Argumentation Framework by using the meta-arguments listed above.

Finally, the authors illustrate the applicability of the $EEAF$ on an example which focuses on two groups of solutions to the *liar paradox*. As stated by the authors, the arguments they considered are extracted from the book *Saving Truth from Paradox* [55].

Example 13 (Ex. from [50])**.** *Given the following arguments:*

ep: *This explanandum represents the paradox.*

a: *The paracomplete, paraconsistent and semi-classical solutions which provide explanations for the paradox by weakening classical logic.*

b: *The underspill and overspill solutions which provide their own explanation of the paradox by suggesting that for some predicates F, F is true of some objects that are not F or vice-versa.*

c: *We did not change logic to hide the defects in other flawed theories such as Ptolemaic astronomy, so why should we change the logic simply to hide these paradoxes?*

d: *There is no known way of saving these flawed theories such as Ptolemaic astronomy and even if there was, there is little benefit to doing so.*

f: *We have worked out the details of the new logics and they allow us to conserve the theory of truth.*

g: *Changing the logic implies changing the meaning.*

h: *Change of meaning is bad.*

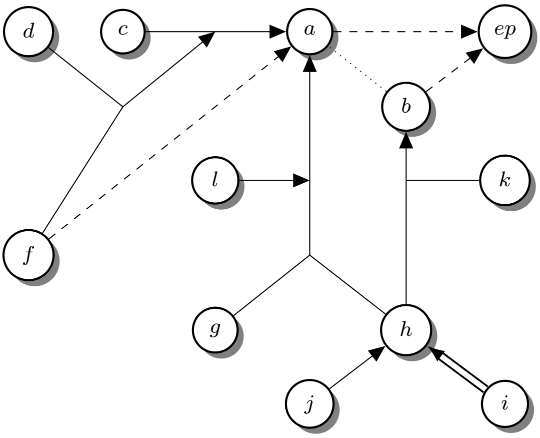
i: *The change is mere.*

j: *This is no "mere" relabelling.*

k: *Change of truth schema is a change of the meaning of "true".*

l: *The paradox forces a change of meaning.*

[50] proposed to model the knowledge in this discussion through the EEAF depicted below, where attacks (including joint attacks) are depicted using solid arrows, the support is depicted using a double arrow, the explanatory relation is depicted using dashed arrows, and the incompatibility relation is depicted with a dotted line:



The flattened Explanatory Argumentation Framework corresponding to this EEAF is depicted in Figure 11 taken from [50]; as argued by the authors, less-relevant auxiliary arguments are omitted in the figure for the sake of visibility (e.g. the $rej(X)$ *meta-arguments that do not attack other meta-arguments, and the* $Z_{X,Y}$ *meta-arguments for which no support* $(X, Y) \in sup$ *exists).*

Then, two argumentative core extensions are identified: $\{a, c, d, f, g, j, k, l\}$ *and* $\{b, c, d, f, g, j, k, l\}$*, each of which corresponds to the two rivaling solutions (because a and b are incompatible). As explained in [50], this is due to the fact that even though the author in [55] might have a preference for one solution or the other, in the excerpt being analyzed, he is merely defending the solutions represented in a from attacks, and making no argument which attacks the solutions represented in b.*

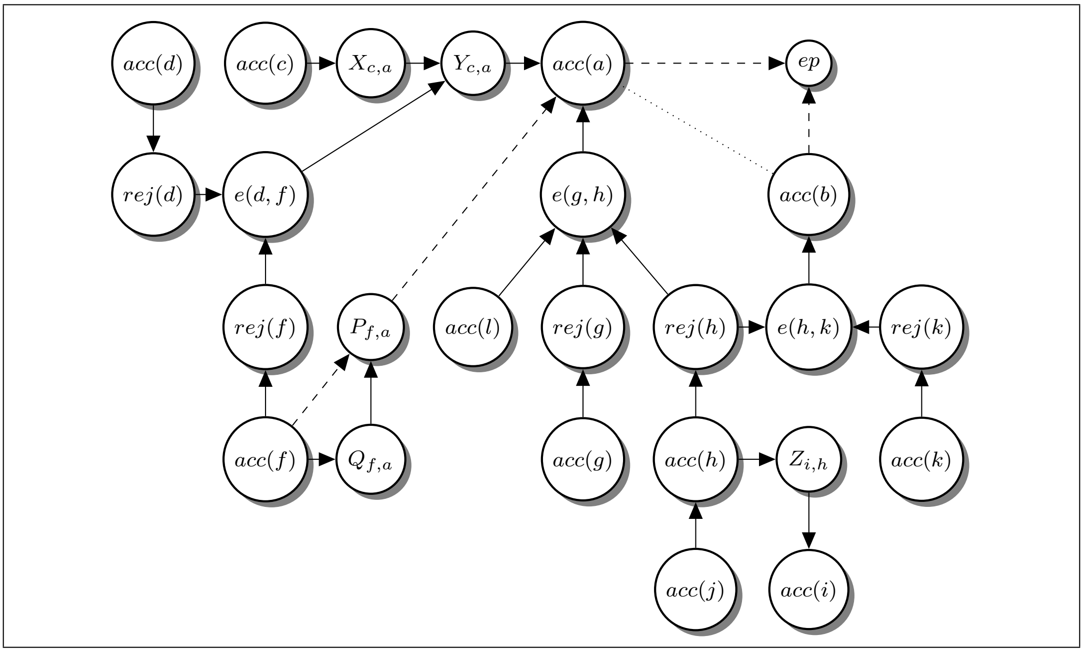


Figure 11: Explanatory Argumentation Framework corresponding to the *EEAF* of Example 13

- Among the existing applications of higher-order frameworks, [21; 22] proposes an application to the domain of deductive mathematical proofs. This application has been implemented under the form of a tool, named CLEAR (Constructing and evaLuating dEductive mAthematical pRoofs), designed for students that take mathematics and logics courses. It allows students to build deductive proofs collaboratively using a structured argumentative debate and allows teachers to evaluate these proofs. A light structure is used for modelling the logical arguments: a pair (Δ, α) such that α is a conclusion safely obtained from Δ. Then, the classical notions of rebuttal and undercutting can be used in order to define the attacks (see [12, Chapter 9]). The higher-order framework used is the one presented in [28] (see Section 5.2.1), with reversed supports, since the meaning of the support relation used in the tool is the deductive one, whereas [28] uses the necessary one. Note that the duality between deductive and necessary support can be used in CLEAR since the support relation cannot target another interaction (Definition 12 in [22]).

  So the support relation stands for deduction, and the attack (defeat) relation stands for conflict, this last one being a higher-order relation (targets can be arguments or other relations). Moreover, the tool gives the possibility to aggregate two or more arguments in order to create a "collective support" to another argument.

The following example illustrates the kind of argumentation framework we can build with CLEAR.

Example 14 (from [22])**.** *Consider the following theorem that must be proven: "Let ABC be a right triangle in A. Consider that $AB = 4$ and $BC = 5$ and prove that $AC = 3$."*

The following propositions are available in order to build this proof:

ABC *is a right triangle in* A
$BC^2 = AC^2 + AB^2$
$AB^2 = 16$
$AC^2 = 9$
$AB = 4$

$AB^2 = BC^2 + AC^2$
$AC^2 = BC^2 - AB^2$
$BC^2 = 25$
$BC = 5$
$AC = 3$

The (simplified) debate between students is shown below. On the one hand, some informal arguments are given. On the other hand, the debate reflects the exchanges between students about the building of the proof; hence, some arguments, deductions or attacks are sometimes "surprising":

a_1**:** *If ABC is a right triangle in A then $BC^2 = AC^2 + AB^2$.*

a_2**:** *No, if ABC is a right triangle in A then $AB^2 = BC^2 + AC^2$ (and so a_2 attacks a_1).*

a_{inf1}**:** *(informal argument) Argument a_2 is false (and so a_{inf1} attacks a_2).*

a_{inf3}**:** *(informal argument) a_2 cannot attack a_1 since a_1 is correct. This relation must be removed. So a_{inf3} attacks the attack from a_2 to a_1.*

a_3**:** *(deduced from a_1) If $BC^2 = AC^2 + AB^2$ then $AC^2 = BC^2 - AB^2$.*

a_{inf2}**:** *(informal argument) Applying the Pythagorean Theorem, $BC^2 = AC^2 + AB^2$ (that gives another way for deducing a_3).*

a_{inf5}**:** *(informal argument) a_{inf2} is redundant with a_1 (and so a_{inf5} attacks a_{inf2}).*

a_4**:** *If $AC^2 = BC^2 - AB^2$ and $AB^2 = 16$ and $BC^2 = 25$ then $AC^2 = 9$. Moreover a_4 can be deduced from a_2.*

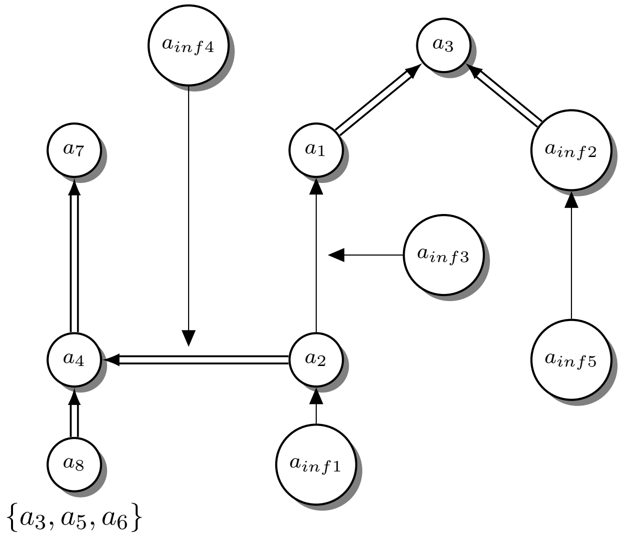
a_{inf4}**:** *(informal argument) No, a_2 does not allow the deduction of a_4. This relation must be removed. So a_{inf4} attacks the support from a_2 to a_4.*

a_5**:** *if $BC = 5$ then $BC^2 = 25$.*

- a_6: *if $AB = 4$ then $AB^2 = 16$. And so the aggregation of a_3, a_5 and a_6 allows the deduction of a_4.*
- a_7: *(deduced from a_4) If $AC^2 = 9$ then $AC = 3$.*

An additional argument a_8 can be created in order to represent the aggregation of a_3, a_5 and a_6 that must be used together for deducing a_4 (joint support).

And so the corresponding higher-order argumentation framework with deductive supports can be represented as follows:



Note that the link between a_3 and a_8 does not really appear in the graph (but it is recorded in the tool and can be used in the final steps). In the same way, a_5 and a_6 are not depicted in the figure; they are isolated and their only role is to be involved in the creation of a_8. The corresponding ASAF can be represented with the same graph in which the direction of the support edges has been reversed. And then, using the ASAF semantics, the preferred extensions can be computed. Of course, we are interested in the extensions that contain argument a_7 that corresponds to the conclusion we try to prove. These extensions will be used for building "proof graphs" for a_7, and then these graphs will be presented to the teachers for evaluation and discussion.

Here, a proof graph for a_7 is: from a_1 one can deduce a_3; from a_3, a_5 and a_6 one can deduce a_4; from a_4 one can deduce a_7.

7 Analysis

In this chapter, at least ten different approaches are presented and, clearly, a synthetic point of view is needed. This is the aim of this section.

First, Table 1 describes the kind of interaction that is taken into account by the different approaches. As we can see in this table, many different possibilities exist regarding the type of interaction (attacks and/or supports, with at least 4 “types” of support), regarding the “order” of these interactions (no interaction, first-order, second-order, and at any level), and regarding the form of their source or their target (one element or a set of elements). No approach is general enough to take into account all these possibilities. But in fact, the question arises whether it would be interesting to have such an approach.

Then, for each approach (except from [13], which does not propose semantics), Table 2 gives the method followed for defining semantics, whereas Table 3 presents the type of results produced by these semantics. The interesting point here is the fact that almost all approaches have developed semantics in a direct way, some of them also proposing a transformation of their framework into a Dung-like meta-argumentation framework. This transformation facilitates the understanding of the framework and the use of the existing solvers in computational issues. So, for the approaches that do not propose this transformation, it could perhaps be interesting to identify the associated meta-argumentation framework. Concerning the semantics results, four alternatives exist and our personal opinion is that, since interactions can be attacked or supported, they should also appear as outputs of the semantics.

Table 4 synthesizes the links between all approaches answering to the question: Who extends who? Clearly, all the proposed frameworks extend Dung’s framework; then, they differ, either on the type of support that is taken into account, or on the way in which they take into account the higher-order attacks. So, three distinct families appear:

- the first one is issued from the seminal work [13] and the EAF,
- the second family follows Baroni *et al*’s work around the $AFRA$,
- and the last family follows Cayrol *et al*’s work with the notion of RAF.

Note that the first two families also follow ideas of the meta-argumentation approach.

And finally, Table 5 lists the known links between these approaches in terms of their semantics (and the associated properties). Considering two approaches i and j, several links can exist:

Approach	attack order	support order (0 means no support)	support type: deductive (D) necessary (N), evidential (E), undefined (U)	collective attack (source is a set)	collective support (source is a set)	disjunctive interaction (target is a set)	source can be an interaction
[13]	any	any	U	no	no	no	no
EAF	2	0	NC	no	no	no	no
[62]	any	0	NC	yes	yes	yes	yes
[57]	any	0	NC	yes	yes	yes	yes
$AFRA$	any	0	NC	no	no	no	no
RAF	any	0	NC	no	no	no	no
[16]	2	1	D	no	no	no	yes
$ASAF$	any	any	N	no	no	no	no
$RAFN$	any	any	N	no	yes	no	no
$REBAF$	any	any	E	yes	yes	no	no

Table 1: Interactions taken into account (NC means "Not Concerned")

- First i is a conservative generalization of j (denoted by $i \rhd j$). That corresponds to the fact that, when the approach i is used on an argumentation framework corresponding to the approach j, all results are strictly identical (for all notions involved in semantics: conflict-freeness, acceptability, admissibility, ...). So this link appears only when i is an extension of j. However, it can be the case that i extends j but i is not a conservative generalization of j (see *e.g.* the relationship between $AFRA$ and AF).

- Other links can appear between two approaches, allowing to relate frameworks of the same kind (*e.g.* two approaches using a set of arguments and a set of higher-order attacks) or frameworks with a different structure (*e.g.* one

Approach	Directly	After transformation into an AF
[13]	No semantics definition	
EAF	yes	yes (meta-arg+flattening)
[62]	yes	no
[57]	yes	yes (meta-arg+flattening)
$AFRA$	yes	yes (meta-arg+flattening)
RAF	yes	no
[16]	no	yes (meta-arg+flattening)
$ASAF$	yes	yes (meta-arg+flattening)
$RAFN$	yes	no
$REBAF$	yes	no

Table 2: Semantics definition

Approach	Set of			
	arguments	labellings arg+int	arguments + interactions	arguments + meta-arg
[13]	No semantics definition			
EAF	yes	yes	no	yes
[62]	yes	no	no	no
[57]	no	yes	no	yes
$AFRA$	no	no	yes	yes
RAF	no	no	yes	no
[16]	no	no	no	yes
$ASAF$	no	yes	yes	yes
$RAFN$	no	no	yes	no
$REBAF$	no	no	yes	no

Table 3: Semantics output, defined in terms of a set of: arguments, labellings of arguments, arguments + interactions or arguments + meta-arguments

approach that considers a set of arguments and higher-order attacks and supports, and another considering a set of arguments and first-order attacks and supports). We identify four cases:

- either there is a complete one-to-one correspondence between the approaches i and j (denoted by $i = j$): i and j give exactly the same results (for all notions involved in semantics: conflict-freeness, accept-

	AF	*BAF* deductive (D), necessary (N), evidential (E), undefined interpretation (U)	[13]	*EAF*	[62]	[57]	*AFRA*	*RAF*	[16]	*ASAF*	*RAFN*
[13]	✓	✓(U)									
EAF	✓										
[62]	✓			✓							
[57]	✓			✓							
AFRA	✓										
RAF	✓										
[16]	✓	✓(D)		✓							
ASAF	✓	✓(N)					✓				
RAFN	✓	✓(N)						✓			
REBAF	✓	✓(E)						✓			

Let i (resp. j) be the approach given on the line (resp. column). $i\checkmark j$ means that i is an extended argumentation framework issued from j.

Table 4: Who extends who?

ability, admissibility, ...); in this case, i and j are applied on the same data, whereas in the case of $\rhd$, the link exists only if i is applied on the more simple data corresponding to the scope of j;

- or there is a partial one-to-one correspondence between the approaches i and j (denoted by $i \approx j$): i and j give the same results (for all notions involved in semantics: conflict-freeness, acceptability, admissibility, ...) if we consider *some constraints either on i, or on j*;
- or there is a partial one-to-one correspondence between the approaches i and j *but only at the semantics level* (denoted by $i \sim j$): i and j give the same results if we only consider semantics such as complete, grounded, preferred or stable (and so the results differ when we consider some other notions as, for instance, conflict-freeness or acceptability); sometimes,

some constraints must also to be considered (for instance, if i is applied on the more simple data corresponding to the scope of j);

- or there is no one-to-one correspondence between the approaches i and j (denoted by $i \neq j$): i and j do not give the same results for some semantics, even if some constraints are given on i or j.

Of course, there is also the trivial case that no link can exist between i and j only because they correspond to frameworks of different nature (for instance i takes into account evidential supports, whereas j takes into account necessary supports). This case will be denoted by NC ("Not Concerned") in Table 5. Here the main point is that, even if some links have already been established, a lot of work remains to be done in order to completely compare all these approaches. Note also that no approach is strictly equivalent to another one (there is no i, j such that $i = j$). That means that each approach has its own peculiarities and meets special needs. That also explains why it is difficult to unify these approaches.

8 Conclusion

It is now time to conclude this long chapter. Its aim was to propose a state of the art on higher-order abstract bipolar argumentation frameworks, *i.e.* abstract argumentation frameworks that allow interactions targeting other interactions, these interactions being either attacks or supports. This survey is as exhaustive as possible, but, since this topic is currently a very hot topic, it is possible, even probable, that some works are missing.

Nevertheless, considering all the works presented here, we can at least conclude on some points:

- the study of higher-order interactions (bipolar or not) in abstract argumentation is clearly an important topic since it allows an enriched representation of knowledge;
- many distinct approaches addressing this topic were proposed since the seminal work published in [13];
- these approaches can be partitioned into a smaller number of "families";
- even if there exist some links between these families, it is not so simple to unify them into a single general approach because they address different needs and use different methods, or even adopt different interpretations for the notion of support;

	AF	*BAF* deductive (D), necessary (N), evidential (E)	[13]	*EAF*	[62]	[57]	*AFRA*	*RAF*	[16]	*ASAF*	*RAFN*
[13]	No semantics										
EAF	$\triangleright$	*NC*	No semantics								
[62]	$\triangleright$	*NC*		$\neq$							
[57]	$\triangleright$	*NC*		$\neq$	$\neq$						
AFRA	$\sim$	*NC*		$\neq$	$\neq$	$\neq$					
RAF	$\triangleright$	*NC*					$\sim$				
[16]	$\triangleright$	$\triangleright$ (D)		$\triangleright$		$\neq$	$\neq$	$\neq$			
ASAF	$\sim$	$\sim$ (N)					$\triangleright$	$\sim$	*NC*		
RAFN	$\triangleright$	$\triangleright$ (N)					$\sim$	$\triangleright$	*NC*	$\neq$	
REBAF	$\triangleright$	$\approx$ (E)					$\sim$	$\triangleright$	*NC*	*NC*	*NC*

Let i (resp. j) be the approach given on the line (resp. column).

- $i \triangleright j$ means that i is a conservative generalization of j (i gives exactly the same results that j when we consider the restriction of i to j).
- $i \approx j$ means that there exists a one-to-one correspondence, but with some constraints (depending of the case).
- $i \sim j$ means that there exists a one-to-one correspondence, but only at the semantics level (sometimes with constraints).
- $i \neq j$ means that no one-to-one correspondence can exist between i and j.
- *NC* means "Not Concerned".

Table 5: Links between approaches in terms of semantics and associated properties

- and so, a lot of work remains to be done in this topic: for an eventual unification, but also for computational issues (study of complexity, algorithms);
- in order to boost this last point, the introduction of some dedicated tracks in the ICCMA competition could be of great help.

Acknowledgements

This work was partially supported by Universidad Nacional del Sur under grant 24/N046.

References

[1] Gianvincenzo Alfano, Andrea Cohen, Sebastian Gottifredi, Sergio Greco, Francesco Parisi, and Guillermo R. Simari. Dynamics in abstract argumentation frameworks with recursive attack and support relations. In *Proc. of ECAI 2020 - 24th European Conference on Artificial Intelligence*, 2020.

[2] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. Efficient computation of extensions for dynamic abstract argumentation frameworks: An incremental approach. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 49–55, 2017.

[3] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. A meta-argumentation approach for the efficient computation of stable and preferred extensions in dynamic bipolar argumentation frameworks. *Intelligenza Artificiale*, 12(2):193–211, 2018.

[4] Leila Amgoud and Claudette Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–216, 2002.

[5] Leila Amgoud, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *10th International Workshop on Non-Monotonic Reasoning (NMR 2004), Whistler, Canada, June 6-8, 2004, Proceedings*, pages 1–9, 2004.

[6] Leila Amgoud, Nicolas Maudet, and Simon Parsons. Modelling dialogues using argumentation. In *Proc. of ICMAS*, pages 31–38, 2000.

[7] Ryuta Arisaka, Francesco Santini, and Stefano Bistarelli. Block argumentation. In Matteo Baldoni, Mehdi Dastani, Beishui Liao, Yuko Sakurai, and Rym Zalila-Wenkstern, editors, *PRIMA 2019: Principles and Practice of Multi-Agent Systems - 22nd International Conference, Turin, Italy, October 28-31, 2019, Proceedings*, volume 11873 of *Lecture Notes in Computer Science*, pages 618–626. Springer, 2019.

[8] Ryuta Arisaka and Ken Satoh. Voluntary manslaughter? a case study with meta-argumentation with supports. In *New Frontiers in Artificial Intelligence. JSAI-isAI 2016. LNCS 10247*, pages 241–252. Springer, 2017.

[9] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 4, pages 159–236. College Publications, 2018.

[10] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. Encompassing attacks to attacks in abstract argumentation frameworks. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 10th European Conference, ECSQARU 2009, Verona, Italy, July 1-3, 2009. Proceedings*, pages 83–94, 2009.

[11] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52:19–37, 2011.

[12] Pietro Baroni, Dov Gabbay, and Massimiliano Giacomin. *Handbook of Formal Argumentation.* College Publications, 2018.

[13] Howard Barringer, Dov Gabbay, and John Woods. Temporal dynamics of support and attack networks : From argumentation to zoology. In Dieter Hutter and Werner Stephan, editors, *Mechanizing Mathematical Reasoning, Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday. LNAI 2605*, pages 59–98. Springer Verlag, 2005.

[14] Trevor Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[15] Guido Boella, Dov Gabbay, Leendert van der Torre, and Serena Villata. Meta-argumentation modelling I: methodology and techniques. *Studia Logica*, 93(2-3):297–355, 2009.

[16] Guido Boella, Dov Gabbay, Leendert van der Torre, and Serena Villata. Support in abstract argumentation. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010.*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 111–122. IOS Press, 2010.

[17] Guido Boella, Leendert W. N. van der Torre, and Serena Villata. Attack relations among dynamic coalitions. In *Proceedings of the 20th Belgian-Dutch Artificial Intelligence Conference, BNAIC 2008*, pages 25–32. Universiteit Twente, Enschede, 2008.

[18] Guido Boella, Leendert W. N. van der Torre, and Serena Villata. Social viewpoints for arguing about coalitions. In The Duy Bui, Tuong Vinh Ho, and Quang-Thuy Ha, editors, *Intelligent Agents and Multi-Agent Systems, 11th Pacific Rim International Conference on Multi-Agents, PRIMA 2008, Hanoi, Vietnam, December 15-16, 2008. Proceedings*, volume 5357 of *Lecture Notes in Computer Science*, pages 66–77. Springer, 2008.

[19] Guido Boella, Leendert W. N. van der Torre, and Serena Villata. On the acceptability of meta-arguments. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2009, Milan, Italy, 15-18 September 2009*, pages 259–262. IEEE Computer Society, 2009.

[20] Imane Boudhar, Farid Nouioua, and Vincent Risch. Handling preferences in argu-

mentation frameworks with necessities. In *ICAART 2012 - Proceedings of the 4th International Conference on Agents and Artificial Intelligence, Volume 1 - Artificial Intelligence, Vilamoura, Algarve, Portugal, 6-8 February, 2012*, pages 340–345, 2012.

[21] Nadira Boudjani, Abdelkader Gouaïch, and Souhila Kaci. CLEAR: argumentation frameworks for constructing and evaluating deductive mathematical proofs. In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *Computational Models of Argument - Proceedings of COMMA 2018, Warsaw, Poland, 12-14 September 2018*, volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 281–288. IOS Press, 2018.

[22] Nadira Boudjani, Abdelkader Gouaich, and Souhila Kaci. Argumentation frameworks for constructing, representing and evaluating deductive mathematical proofs. Technical Report lirmm-02458939, LIRMM, 2020.

[23] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract dialectical frameworks. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 5, pages 237–285. College Publications, 2018.

[24] Elena Cabrio and Serena Villata. Detecting bipolar semantic relations among natural language arguments with textual entailment: a study. In *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora, JSSP 2013, Trento, Italy, November 20-22, 2013*, pages 24–32, 2013.

[25] Martin Caminada. On the issue of reinstatement in argumentation. In *Proc. of the European Conference of Logics in Artificial Intelligence (JELIA)*, pages 111–123, 2006.

[26] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5–6):286–310, 2007.

[27] Martin Caminada and Dov Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2-3):109–145, 2009.

[28] Claudette Cayrol, Andrea Cohen, and Marie-Christine Lagasquie-Schiex. Towards a new framework for recursive interactions in abstract bipolar argumentation. In Pietro Baroni, editor, *Proceedings of International Conference on Computational Models of Argument (COMMA)*, pages 191–198. IOS Press, 2016.

[29] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Valid attacks in argumentation frameworks with recursive attacks. In *Proc. of Thirteenth International Symposium on Commonsense Reasoning*, volume 2052. CEUR Workshop Proceedings, 2017.

[30] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Argumentation frameworks with recursive attacks and evidence-based support. In F. Ferrarotti and S. Woltran, editors, *Proc. of Tenth International Symposium on Foundations of Information and Knowledge Systems (FoIKS)*, volume LNCS 10833, pages 150–169. Springer-Verlag, 2018.

[31] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Structure-based Semantics of Argumentation Frameworks with Higher-order Attacks and Supports. In Carlos I. Chesñevar et al, editor, *Argumentation-*

based Proofs of Endearment. Essays in Honor of Guillermo R. Simari on the Occasion of his 70 th Birthday, volume 37 of *Tributes*, pages 43–72. College Publications, http://www.collegepublications.co.uk/, 2018.

[32] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Structure-Based Semantics of Argumentation Frameworks with Higher-Order Attacks and Supports (short paper). In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *International Conference on Computational Models of Argument (COMMA)*, pages 29–36. IOS Press, septembre 2018.

[33] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Valid attacks in argumentation frameworks with recursive attacks. *Annals of Mathematics and Artificial Intelligence (Special Issue: Commonsense 2017)*, 2020.

[34] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 8th European Conference, ECSQARU 2005, Barcelona, Spain, July 6-8, 2005, Proceedings*, pages 378–389, 2005.

[35] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Coalitions of arguments in bipolar argumentation frameworks. In *7th International Workshop on Computational Models of Natural Argument*, pages 14–20, 2007.

[36] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. The Grafix website, 2008. http://www.irit.fr/grafix.

[37] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolar abstract argumentation systems. In Iyad Rahwan and Guillermo Simari, editors, *Argumentation in Artificial Intelligence*, pages 65–84. Springer, 2009.

[38] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Coalitions of arguments: A tool for handling bipolar argumentation frameworks. *International Journal of Intelligent Systems*, 25(1):83–109, 2010.

[39] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolarity in argumentation graphs: Towards a better understanding. *Int. J. Approx. Reasoning*, 54(7):876–899, 2013.

[40] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Logical encoding of argumentation frameworks with higher-order attacks. In *Proc. of the 30th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2018.

[41] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Logical Encoding of Argumentation Frameworks with Higher-order Attacks and Evidential Supports. *International Journal on Artificial Intelligence Tools*, 29(03n04):2060003, June 2020.

[42] Andrea Cohen, Alejandro Javier García, and Guillermo Ricardo Simari. Backing and undercutting in defeasible logic programming. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 11th European Conference, ECSQARU 2011, Belfast, UK, June 29-July 1, 2011. Proceedings*, pages 50–61, 2011.

[43] Andrea Cohen, Alejandro Javier García, and Guillermo Ricardo Simari. Backing and undercutting in abstract argumentation frameworks. In *Foundations of Information*

and Knowledge Systems - 7th International Symposium, FoIKS 2012, Kiel, Germany, March 5-9, 2012. Proceedings, pages 107–123, 2012.

[44] Andrea Cohen, Sebastian Gottifredi, Alejandro J García, and Guillermo R Simari. An approach to abstract argumentation with recursive attack and support. *Journal of Applied Logic*, 13(4):509–533, 2015.

[45] Andrea Cohen, Sebastian Gottifredi, Alejandro J. García, and Guillermo R. Simari. On the acceptability semantics of argumentation frameworks with recursive attack and support. In Pietro Baroni, editor, *Proceedings of International Conference on Computational Models of Argument (COMMA)*, pages 231–242. IOS Press, 2016.

[46] Andrea Cohen, Sebastian Gottifredi, Alejandro Javier García, and Guillermo Ricardo Simari. A survey of different approaches to support in argumentation systems. *Knowledge Eng. Review*, 29(5):513–550, 2014.

[47] Andrea Cohen, Simon Parsons, Elizabeth I. Sklar, and Peter McBurney. A characterization of types of support between structured arguments and their relationship with support in abstract argumentation. *Int. J. Approx. Reason.*, 94:76–104, 2018.

[48] Kristijonas Cyras, Xiuyi Fan, Claudia Schulz, and Francesca Toni. Assumption-based argumentation: Disputes, explanations, preferences. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 7, pages 365–408. College Publications, 2018.

[49] Kristijonas Cyras, Claudia Schulz, and Francesca Toni. Capturing bipolar argumentation in non-flat assumption-based argumentation. In *PRIMA 2017: Principles and Practice of Multi-Agent Systems - 20th International Conference, Nice, France, October 30 - November 3, 2017, Proceedings*, pages 386–402, 2017.

[50] Jérémie Dauphin and Marcos Cramer. Extended explanatory argumentation frameworks. In *Theory and Applications of Formal Argumentation - 4th International Workshop, TAFA 2017, Melbourne, VIC, Australia, August 19-20, 2017, Revised Selected Papers*, pages 86–101, 2017.

[51] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

[52] Phan Minh Dung and Phan Minh Thang. A unified framework for representation and development of dialectical proof procedures in argumentation. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 746–751, 2009.

[53] Wolfgang Dvorak, Sarah A. Gaggl, Thomas Linsbichler, and Johannes P. Wallner. Reduction-based approaches to implement Modgil's extended argumentation frameworks. In Miroslaw Truszczynski Thomas Eiter, Hannes Strass and Stefan Woltran, editors, *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation. Essays Dedicated to Gerhard Brewka on the Occasion of His 60th Birthday (LNCS, volume 9060)*, pages 249–264. Springer, 2015.

[54] Uwe Egly, Sarah Alice Gaggl, and Stefan Woltran. Answer-set programming encodings for argumentation frameworks. *Argument & Computation*, 1(2):147–177, 2010.

[55] Hartry Field. *Saving Truth from Paradoxes.* Oxford University PRess, 2008.

[56] Dov Gabbay. Fibring argumentation frames. *Studia Logica*, 93:231–295, 2009.

[57] Dov Gabbay. Semantics for higher level attacks in extended argumentation frames. *Studia Logica*, 93:357–381, 2009.

[58] Dov Gabbay and Michael Gabbay. The attack as intuitionistic negation. *Logic Journal of the IGPL*, 24(5):807–837, 2016.

[59] Sarah Alice Gaggl, Thomas Linsbichler, Marco Maratea, and Stefan Woltran. Design and results of the second international competition on computational models of argumentation. *Artif. Intell.*, 279, 2020.

[60] Alejandro Javier García and Guillermo Ricardo Simari. Argumentation based on logic programming. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 8, pages 409–435. College Publications, 2018.

[61] Sebastian Gottifredi, Andrea Cohen, Alejandro Javier García, and Guillermo Ricardo Simari. Characterizing acceptability semantics of argumentation frameworks with recursive attack and support relations. *Artificial Intelligence*, 262:336–368, 2018.

[62] Do Duc Hanh, Phan Minh Dung, Nguyen Duy Hung, and Phan Minh Thang. Inductive defense for sceptical semantics of extended argumentation. *Journal of Logic and Computation*, 21(2):307–349, 2011.

[63] Nikos Karacapilidis and Dimitris Papadias. Computer supported argumentation and collaborative decision making: the hermes system. *Information systems*, 26(4):259–277, 2001.

[64] Tastuki Kawasaki and Sosuke Moriguchiand Kazuko Takahashi. Transformation from PROLEG to a bipolar argumentation framework. In *2nd International Workshop on Systems and Algorithms for Formal Argumentation (SAFA 2018), Warsaw, Poland, September*, pages 36–47, 2018.

[65] Hengfei Li and Jiachao Wu. Semantics of extended argumentation frameworks defined by renovation sets. In *PRIMA 2019: Principles and Practice of Multi-Agent Systems - 22nd International Conference, Turin, Italy, October 28-31, 2019, Proceedings*, pages 532–540, 2019.

[66] Victor W. Marek and Mirosław Truszczyński. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm – A 25-Year Perspective*, pages 375–398. Springer, 1999.

[67] Sanjay Modgil. An abstract theory of argumentation that accommodates defeasible reasoning about preferences. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 9th European Conference, ECSQARU 2007, Hammamet, Tunisia, October 31 - November 2, 2007, Proceedings*, pages 648–659, 2007.

[68] Sanjay Modgil. Labellings and games for extended argumentation frameworks. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 873–878, 2009.

[69] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artificial*

Intelligence, 173:901–934, 2009.

[70] Sanjay Modgil and Trevor Bench-Capon. Integrating object and meta-level value based argumentation. In *Proc of International Conference on Computational models of arguments (COMMA)*, pages 240–251, 2008.

[71] Sanjay Modgil and Trevor J. M. Bench-Capon. Metalevel argumentation. *Journal of Logic and Computation*, 21(6):959–1003, 2011.

[72] Sanjay Modgil and Henry Prakken. Reasoning about preferences in structured extended argumentation frameworks. In *Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010.*, pages 347–358, 2010.

[73] Sanjay Modgil and Henry Prakken. Abstract rule-based argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 6, pages 287–364. College Publications, 2018.

[74] Søren Holbech Nielsen and Simon Parsons. A generalization of dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Argumentation in Multi-Agent Systems, Third International Workshop, ArgMAS 2006, Hakodate, Japan, May 8, 2006, Revised Selected and Invited Papers*, pages 54–73, 2006.

[75] Ilkka Niemelä. Logic programming with stable model semantics as a constraint programming paradigm. *Ann. Math. Artif. Intell.*, 25(3–4):241–273, 1999.

[76] Samer Nofal, Katie Atkinson, and Paul E. Dunne. Algorithms for argumentation semantics: Labeling attacks as a generalization of labeling arguments. *Journal of Artificial Intelligence Research*, 49:635–668, 2014.

[77] Farid Nouioua. AFs with necessities: Further semantics and labelling characterization. In *Scalable Uncertainty Management - 7th International Conference, SUM 2013, Washington, DC, USA, September 16-18, 2013. Proceedings*, pages 120–133, 2013.

[78] Farid Nouioua and Vincent Risch. Bipolar argumentation frameworks with specialized supports. In *22nd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2010, Arras, France, 27-29 October 2010 - Volume 1*, pages 215–218, 2010.

[79] Farid Nouioua and Vincent Risch. Argumentation frameworks with necessities. In *Scalable Uncertainty Management - 5th International Conference, SUM 2011, Dayton, OH, USA, October 10-13, 2011. Proceedings*, pages 163–176, 2011.

[80] Nir Oren and Timothy J. Norman. Semantics for evidence-based argumentation. In P. Besnard, S. Doutre, and A. Hunter, editors, *Proc. of COMMA*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 276–284. IOS Press, 2008.

[81] Sylwia Polberg. Understanding the abstract dialectical framework (preliminary report). *CoRR*, abs/1607.00819, 2016.

[82] Sylwia Polberg. Intertranslatability of abstract argumentation frameworks. Technical report, Technical Report DBAI-TR-2017–104, Institute for Information Systems, Technical University of Vienna, 2017.

[83] Sylwia Polberg and Anthony Hunter. Empirical evaluation of abstract argumentation:

Supporting the need for bipolar and probabilistic approaches. *Int. J. Approx. Reason.*, 93:487–543, 2018.

[84] Sylwia Polberg and Nir Oren. Revisiting support in abstract argumentation systems. In *Computational Models of Argument - Proceedings of COMMA 2014, Atholl Palace Hotel, Scottish Highlands, UK, September 9-12, 2014*, pages 369–376, 2014.

[85] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument & Computation*, 1(2):93–124, 2010.

[86] Henry Prakken. On support relations in abstract argumentation as abstractions of inferential relations. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, pages 735–740, 2014.

[87] Henry Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 4, pages 218–319. Kluwer Academic Pub., 2002.

[88] Iyad Rahwan and Guillermo R. Simari, editors. *Argumentation in Artificial Intelligence.* Springer, 2009.

[89] Ken Satoh, Kento Asai, Takamune Kogawa, Masahiro Kubota, Megumi Nakamura, Yoshiaki Nishigai, Kei Shirakawa, and Chiaki Takano. PROLEG: an implementation of the presupposed ultimate fact theory of japanese civil code by PROLOG technology. In *New Frontiers in Artificial Intelligence - JSAI-isAI 2010 Workshops, LENLS, JURISIN, AMBN, ISS, Tokyo, Japan, November 18-19, 2010, Revised Selected Papers*, pages 153–164, 2010.

[90] Dunja Seselja and Christian Straßer. Abstract argumentation and explanation applied to scientific debates. *Synthese*, 190(12):2195–2217, 2013.

[91] Phan Minh Thang, Phan Minh Dung, and Nguyen Duy Hung. Toward a common framework for dialectical proof procedure in abstract argumentation. *Journal of Logic and Computation*, 19(6):1071–1109, 2009.

[92] Matthias Thimm and Serena Villata. The first international competition on computational models of argumentation: Results and analysis. *Artif. Intell.*, 252:267–294, 2017.

[93] Stephen E. Toulmin. *The Uses of Argument.* Cambridge University Press, 1958.

[94] University of Le Havre. GraphStream: A Dynamic Graph Library. http://graphstream-project.org/, 2011.

[95] Bart Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation*, 13(3):319–346, 2003.

[96] Serena Villata, Guido Boella, Dov Gabbay, and Leendert van der Torre. Modelling defeasible and prioritized support in bipolar argumentation. *Annals of Mathematics and Artificial Intelligence*, 66(1-4):163–197, 2012.

[97] Serena Villata, Guido Boella, and Leendert W. N. van der Torre. Argumentation patterns. In Peter McBurney, Simon Parsons, and Iyad Rahwan, editors, *Proceeding of the Eighth International Workshop on Argumentation in Multi-Agent Systems, ArgMAS,*

Taipei, Taiwan, May 2011, pages 133–150, 2011.

Received May 2021

Joint Attacks and Accrual in Argumentation Frameworks

Antonis Bikakis
University College London (UCL), UK
a.bikakis@ucl.ac.uk

Andrea Cohen
Institute for Computer Science and Engineering, CONICET-UNS
Dept. of Computer Science and Engineering, Universidad Nacional del Sur
Bahía Blanca, Argentina
ac@cs.uns.edu.ar

Wolfgang Dvořák
Institute of Logic and Computation, TU Wien, Austria
dvorak@dbai.tuwien.ac.at

Giorgos Flouris
Foundation for Research and Technology - Hellas (FORTH), Greece
fgeo@ics.forth.gr

Simon Parsons
University of Lincoln, UK
sparsons@lincoln.ac.uk

Abstract

While modelling arguments, it is often useful to represent “joint attacks”, i.e., cases where multiple arguments jointly attack another (note that this is different from the case where multiple arguments attack another in isolation). Based on this remark, the notion of joint attacks has been proposed as a useful extension of classical Abstract Argumentation Frameworks, and has been shown to constitute a genuine extension in terms of expressive power. In this chapter, we review various works considering the notion of joint attacks from various perspectives, including abstract and structured frameworks. Moreover, we present results detailing the relation among frameworks with joint attacks and classical argumentation frameworks, computational aspects, and applications of joint attacks. Last but not least, we propose a roadmap for future research on the subject, identifying gaps in current research and important research directions.

Vol. 8 No. 6 2021
Journal of Applied Logics — IfCoLog Journal of Logics and their Applications

1 Introduction

As many have already pointed out, the work of Dung [40] is a cornerstone, arguably *the* cornerstone, of current work on computational argumentation. It was the work that introduced the notion of abstract argumentation and the idea that argumentation could be modelled just as a set of arguments and attacks between them, and it provided an initial set of semantics — complete, grounded, preferred and stable — for the evaluation of a set of arguments and attacks. As such, it is the work on which all subsequent work on abstract argumentation has been built. In addition, because many structured argumentation systems adopt the Dung semantics as a means of establishing which arguments are acceptable, these systems are also built upon [40].

Much of the appeal of [40] lies in its elegant simplicity. The approach relies on just two concepts — arguments and attacks — and yet these simple components can capture a complex range of types of reasoning, reflected in the large set of semantics that have been defined for abstract argumentation systems. However, this very simplicity means that abstract argumentation has limitations in terms of what it can represent. The limitations of representing arguments as atomic entities is widely recognised, and is addressed by work on structured argumentation[1]. However, there are also limitations in the way that [40] handles interactions between arguments. Attacks are binary, so that a given attack is from a single argument to a single argument. Attacks are also atomic in the sense that their impact is assessed independently of other attacks. To use the terminology of [6], an argument will be `out` as soon as it is attacked by a single `in` argument, regardless of any other attacks that may exist. The evaluation of an argument does not, even where arguments have different strengths, take account of whether there are multiple attacks on it. Where strengths are taken into account, it is, effectively, only the strongest attacker that matters.

These limitations, and in particular how they may be overcome, is the subject of this chapter. We are primarily interested in the extension of the [40] model of abstract argumentation to allow non-binary, or "joint" attacks. In particular, we consider the "sets of attacking arguments" (SETAF) approach first suggested in [84]. In this approach it is possible to model situations in which two or more arguments jointly attack a single argument, and we explore this approach in depth. This focus also leads us to consider bipolar argumentation frameworks, where joint attacks are a key element, and these frameworks, in turn, lead us to consider joint supports between arguments. We also briefly discuss how joint attacks might be modelled in

[1] In some systems of structured argumentation, ASPIC+ [81] for example, it is possible to cleanly "lift" a set of abstract arguments from a set of structured arguments in such a way that Dung-style semantics can be applied. In other systems, DeLP [65] for example, this is not possible.

structured argumentation, and touch on the rather neglected topic of *accrual*, which models situations in which the strength of sets of independent attacking arguments is an aggregate of the strengths of the arguments it contains.

The rest of this chapter is structured as follows. Section 2 motivates the study of joint attacks. Section 3 is perhaps the most central section of the chapter. It introduces the formal model of SETAFs, relates the model to classical abstract argumentation models, considers the computational aspects of SETAFs, and looks at alternative formulations for set-based attacks. Section 4 looks at the uses of joint attacks in bipolar argumentation frameworks, and considers the models of joint support that occur in those frameworks as well, while also discussing the use of joint attacks to model higher-order interactions. Section 5 then briefly covers the related topic of accrual, the combination of arguments for or against a given claim. Finally Section 6 looks at future lines of work on joint attacks, and Section 7 provides a brief summary and draws some conclusions.

2 Motivating the need for joint attacks

There are a number of possible motivations for work on joint attacks. One comes from a purely formal consideration of [40]. Dung [40] considers argumentation frameworks that take the form of a directed graph, with nodes being arguments and edges being attacks between arguments. It is natural to consider a generalisation of these frameworks to ones where the directed graph becomes a directed hypergraph. In its most general form, such a framework would have nodes that represent sets of arguments, and edges that represent attacks between sets of nodes[2]. What we study here is a less general representation in which nodes represent single arguments, and edges represents attacks where the attackers can be a set, but the attackee is constrained to be a singleton. Though less general than the representation just sketched, this is, as we discuss below, a genuine extension of the Dung argumentation framework.

This representation can also be motivated by considering knowledge that is most elegantly represented in a formalism that allows for joint attacks. For example, taken from [60], consider the following aspects of the UK laws governing marriage and civil

[2] [84] briefly considers explicitly representing the most general case of sets of arguments as both attacker and attackee, before settling on the SETAF formalism that we describe below. As [84] points out, SETAFs were originally devised during work that allowed arguments about Bayesian networks — work summarised in [83] — and not only is the SETAF approach able to capture attacks of sets of arguments on sets of arguments (by attacking each member of the attackee set separately), but it also mirrors the structure of a Bayesian network where conditional probability distributions capture multiple parents affecting a common child, but do not capture a single parent affecting multiple children.

partnerships[3] (as of early 2020). One is not allowed to enter into a marriage or civil partnership if

(a) you are under 16;

(b) you are closely related to your partner;

(c) you are not single; or

(d) you are under 18 and do not have permission to marry from your parents or guardians.

Much of this can be represented in a standard Dung argumentation framework, with an argument to represent the right to get married or enter a civil partnership (M), which is attacked by arguments that represent being under 16 ($A16$), being closely related (R), and not being single (NS). One might also represent case (d) with a single argument, but this single argument captures both being under 18 *and* not having permission — let's call this argument MWP (for minor without permission). That is fine on its own, but now consider adding additional information about the UK legal system into the framework, [60] again, this time on voting rights. In the UK you are allowed to vote[4], unless you are under 18, and the natural way to capture this is with an argument (V) representing the right to vote, which is attacked by an argument ($A18$) representing being under 18. How, then, do we capture the relationship between MWP, which incorporates the fact that the individual in question is under 18, and $A18$? We would argue that a natural and elegant way to do this is by replacing MWP by the argument NP, representing the fact that there is no parental permission, and having $A18$ and NP jointly attack M. The resulting SETAF is shown in Figure 1.

Just to make the point that this example of a joint attack is not contrived, Figure 1 contains some other arguments that are found in UK legislation and have a natural representation as a SETAF. (These all reflect the age of majority in the UK, which, as one might expect, crops up a lot in the law.) For example, consider the law around alcohol consumption[5]. In the UK, one is allowed to consume alcohol in public (Alc), unless one is under 16, or one is under 18 and not accompanied by an adult (NA), or one is under 18 and not having a meal (NM).

Of course, we are not claiming that using joint attacks is the *only* way to represent the above information. As we mentioned, it is possible to capture all of this in a standard abstract argumentation framework, using what are effectively compound

[3] https://www.gov.uk/marriages-civil-partnerships

[4] https://www.gov.uk/elections-in-the-uk

[5] https://www.gov.uk/alcohol-young-people-law

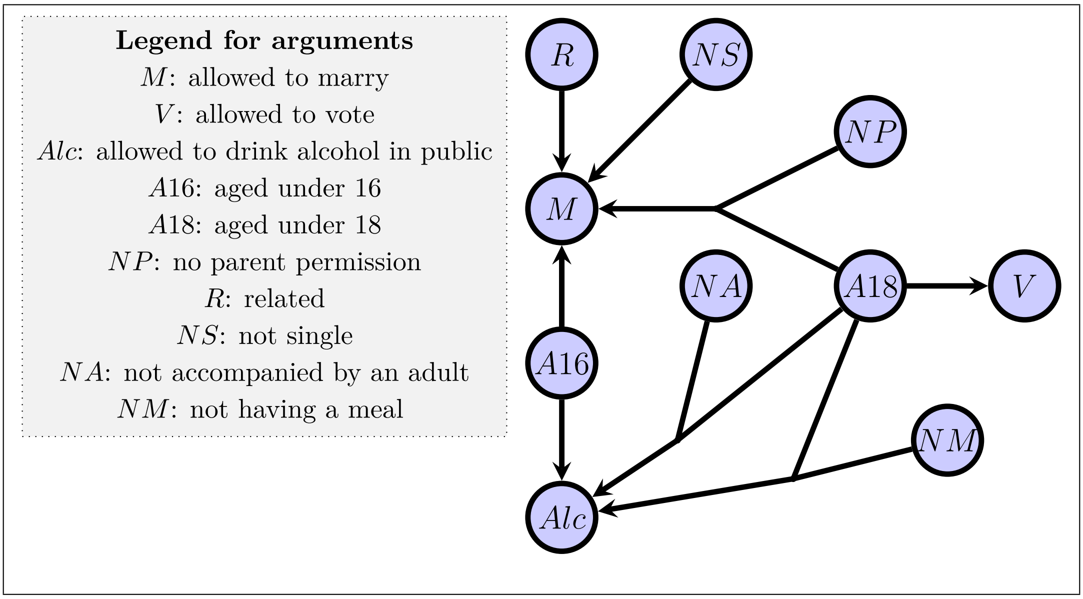


Figure 1: Example of a SETAF, encoding a part of UK legislation.

arguments such as "under 18 and not accompanied by an adult". Indeed, [60] shows that it is always possible to represent a SETAF as a standard abstract argumentation framework, albeit at the cost of a possibly substantial increase in the number of arguments. In addition to this potential cost, a cost both representational and computational, we echo the sentiment expressed in [84], that using standard frameworks rather than SETAFs in cases like that of Figure 1 tends to muddle the distinction between arguments and attacks which is the essence of the abstract argumentation approach.

3 Modelling joint attacks

In this section, we provide formal considerations associated with the use of collective attacks in argumentation frameworks. These are meant to provide the basic tools towards further formal results on the issue.

More specifically, in Section 3.1, we provide the basic formal definitions associated with SETAFs, as well as their semantics (provided both in terms of extensions, and in terms of labellings). Moreover, a series of formal results on extensions, labellings and their relations are presented, most of which are a direct adaptation of similar results from the standard AF setting.

Section 3.2 studies the relationship among AF and SETAF, and provides answers to the fundamental question of whether SETAFs constitute a more expressive tool than AFs for describing arguments and their relationships.

Section 3.3 provides various computational complexity results related to SETAFs, for different problems pertaining to different semantics. Moreover, algorithms and system implementations that address these problems are considered, including reduction-based approaches.

Further, in Section 3.4 we discuss various alternative models of abstract and structured argumentation accounting for collective attacks, i.e., attacks where a group (i.e., set) of arguments can act either as the attacker, or as the attackee.

3.1 Definitions and semantics

We start our description with the formal definition of SETAFs, including their semantics. In this subsection, we formally describe various types of semantics that have been proposed in the literature, as well as relevant results that should form the formal background and toolbox of anyone aiming to study SETAFs and their properties.

3.1.1 AFs and AF semantics: A brief reminder

An AF was defined in [40] as a pair $AF^{\mathcal{D}} = \langle Ar, att \rangle$ consisting of a (possibly infinite) set of arguments Ar and a binary attack relation att on this set. In principle, an AF is a directed graph, whose nodes correspond to arguments and whose edges correspond to attacks, which essentially represent the fact that a certain argument invalidates another. AFs are given semantics through *extensions*, which are sets of arguments (nodes) that are non-conflicting (i.e., they do not attack each other) and, as a group, "shield" themselves from attacks by other arguments (which are not in the extension). The exact formal meaning given to the term "shield" gives rise to a multitude of different semantics (complete, preferred, stable, etc.) which have been considered in the literature (e.g., see [6]).

Informally, a set of arguments $S \subseteq Ar$ is: (i) a *conflict-free* extension of $AF^{\mathcal{D}}$ iff it contains no arguments attacking each other; (ii) an *admissible* extension iff it is conflict-free and defends all its elements (i.e., for each argument $a \in Ar$ attacking an argument in S, there is an argument in S attacking a); (iii) a *complete* extension iff it is admissible and contains all the arguments it defends; (iv) a *grounded* extension iff it is minimal (w.r.t. set inclusion) among the complete extensions; (v) a *preferred* extension iff it is maximal among the complete extensions; (vi) a *stable* extension iff it is conflict-free and attacks all the arguments that it does not contain (i.e., all arguments in $Ar \setminus S$); (*vii*) a *naive* extension iff it is maximal among the conflict-free extensions; (*viii*) a *semi-stable* extension iff its union with the set of arguments it attacks is maximal among the complete extensions; (*ix*) an *eager* extension iff it is maximal among the complete extensions that are subsets of every semi-stable extension; (*x*) an *ideal* extension iff it is maximal among the complete extensions that are subsets of every preferred extension; and (*xi*) a *stage* extension iff its union with the set of arguments it attacks is maximal among the conflict-free extensions.

3.1.2 A formalism for joint attacks (SETAFs)

To formally represent the notion of joint attacks, Dung's definition for argumentation frameworks was extended in [84] for the case where an argument can be attacked by a set of other arguments:

Definition 3.1. *A* Framework with Sets of Attacking Arguments *(*SETAF *for short) is a pair* $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ *such that* Ar *is a set of arguments and* $\triangleright \subseteq (2^{Ar} \setminus \{\emptyset\}) \times Ar$ *is the* attack *relation.*

It is interesting to note the asymmetry in Definition 3.1: a group of arguments can be the attacker, but not the recipient of an attack. The reason for this asymmetry is justified in [84], where it is shown that allowing a set of arguments to be jointly attacked by another does not add to the expressiveness of the proposed model. Indeed, there can be two ways in which a many-to-many attack (say $\{a_1, \ldots, a_n\} \triangleright \{b_1, \ldots, b_m\}$) can be interpreted:

1. The first, called "collective defeat" in [104], states that no b_i is accepted whenever all of $a_1, \ldots, a_n$ are accepted. This case can be easily modelled in the setting of Definition 3.1 by creating m attacks of the form $\{a_1, \ldots, a_n\} \triangleright b_i$.
2. The second, called "indeterministic defeat" in [104], states that at least one of b_i should not be accepted whenever all of $a_1, \ldots, a_n$ are accepted. This case can also be modelled in the setting of Definition 3.1, by creating m attacks of the form $\{a_1, \ldots, a_n, b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_m\} \triangleright b_i$.

Nevertheless, for simplicity, the attack relationship $\triangleright$ of Definition 3.1 can be extended to apply among sets of arguments. Formally, we say that a set of arguments S attacks another set of arguments T (denoted by $S \blacktriangleright T$) iff there exist $U \subseteq S, a \in T$ such that $U \triangleright a$. Note that we used a different symbol for the extended relation, to avoid confusion. Importantly, $\blacktriangleright$ does not change the semantics of the attack and does not generalise it to attacks among sets of arguments; it is just a syntactic shorthand.

We will write $S \not\triangleright a$ when it is not the case that $S \triangleright a$, and $S \not\blacktriangleright T$ when it is not the case that $S \blacktriangleright T$. For singleton sets, we often write $S \blacktriangleright a$ to denote $S \blacktriangleright \{a\}$. We say that S *defends* an argument a from a set of arguments T that attacks a, iff $S \blacktriangleright T$.

An interesting note for SETAFs, is that certain attacks may be redundant. In particular, if we have that $S \triangleright a$ and $S' \triangleright a$, for $S \subseteq S'$, then the latter attack is implied by the former and is thus redundant (can be removed from the $AF^{\mathcal{S}}$ without change of semantics). This is also evident from the definition of $\blacktriangleright$, which is, in a sense, the "closure" of $\triangleright$.

3.1.3 Semantics (extensions) for SETAFs

With regards to semantics, it is easy to extend the definitions provided for the AF setting (e.g., in [40; 6]) so as to apply for the case of SETAFs (see [84; 60]). In all the following definitions, we consider a fixed SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ and a set of arguments $S \subseteq Ar$.

Definition 3.2. *S is said to be* conflict-free *iff it does not attack itself. Formally, S is conflict-free iff $S \not\blacktriangleright S$.*

Definition 3.3. *An argument $a \in Ar$ is said to be* acceptable *with respect to S, iff S defends a from all attacking sets of arguments in Ar. Formally, a is acceptable with respect to S iff $S \blacktriangleright T$ for all $T \subseteq Ar$ such that $T \blacktriangleright a$. S is said to be* admissible *iff it is conflict-free and each argument in S is acceptable with respect to S. Formally, S is admissible iff $S \not\blacktriangleright S$ and $S \blacktriangleright T$ for all $T \subseteq Ar$ such that $T \blacktriangleright S$.*

In [40], a characteristic function $\mathrm{F}_{AF^{\mathcal{D}}}$ was defined to return the arguments acceptable by a set of arguments in an argumentation framework $AF^{\mathcal{D}}$. This can be easily extended for SETAF (say $AF^{\mathcal{S}}$) as follows: $\mathrm{F}_{AF^{\mathcal{S}}} : 2^{Ar} \mapsto 2^{Ar}$, such that: $\mathrm{F}_{AF^{\mathcal{S}}}(S) = \{a \mid a \text{ is acceptable with respect to } S\}$.

Note that admissible extensions can (equivalently) be defined in terms of the characteristic function $\mathrm{F}_{AF^{\mathcal{S}}}$ as any conflict-free set such that $S \subseteq \mathrm{F}_{AF^{\mathcal{S}}}(S)$.

Definition 3.4. *An admissible set S is called a* complete extension *of $AF^{\mathcal{S}}$, iff all arguments that are acceptable with respect to S are in S. Formally, S is a complete extension of $AF^{\mathcal{S}}$ iff all the following conditions hold: (a) $S \not\blacktriangleright S$; (b) $S \blacktriangleright T$ for all $T \subseteq Ar$ such that $T \blacktriangleright S$; (c) If, for some $a \in Ar$, $S \blacktriangleright T$ for all $T \subseteq Ar$ such that $T \blacktriangleright a$, then $a \in S$.*

Obviously, complete extensions (of both AFs and SETAFs) can also be equivalently defined using the characteristic function: a conflict-free set S is a complete extension if and only if $\mathrm{F}_{AF^{\mathcal{S}}}(S) = S$.

Definition 3.5. *S is called a* preferred extension *of $AF^{\mathcal{S}}$, iff it is a complete extension and there is no other complete extension T such that $S \subset T$.*

In other words, a preferred extension is a maximal complete extension. In the standard AF setting, it has been shown that preferred extensions can be equivalently defined as maximal admissible extensions (see, e.g., [6]). It can be easily shown that the same holds true in the SETAF setting [55].

Definition 3.6. *S is called a* grounded extension *of $AF^{\mathcal{S}}$, iff it is a complete extension and there is no other complete extension T such that $T \subset S$.*

Essentially, grounded extensions are minimal complete extensions. Following similar results in the AF setting ([6]) we can easily show that the following are equivalent also in the SETAF setting:

- S is a grounded extension
- S is the complete extension such that $\{a \in Ar \mid S \rhd a\}$ is minimal
- S is the complete extension such that $Ar \setminus (S \cup \{a \in Ar \mid S \rhd a\})$ is maximal

Using the characteristic function, another equivalent characterisation can be formulated, namely that S is a grounded extension if and only if it is the least fixed point of $\mathrm{F}_{AF^{\mathcal{S}}}$ (see also [40]).

Definition 3.7. *S is called a* stable extension *of $AF^{\mathcal{S}}$, iff it is conflict-free and attacks all arguments in $Ar \setminus S$.*

Equivalently, S is stable if and only if $S = \{a \mid S \not\rhd a\}$. Also, for a stable extension S it holds that $S \cup \{a \mid S \rhd a\} = Ar$.

Moreover, we can easily show that stable extensions are also preferred, complete and admissible (see also [60] and Figure 4), thus S is a stable extension if and only if S is a preferred, complete or admissible extension that attacks all arguments in $Ar \setminus S$.

Example 3.8. *Consider the SETAF shown in Figure 2, whose extensions are shown in Table 1. Let us consider in more detail the complete extensions, which are: $\emptyset$, $\{a_1\}$, $\{a_2, a_3, a_5\}$. Note that, for example, $\{a_2, a_3\}$ is admissible and conflict-free but not complete, because it leaves out a_5, which is acceptable with respect to $\{a_2, a_3\}$. Similarly, $\{a_1, a_2\}$ is not a complete extension because it is not conflict-free, whereas $\{a_5\}$ and $\{a_1, a_5\}$ are not complete extensions because they are not admissible (a_5 is not acceptable with respect to the corresponding set in either case).*
Further, the minimal of the complete extensions (namely $\emptyset$) is also grounded, whereas

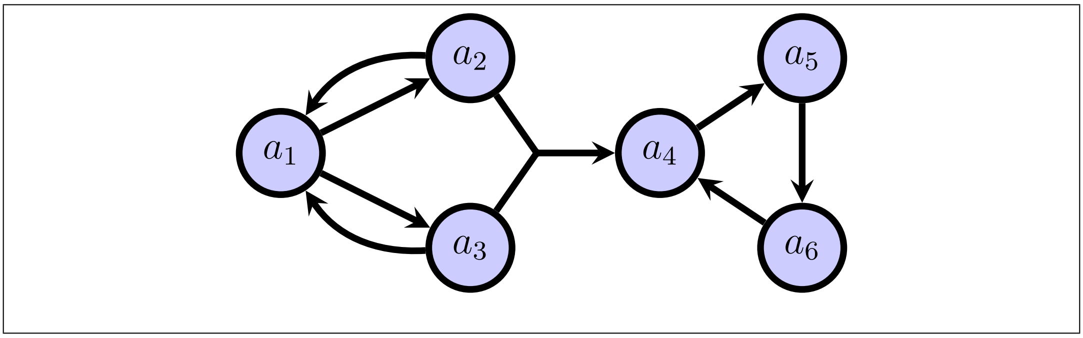


Figure 2: An example SETAF; set attacks are represented as arrows with multiple sources (e.g., $\{a_2, a_3\} \blacktriangleright a_4$); its extensions are shown in Table 1

Extension type	**Extensions**
Conflict-free	$\emptyset$, $\{a_1\}$, $\{a_2\}$, $\{a_3\}$, $\{a_4\}$, $\{a_5\}$, $\{a_6\}$, $\{a_1, a_4\}$, $\{a_1, a_5\}$, $\{a_1, a_6\}$, $\{a_2, a_3\}$, $\{a_2, a_3, a_5\}$, $\{a_2, a_3, a_6\}$, $\{a_2, a_4\}$, $\{a_2, a_5\}$, $\{a_2, a_6\}$, $\{a_3, a_4\}$, $\{a_3, a_5\}$, $\{a_3, a_6\}$
Admissible	$\emptyset$, $\{a_1\}$, $\{a_2\}$, $\{a_3\}$, $\{a_2, a_3\}$, $\{a_2, a_3, a_5\}$
Complete	$\emptyset$, $\{a_1\}$, $\{a_2, a_3, a_5\}$
Preferred	$\{a_1\}$, $\{a_2, a_3, a_5\}$
Grounded	$\emptyset$
Stable	$\{a_2, a_3, a_5\}$
Naive	$\{a_1, a_4\}$, $\{a_1, a_5\}$, $\{a_1, a_6\}$, $\{a_2, a_4\}$, $\{a_2, a_6\}$, $\{a_3, a_4\}$, $\{a_3, a_6\}$, $\{a_2, a_3, a_5\}$
Semi-stable	$\{a_2, a_3, a_5\}$
Eager	$\{a_2, a_3, a_5\}$
Ideal	$\emptyset$
Stage	$\{a_2, a_3, a_5\}$

Table 1: Extensions for the SETAF of Figure 2

the maximal ones ($\{a_1\}, \{a_2, a_3, a_5\}$) are also preferred. The latter ($\{a_2, a_3, a_5\}$) is also stable, because it attacks all other arguments.
Looking at the SETAF illustrated in Figure 3, we note that it also has three complete extensions (namely, $\{a_1\}$, $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$), the first of which is also the grounded one ($\{a_1\}$), whereas the other two are the preferred ones $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$). However, there is no stable extension, because none of the complete extensions attacks all other arguments in the SETAF.

Definition 3.9. *S is called a* naive extension *of $AF^{\mathcal{S}}$, iff it is conflict-free and is maximal w.r.t. set inclusion among the conflict-free subsets of Ar.*

Example 3.10. *Returning to the SETAF shown in Figure 2, we note that it has several naive extensions (see Table 1), which are essentially all the maximal subsets of Ar that do not attack themselves. On the other hand, the SETAF of Figure 3 has three naive extensions, namely $\{a_2, a_4\}$, $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$.*

Definition 3.11. *S is called a* semi-stable extension *of $AF^{\mathcal{S}}$, iff it is a complete extension and the set $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal w.r.t. set inclusion among all complete extensions of $AF^{\mathcal{S}}$.*

Essentially, semi-stable semantics give up the strict requirement of stable semantics that $S \cup \{a \in Ar \mid S \blacktriangleright a\} = Ar$, and require just that $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal.

Just like in stable extensions, semi-stable extensions are also preferred, complete and admissible (see also [60] and Figure 4), so the following are equivalent [55]:

- S is a semi-stable extension
- S is an admissible extension and the set $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal w.r.t. set inclusion among all admissible extensions of $AF^{\mathcal{S}}$.
- S is a preferred extension and the set $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal w.r.t. set inclusion among all preferred extensions of $AF^{\mathcal{S}}$.

Example 3.12. *For the SETAF illustrated in Figure 2, where a stable extension exists, this is also the (only) semi-stable extension of the SETAF (see Table 1). However, in the SETAF of Figure 3, where no stable extension exists, one can find two semi-stable extensions, namely: $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$. Each of these semi-stable extensions attack (or contain) all arguments except one (a_6 and a_7 respectively).*

Definition 3.13. *S is called an* eager extension *of $AF^{\mathcal{S}}$, iff it is a maximal (with respect to set inclusion) complete extension that is a subset of each semi-stable extension of $AF^{\mathcal{S}}$.*

The maximality requirement implies that we can replace the completeness requirement regarding S with admissibility, i.e., S is an eager extension of $AF^{\mathcal{S}}$, iff it is a maximal (with respect to set inclusion) admissible extension that is a subset of each semi-stable extension of $AF^{\mathcal{S}}$ (see [55]).

Example 3.14. *For the SETAF shown in Figure 2, there is only one semi-stable extension, so this is also the eager extension. In the SETAF of Figure 3, where there are two semi-stable extensions, the only eager extension is their intersection, i.e., $\{a_1\}$.*

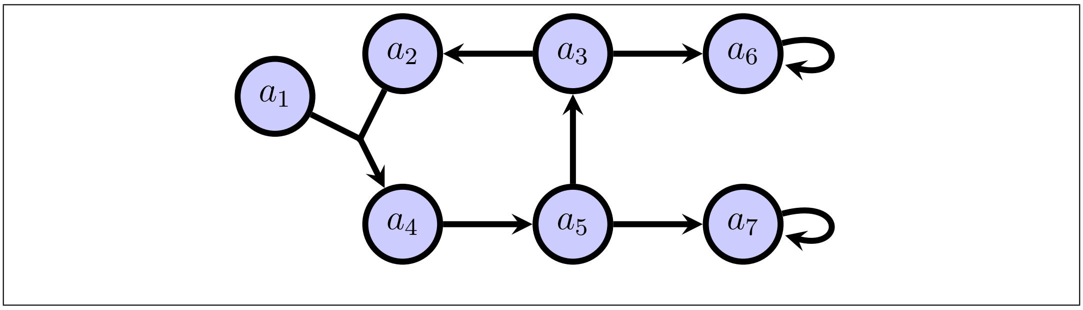


Figure 3: An example SETAF; set attacks are represented as arrows with multiple sources (e.g., $\{a_1, a_2\} \blacktriangleright a_4$); its extensions are shown in Table 2

Definition 3.15. *S is called an* ideal extension *of $AF^{\mathcal{S}}$, iff it is a maximal (with respect to set inclusion) complete extension that is a subset of each preferred extension of $AF^{\mathcal{S}}$.*

Extension type	**Extensions**
Complete	$\{a_1\}$, $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$
Preferred	$\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$
Grounded	$\{a_1\}$
Stable	(none exists)
Naive	$\{a_2, a_4\}$, $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$
Semi-stable	$\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$
Eager	$\{a_1\}$
Ideal	$\{a_1\}$
Stage	$\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$

Table 2: Extensions for the SETAF of Figure 3

Again, we can replace the requirement of S being complete, with S being preferred, or admissible (see [55]). Moreover, since an ideal extension is a subset of all preferred ones, it is not attacked by any preferred extension, and is in fact the largest complete extension (and admissible set) with this property. Using similar arguments, we can show that an ideal extension is the largest admissible set not attacked by any admissible set, and the largest admissible set not attacked by any complete extension [55].

Example 3.16. *For the SETAF shown in Figure 2, the two preferred extensions have an empty intersection, and $\emptyset$ happens to be a complete extension, so the only* *ideal extension is $\emptyset$. Similarly, for the SETAF of Figure 3, there are two preferred extensions, whose intersection is equal to $\{a_1\}$, and this happens to be a complete extension, so it is also ideal.*

Definition 3.17. *S is called a* stage extension *of $AF^{\mathcal{S}}$, iff it is conflict-free and $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal among all conflict-free subsets of Ar.*

Apparently, a stage extension is also naive (see also Figure 4), and, in fact, a stage extension can be equivalently defined as a naive extension such that $S \cup \{a \in Ar \mid S \blacktriangleright a\}$ is maximal among all naive extensions of Ar.

Example 3.18. *For the SETAF shown in Figure 2, which has a stable extension, the (only) stage extension is the stable one, i.e., $\{a_2, a_3, a_5\}$. For the SETAF illustrated in Figure 3, which has no stable extension, there are two stage extensions, which happen to be the same as the semi-stable ones, namely $\{a_1, a_2, a_5\}$, $\{a_1, a_3, a_4\}$. As explained in Example 3.12, each of these semi-stable extensions attack (or contain) all arguments except one (a_6 and a_7 respectively).*

3.1.4 Relationships among extensions

The various extensions are related, in the sense that certain types of extensions are stronger than others (e.g., a preferred extension is also complete, but not vice-versa). Moreover, some types of extensions are guaranteed to exist, others are not, and some extensions are unique. These results have been shown in various works for standard AFs, but [60] recast them for the SETAF case.

Figure 4 summarises these results. Each arrow in the graph pointing from semantics σ to σ' indicates that every σ-extension of a SETAF is also a σ'-extension of the same SETAF (e.g., every stable extension is also a stage extension). The number (possibly followed by $+$) that appears next to each semantics indicates the multiplicity of extensions for the specific semantics (e.g., every SETAF has at least one preferred extension). Similarly to Dung-style AFs, for certain semantics, the multiplicity of extensions is different for finite and infinite SETAFs, i.e., SETAFs with finite (respectively infinite) number of arguments. All such arrows are strict, i.e., no semantics is equivalent to another. Note also that in [60] the relationship among stage and naive semantics is missing.

3.1.5 Labellings

The semantics of AFs can be alternatively defined through labellings, as proposed in [27]. A labelling is formally defined as a function from arguments to the set $\{\texttt{in}, \texttt{out}, \texttt{undec}\}$. Intuitively, an argument belongs to the extension iff it is labelled

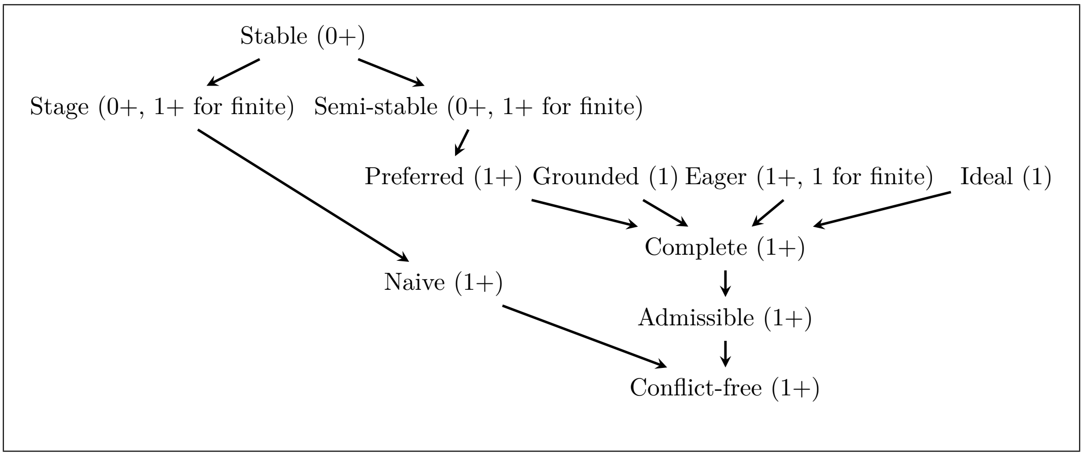


Figure 4: Inclusion relations and multiplicity of extensions for SETAF acceptability semantics

as `in`, whereas arguments labelled `out` are those attacked by the ones labelled `in`. Finally, the `undec` labelling is reserved for arguments that are not accepted, but are not attacked by an accepted argument either. Although labellings have been originally defined for AFs only [27], an adaptation for the SETAF case appears in [60]. Formally, a labelling is a function as follows:

Definition 3.19. *Consider a SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$. A* labelling *for $AF^{\mathcal{S}}$ is a total function $\mathcal{L}ab : Ar \mapsto \{\texttt{in}, \texttt{out}, \texttt{undec}\}$.*

Note that the labellings of a SETAF are defined over arguments (just like in AFs [27]), not sets of arguments.

Special classes of labellings can be defined (e.g., conflict-free labellings, admissible labellings, complete labellings, etc) and formally shown to correspond to the respective extensions (conflict-free, admissible, complete, etc). The correspondence is realised through two functions ($Ext2Lab$, $Lab2Ext$), which determine how to generate an extension given a labelling, or vice-versa. It can be shown that, if $\mathcal{L}ab$ is a labelling of a certain type (e.g., complete), then $Lab2Ext(\mathcal{L}ab)$ is an extension of the same type, and, vice-versa, if S is an extension of a certain type (e.g., complete), then $Ext2Lab(S)$ is a labelling of the same type.

In this section, we illustrate these ideas, dealing with complete labellings only, and refer to [27] and [60] for further details. We start with the definition of the functions $Lab2Ext$, $Ext2Lab$:

Definition 3.20. *Consider a SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$, and let $\mathcal{E}$ be the set of all possible extensions that can be created over $AF^{\mathcal{S}}$, $\mathcal{L}$ be the set of all possible labellings that can be created over $AF^{\mathcal{S}}$. Then:*

Ext2Lab: *We define the function $Ext2Lab : \mathcal{E} \mapsto \mathcal{L}$ such that, for $S \in \mathcal{E}$, $\mathcal{L}ab = Ext2Lab(S)$:*

- $\mathcal{L}ab(a) = \texttt{in}$ *for all* $a \in S$
- $\mathcal{L}ab(a) = \texttt{out}$ *for all* $a \notin S$, $S \blacktriangleright a$
- $\mathcal{L}ab(a) = \texttt{undec}$ *for all* $a \notin S$, $S \not\blacktriangleright a$

Lab2Ext: *We define the function $Lab2Ext : \mathcal{L} \mapsto \mathcal{E}$ such that, for $\mathcal{L}ab \in \mathcal{L}$, $Lab2Ext(\mathcal{L}ab) = \{a \in Ar \mid \mathcal{L}ab(a) = \texttt{in}\}$.*

Clearly, both *Ext2Lab* and *Lab2Ext* are well-defined. Moreover, note that $Ext2Lab(S)$ essentially labels `in` those arguments that are in S, `out` those arguments attacked by S, and `undec` the rest. On the other hand, $Lab2Ext(\mathcal{L}ab)$ contains only the arguments that are labelled `in` by $\mathcal{L}ab$.

Now, we can define complete labellings as follows:

Definition 3.21. *Let $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ be a SETAF. A labelling $\mathcal{L}ab : Ar \mapsto \{\texttt{in}, \texttt{out}, \texttt{undec}\}$ of $AF^{\mathcal{S}}$ is called* complete *iff for all $a \in Ar$:*

1. $\mathcal{L}ab(a) = \texttt{in}$ *if and only if* $\forall S \blacktriangleright a, \exists b \in S : \mathcal{L}ab(b) = \texttt{out}$
2. $\mathcal{L}ab(a) = \texttt{out}$ *if and only if* $\exists S \subseteq Ar$ *such that* $S \blacktriangleright a$ *and* $\mathcal{L}ab(b) = \texttt{in}$ *for all* $b \in S$

The next step is to prove that complete labellings correspond to complete extensions and vice-versa. The following two theorems prove these points:

Theorem 3.22. *Let $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ be a SETAF and $S \subseteq Ar$ a complete extension of $AF^{\mathcal{S}}$. Then, $Ext2Lab(S)$ is a complete labelling of $AF^{\mathcal{S}}$.*

Theorem 3.23. *Let $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ be a SETAF and $\mathcal{L}ab : Ar \mapsto \{\texttt{in}, \texttt{out}, \texttt{undec}\}$ a complete labelling of $AF^{\mathcal{S}}$. Then, $Lab2Ext(\mathcal{L}ab)$ is a complete extension of $AF^{\mathcal{S}}$.*

The above theorems show that complete labellings and complete extensions are essentially analogous ways to define the semantics of a SETAF. Similar theorems hold for the other types of extensions/labellings (see [60], Theorems 5.10, 5.11).

Complete extensions	Complete labellings
$S_1 = \emptyset$	$\mathcal{L}ab_1(a_1) = \texttt{undec}$, $\mathcal{L}ab_1(a_2) = \texttt{undec}$, $\mathcal{L}ab_1(a_3) = \texttt{undec}$, $\mathcal{L}ab_1(a_4) = \texttt{undec}$, $\mathcal{L}ab_1(a_5) = \texttt{undec}$, $\mathcal{L}ab_1(a_6) = \texttt{undec}$
$S_2 = \{a_1\}$	$\mathcal{L}ab_2(a_1) = \texttt{in}$, $\mathcal{L}ab_2(a_2) = \texttt{out}$, $\mathcal{L}ab_2(a_3) = \texttt{out}$, $\mathcal{L}ab_2(a_4) = \texttt{undec}$, $\mathcal{L}ab_2(a_5) = \texttt{undec}$, $\mathcal{L}ab_2(a_6) = \texttt{undec}$
$S_3 = \{a_2, a_3, a_5\}$	$\mathcal{L}ab_3(a_1) = \texttt{out}$, $\mathcal{L}ab_3(a_2) = \texttt{in}$, $\mathcal{L}ab_3(a_3) = \texttt{in}$, $\mathcal{L}ab_3(a_4) = \texttt{out}$, $\mathcal{L}ab_3(a_5) = \texttt{in}$, $\mathcal{L}ab_3(a_6) = \texttt{out}$

Table 3: Complete extensions and complete labellings for the SETAF of Figure 2.

Example 3.24. *Table 3 shows the complete labellings that correspond to the SETAF of Figure 2. Comparing complete extensions with complete labellings, we see that, e.g., the third labelling explicitly rejects a_6 (because it is attacked by a_5, which is accepted), but the second one makes no explicit decision on a_6, as the agent cannot make up its mind on how to resolve the cyclic attack among a_4, a_5, a_6. This distinction cannot be made with the corresponding complete extensions (first column of Table 3).*
Moreover, we can easily verify that:

- *the labellings can be generated through the corresponding extensions, using Definition 3.20;*
- *the labellings are all complete labellings (under Definition 3.21);*
- *the extensions could be generated from the labellings, using Definition 3.20.*

Another interesting point to note is that, for complete extensions and labellings, the relationship established by $Ext2Lab$, $Lab2Ext$, is bijective. In other words, for every labelling $\mathcal{L}ab$ and extension S of a SETAF, it holds $Ext2Lab(Lab2Ext(\mathcal{L}ab)) = \mathcal{L}ab$ and $Lab2Ext(Ext2Lab(S)) = S$. This is true for most, but not all, types of labellings; e.g., for admissible labellings, several different labellings may correspond to the same extension through $Lab2Ext$. A complete analysis of this phenomenon can be found in [60], where the concept of *proper labellings* is introduced to settle this question. Moreover, a rich set of results showing various properties of labellings can be found in [5; 6]. Although these results have been shown for AFs, recasting them for SETAFs is in most cases easy. Further details on the above are omitted, and the reader is referred to [5; 6; 60; 27] for more information.

3.2 Relating models for joint attacks with classical AFs

One of the obvious questions regarding SETAFs is whether they constitute a genuine extension of standard AFs (with more expressive power), or whether they are just a shorthand, i.e., syntactic sugar for knowledge that can be anyway represented in the standard Dung setting.

This is a very important question, because, if it turns out that AFs can be used to represent SETAFs, then we would be able to use the more intuitive SETAF formalism for modelling the attacks among arguments, while at the same time exploiting implementations and tools (and complexity results) developed for simple AFs to perform reasoning over the SETAF, by exploiting these translations. In the opposite case, SETAFs should be viewed as a separate, and more expressive branch of computational argumentation, and would require a different set of tools to support reasoning over them.

Interestingly, different works have addressed this problem, and answers have been given from different perspectives. In the rest of this section, we analyse four such works, namely:

- [48], who characterise the expressive power of AFs and SETAFs based on the notion of signatures [44], showing that SETAFs are strictly more expressive than AFs for the most popular semantics.
- [60], who circumvent the negative result of [48] by considering an exponential-sized translation of SETAFs to AFs and appropriate mappings among their semantics, for various semantics.
- [94], who applies an approach similar to [60], considering various alternative (and more condensed) translations with similar results (for the most popular semantics).
- [20], who consider the problem of translating Abstract Dialectical Frameworks (ADFs) [22] to AFs; given that SETAFs are a special case of ADFs, this result can be applied for the purposes of this chapter as well, albeit for a limited set of semantics.

3.2.1 Characterising the expressive power using signatures

The approach of [48] is based on *signatures* of different semantics (namely complete, grounded, preferred, stable, semi-stable, stage and naive [40; 24; 105; 19]) for AFs and SETAFs. Signatures have been originally defined in [44] as a way to characterise the expressive power of an AF, by way of conditions under which a candidate set of subsets of arguments are "realistic", i.e., they correspond to the extensions of some argumentation framework AF for a semantics of interest.

The idea has been extended to other types of argumentation frameworks (e.g., in [77; 98; 99; 100] for the ADF case [22]), and employed heavily as a means to compare the expressiveness of different argumentation frameworks with, e.g., normal logic programs and propositional logic [99; 100].

Formally, given a set of extensions (i.e., a set of sets of arguments) $\mathcal{E}$, $\mathcal{E}$ belongs to the signature Σ_σ^{AF} iff there is an AF framework whose set of extensions, under σ-semantics, is $\mathcal{E}$. Similarly, one can define Σ_σ^k, where k corresponds to a SETAF that admits only attacks where the attacking set has arity at most k (note that Σ_σ^1 coincides with Σ_σ^{AF} and Σ_σ^∞ coincides with the generic SETAF framework Σ_σ^{SETAF}). By definition, the notion of a signature expresses exactly the sets of extensions that can be constructed given a certain framework type, and for a certain semantics.

The focus of [48] is to compute the signatures Σ_σ^k for the considered semantics and for different k. As an example, they define the notion of an incomparable set of sets, where a set of sets $\mathcal{E}$ is incomparable iff all elements of $\mathcal{E}$ are pairwise incomparable, i.e., for $T, U \in \mathcal{E}$, $T \subseteq U$ implies $T = U$. Then, they prove that the set comprising all stable extensions of a SETAF is incomparable, i.e., $\Sigma_{\mathcal{ST}}^\infty = \{\mathcal{E} \mid \mathcal{E}$ is incomparable$\}$.

Signatures are a powerful tool for determining expressive power. Larger signatures imply that the corresponding framework type is more flexible (and thus more expressive). In particular, if $\mathcal{E} \notin \Sigma_\sigma^1$, then this means that one cannot construct an AF whose σ-extensions are exactly the ones in $\mathcal{E}$. Thus, by comparing Σ_σ^k for various $k \in \{1, 2, ..., \infty\}$, we can determine the relative expressive power of the different framework types.

Using this reasoning, the main conclusion of the paper is that, for all the considered semantics, and for all $k > 0$, SETAFs that allow for collective attacks of $k + 1$ arguments are more expressive than SETAFs that only allow for collective attacks of at most k arguments, because $\Sigma_\sigma^k \subset \Sigma_\sigma^{k+1}$. As a corollary, SETAFs are strictly more expressive than AFs, even if restricted to attacks of at most 2 arguments.

It is important however to interpret the above results under the correct lens. In particular, the results of [48] tell us that certain sets of extensions that can be constructed using SETAFs, cannot be *directly* constructed through AFs. More specifically, for a given $\mathcal{E} \in \Sigma_\sigma^{SETAF} \setminus \Sigma_\sigma^{AF}$, we know that one can create a SETAF whose set of σ-extensions is exactly $\mathcal{E}$; moreover, there is no AF whose set of σ-extensions is exactly $\mathcal{E}$.

However, if we don't insist on the *direct* construction, one may be able to succeed in constructing $\mathcal{E}$ through some AF, but in another, indirect way. In particular, one could define an appropriate mapping (algorithm) among sets of extensions (say f), and then construct an AF $AF^{\mathcal{D}}$ whose set of extensions is, say, $\mathcal{E}'$, where $f(\mathcal{E}') = \mathcal{E}$. For generality, one should also define a generic way to construct $AF^{\mathcal{D}}$ from the original SETAF $AF^{\mathcal{S}}$, via some other mapping (algorithm), say g. By the results of [48], this transformation cannot be a simple rearrangement of the attacks among the existing arguments of the SETAF, but should necessarily involve new, artificial arguments that would somehow encode the "collectivity of attacks".

3.2.2 An exponential translation to encode collectivity of attacks

This approach of "expanding" the SETAF with new arguments in order to get rid of collective attacks (and thus result in an AF) is followed in [60]. In that paper, a rather straightforward translation is followed, where, for any given SETAF $AF^{\mathcal{S}} = \langle Ar, \rhd \rangle$, one constructs a so-called *generated AF* $AF^{\mathcal{D}} = \langle Ar', att \rangle$, whose "arguments" are all the non-empty sets of arguments of the original SETAF (i.e., $Ar' = 2^{Ar} \setminus \{\emptyset\}$). The corresponding attack relation *att* follows in the obvious manner from $\rhd$. In the above terminology, this is the mapping g.

Then, the authors go on to identify the relationship among the σ-extensions of the $AF^{\mathcal{S}}$ and its corresponding generated $AF^{\mathcal{D}}$, as well as how one can identify the σ-extensions of $AF^{\mathcal{S}}$ through the σ-extensions of $AF^{\mathcal{D}}$, and vice versa (i.e., the mapping f and its inverse).

Various different semantics are considered, including the ones originally defined in [40] (conflict-free, admissible, complete, grounded, preferred, stable), but also naive [19], semi-stable [24], eager [25], ideal [41] and stage [105].

The conclusion of the above analysis is that many of the semantics (namely, complete, preferred, grounded, stable and ideal) admit a very simple one-to-one correspondence among the semantics of the SETAF and the generated AF. In particular, a set of arguments $S \subseteq Ar$ of the SETAF ($AF^{\mathcal{S}}$) is a σ-extension if and only if the set $2^S \setminus \{\emptyset\}$ is a σ-extension of the generated AF (recall that an argument in $AF^{\mathcal{D}}$ is a set of arguments from $AF^{\mathcal{S}}$).

For conflict-free and admissible extensions, the situation is similar, except that there are some additional σ-extensions of $AF^{\mathcal{D}}$ which do not follow this exact pattern. This has effects on the correspondence among naive extensions as well (recall that a naive extension is a maximal conflict-free set). Further, more convoluted correspondences exist for semi-stable, stage, and eager semantics, where the characterisations are complicated by the requirement of maximality (see [60] for details).

Complexity of characterisations put aside, the work of [60] shows that one can model a SETAF as an AF in a way that "preserves" the semantics, in the sense that one can determine the σ-extensions of the SETAF by just looking at the AF (and vice-versa). Alas, the proposed transformation for achieving this effect, results to an AF with an exponentially larger number of arguments compared to the SETAF. Note that if we count the size of a SETAF in terms of the number of arguments plus the number of attacks, then we may not get an exponential increase (if a sufficiently large number of attacks exist), although the exponential increase is still true in the worst-case scenario.

3.2.3 Considering more compact translations

A similar, but less extreme "expansion" scheme is followed in [94], where the problem of translating SETAFs to AFs is considered, among other things. The considered semantics are the standard Dung semantics, i.e., conflict-free, admissible, complete, preferred, grounded and stable [40].

To perform the translation, two translation schemes (and variations thereof) are considered: one is inspired by the so-called coalition approach and the other by the so-called defender approach. Both have a polynomial size compared to the SETAF (assuming that the size of the SETAF is considered to be equal to the number of attacks plus the number of arguments).

The coalition approach is similar to the one proposed in [60], where an argument in the AF is a set of arguments from the SETAF. However, in [94] a "condensed" version of the translation is considered, where not all subsets of Ar are included in the generated AF, but only those that are actually the initiators of an attack. Different ways to translate the attack relation are then considered, with different results with respect to the correspondence among the semantics of the SETAF and the corresponding AF.

The second translation scheme is inspired by [80], and uses arguments in the translated AF that represent "statements" regarding an argument in the SETAF (*e.g.* whether it is accepted, justified, rejected etc). More precisely, for every argument a in the SETAF, two arguments are included in the AF: the argument itself (a), as well as a' which stands for "a is rejected". Moreover, every attack in the SETAF is represented as an argument in the AF (these are called auxiliary arguments).

Then, appropriate attacks are introduced in the new framework. Namely, each argument a attacks its corresponding a', and a' attacks the auxiliary arguments representing an attack involving a as an attacker. The auxiliary arguments representing attacks, attack the corresponding recipient of the attack. In this way, a defends the auxiliary arguments it is involved in, so if a is not accepted, the attack itself (i.e., the auxiliary argument representing it) will not be accepted, and thus the recipient of the attack will be unaffected by the attack. Using this trick, the semantics of the SETAF can be appropriately captured by the AF.

For both translations, the correspondences provided among the σ-extensions of the SETAF and its generated AF are generally elegant, and quite similar to the correspondences of [60] (note however that the more complex cases of semi-stable, stage and eager semantics are not considered by [94]).

Despite that, a strong statement is made in [94] that no full exact SETAF-AF translation can be created. This statement is based on the idea of signatures, and follows similar lines of reasoning as in [48]. Therefore, it should be interpreted in the sense of a direct translation, as explained also in our analysis of the results of [48].

3.2.4 An indirect translation path, through ADFs

Another interesting translation results as a corollary of the work in [20]. In that paper, the authors do not study SETAFs, but ADFs [22]. An ADF is similar to an AF, except that the acceptance of an argument is determined by an acceptance condition (expressed as a propositional formula) over the acceptance of all its attackers. Thus, for example, one could say that an argument is accepted iff no more than two of its attackers are accepted, or that an argument is accepted iff all of its attackers are accepted.

Note that the expressive power of acceptance conditions allows ADFs to model various different types of relations among arguments, including attack, support, joint attacks or supports, as well as hybrid cases. In particular, it is easy to see that AFs and SETAFs are special cases of ADFs [93; 77].

Three different types of semantics have been defined for ADFs in [22], namely models, well-founded models and stable models. In the special case where an ADF is used to describe an AF (or a SETAF), models of the ADF correspond to the stable extensions of the AF (or SETAF) and well-founded models of the ADF correspond to the grounded extensions of the AF (or SETAF). Moreover, for this special case, stable models of the ADF and models of the ADF coincide (see [50], Proposition 1), so stable models of the ADF also correspond to stable extensions of the AF (or SETAF). It should be noted here that stable models have been retrospectively redefined in [23], but this redefinition does not break the above correspondences (see Theorem 4 in [23]).

In [20], the authors show that, given an ADF, one can generate an AF such that the stable extensions of the AF correspond (in a formal manner made clear in the paper) to the models of the ADF. A similar correspondence is also shown among the grounded extensions of an AF and the well-founded models of the ADF, as well as among the stable extensions of the AF and the stable models of the ADF. Although SETAFs were not in the scope of the work of [20], the fact that SETAFs are a special case of ADFs, allows us to apply their results to the case considered in this chapter. Moreover, [20] show that the proposed translations are polynomial in size, and can also be computed in polynomial time, where the size of the original ADF (corresponding to a SETAF) is computed as the number of arguments plus the size of the acceptance conditions of the arguments.

3.3 Computational considerations

In this section we give an overview on complexity results of SETAFs and discuss implementation approaches for evaluating SETAFs. As discussed in [47] understanding the inherent complexity of the reasoning tasks is crucial towards efficient implementations of argumentation systems. In particular, problems on different levels of complexity have different limits concerning scalability and require different techniques to be implemented in a scalable manner. We first introduce the computational tasks we are interested in, then discuss their complexity, and finally discuss algorithms and reduction-based approaches for these tasks.

3.3.1 Computational Problems

The standard problems studied in computational (abstract) argumentation are the tasks of computing extensions of a given semantics and computing the credulous or skeptical consequences under a given semantics [36; 47; 35]. These tasks are investigated in the literature on algorithms, systems, and complexity of abstract argumentation, and are the basis for the different tracks of the International Competition on Computational Models of Argumentation (ICCMA)[6] [101; 64]. In the following we provide formal definitions of these computational problems in the context of SETAFs. To this end we will use $\sigma(AF^{\mathcal{S}})$ to denote the σ-extensions of a SETAF $AF^{\mathcal{S}}$. We start with the function problems of computing one or all of the extensions of a SETAF w.r.t. a semantics σ:

- *Some Extension SE_σ*: Given SETAF $AF^{\mathcal{S}}$, compute an extension $E \in \sigma(AF^{\mathcal{S}})$.
- *Enumerate Extensions EE_σ*: Given SETAF $AF^{\mathcal{S}}$, compute the extension-set $\sigma(AF^{\mathcal{S}})$.

Beside these function problems we consider decision problems whose output is either yes or no. These problems are of particular interest as they are well-suited for being analysed with the techniques of complexity theory. To this end we consider the skeptical acceptance of an argument, i.e., an argument is skeptically accepted if it is contained in each extension, and credulous acceptance of an argument, i.e., an argument is credulously accepted if it is contained in some extension (for a given semantics σ):

[6] http://argumentationcompetition.org/

- *Credulous Acceptance* $Cred_\sigma$: Given SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ and an argument $a \in Ar$, is a contained in some $E \in \sigma(AF^{\mathcal{S}})$?
- *Skeptical Acceptance* $Skept_\sigma$: Given SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ and an argument $a \in Ar$, is a contained in each $E \in \sigma(AF^{\mathcal{S}})$?

Moreover, we consider the frequently-studied problems of verifying a given extension, deciding whether a SETAF has at least one extension, and deciding whether a SETAF has a non-empty extension. These problems are of some interest on their own but are in particular relevant as frequent sub-tasks of reasoning procedures. We next provide the formal definitions of these problems:

- *Verification of an extension* Ver_σ: Given SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ and a set of arguments $S \subseteq Ar$, is $S \in \sigma(AF^{\mathcal{S}})$?
- *Existence of an extension* $Exists_\sigma$: Given SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$, is $\sigma(AF^{\mathcal{S}}) \neq \emptyset$?
- *Existence of a non-empty extension* $Exists_\sigma^{\neg\emptyset}$: Given SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$, does there exist a set $E \neq \emptyset$ such that $E \in \sigma(AF^{\mathcal{S}})$?

3.3.2 Complexity results for SETAFs

We next discuss the computational complexity of the decision problems introduced in the previous section. The rationale behind the focus on decision problems is that tools of complexity theory are better suited for decision problems than for function problems and that, when chosen carefully, the complexity of the decision problems is also a good indicator for the complexity of the corresponding function problem. In computational argumentation the credulous and skeptical acceptance decision problems together are considered to be a good indicator for the complexity of a semantics.

In this section we assume the reader to have basic knowledge in computational complexity theory.[7] We will consider the following complexity classes: L (logarithmic space), P (polynomial time), NP (non-deterministic polynomial time), coNP (complement of a NP problem), Θ_2^{P} (polynomial time with non-adaptive NP-oracle calls), Σ_2^{P} (non-deterministic polynomial time with NP-oracle calls), Π_2^{P} (complement of a Σ_2^{P} problem), and $\mathsf{D}_2^{\mathsf{P}}$ (intersection of a Σ_2^{P} and a Π_2^{P} language).

[7] For a gentle introduction into complexity theory in the context of argumentation the reader is referred to [47]).

We have the following relations between these complexity classes:

$$\mathsf{L} \subseteq \mathsf{P} \subseteq \begin{array}{c}\mathsf{NP}\\ \mathsf{coNP}\end{array} \subseteq \Theta_2^{\mathsf{P}} \subseteq \begin{array}{c}\Sigma_2^{\mathsf{P}}\\ \Pi_2^{\mathsf{P}}\end{array} \subseteq \mathsf{D}_2^{\mathsf{P}}$$

We follow [49] and start our complexity analysis with the observation that SETAFs generalize Dung AFs and thus all the decision problems are at least as hard as the corresponding problem for Dung AFs (cf. [47, Table 1]). Interestingly, one can also obtain the same upper bounds (see Table 4) as we discuss below. These results for SETAFs show the same complexity as the corresponding Table for Dung AFs (cf. [47, Table 1][8]). However, there is a subtle difference between the complexity results for Dung AFs and SETAFs. In both cases the complexity is stated w.r.t. the size of the input framework, which in case of Dung AFs is often interpreted as w.r.t. the number of arguments $|Ar|$ in the input framework. This interpretation is not valid for SETAFs where the number of attacks $|\rhd|$ can be exponentially larger than the number of arguments $|Ar|$ (this even holds for normal forms where redundant attacks are removed). Thus, one has to consider the complexity w.r.t. the number of arguments plus the representation size of the attacks $\rhd$. The latter is bounded bound by $|Ar| \cdot |\rhd|$, i.e., is polynomially bounded in $|Ar| + |\rhd|$. We can thus interpret the complexity results for SETAFs in Table 4 as w.r.t. $|Ar| + |\rhd|$ [9].

The crucial observation towards the upper bounds is that checking basic properties of a set of arguments, although it is more evolved than in Dung AFs, can still be performed in L. First, to test whether a set S is conflict-free one can iterate over all attacks $(T, a) \in \rhd$ and check that $T \cup \{a\} \not\subseteq S$. Second, to test $S \blacktriangleright T$ one can iterate over all attacks $(U, b) \in \rhd$ and test whether $U \subseteq S$ and $b \in T$. Finally, a simple algorithm for testing that a set S defends an argument a iterates over all attacks $(T, a) \in \rhd$ and for each of these attacks checks that $S \blacktriangleright T$. That is, for all three problems we just need to store a small number of pointers to the input which can be done in logarithmic space.

Proposition 3.25. *Given a SETAF $AF^{\mathcal{S}} = \langle Ar, \rhd \rangle$, a set of arguments $S \subseteq Ar$, and an argument $a \in Ar$, deciding whether S is conflict-free, deciding whether $S \blacktriangleright a$, and deciding whether $a \in \mathrm{F}_{AF^{\mathcal{S}}}(S)$ are in* L.

Notice that most of the complexity upper bounds for Dung AFs are based on the fact that these three problems can be solved in polynomial-time, and thus these

[8] Notice that [47, Table 1] includes $\mathcal{CF}2$ semantics which has not yet been generalised to SETAFs and is thus not included in Table 4. On the other hand, we include eager semantics which has not been considered in [47, Table 1] (see [45, Table 2] for the complexity results of eager semantics in Dung AFs).

[9] For a more fine-grained analysis of algorithms for SETAFs one might take into account the actual representation size of the attacks, cf. [53].

upper bounds also apply to SETAFs (cf. Table 4). We next exemplify this for the credulous acceptance problem of stable semantics.

Proposition 3.26. *We have that* $\mathit{Ver}_{\mathcal{ST}} \in \mathsf{L}$ *and* $\mathit{Cred}_{\mathcal{ST}}$ *is* NP*-complete.*

Proof. First, consider the verification problem $\mathit{Ver}_{\mathcal{ST}}$ and an arbitrary SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$. We can verify that a given set S is a stable extension of $AF^{\mathcal{S}}$ by (a) checking that S is conflict-free and (b) checking that for each $a \in Ar \setminus S$ we have $S \blacktriangleright a$. As both can be done in L, we obtain the L membership of $\mathit{Ver}_{\mathcal{ST}}$.

Now consider the credulous acceptance problem $\mathit{Cred}_{\mathcal{ST}}$. The NP-hardness is by the corresponding result for AFs. For the upper bound consider an arbitrary SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$ and an argument $a \in Ar$. We can decide the credulous acceptance of a in $AF^{\mathcal{S}}$ by a standard guess & check algorithm. That is, one first uses the non-determinism to guess a set E and then use a deterministic part to verify that E is a stable extension and contains the argument a. This gives an NP procedure for $\mathit{Cred}_{\mathcal{ST}}$. □

Next, let us consider the complexity of ideal semantics, as it is the only case where the upper bound for Dung AFs [43] does not directly apply to SETAFs. Recall that the ideal extension can be characterised as the maximal admissible set that is not attacked by any other admissible set (Definition 3.15). In order to compute the ideal extension we thus use NP-oracle queries that for each argument ask (a) whether it is credulously accepted w.r.t. preferred semantics and (b) whether it is attacked by some admissible set. We then consider the set E^0 of all arguments that are credulously accepted but not attacked by an admissible set. Notice that E^0 is conflict-free by construction and it is an over-approximation of the ideal extension. We then compute the maximal admissible subset of E^0 by iteratively computing sets E^{i+1} by removing arguments that are not defended by E^i until we reach a fixed-point E. We then have that E is the ideal extension. We have that the NP-oracle queries of the above procedure are independent of each other and thus can be executed in parallel. Moreover, each iteration of the fixed-point computation is in polynomial-time and the fixed-point is reached after at most $n/2$ iterations, i.e., one can compute the fixed-point in polynomial time. Thus the above is a Θ_2^{P}-algorithm for computing the ideal extension. Hence, we obtain Θ_2^{P} upper bounds for all reasoning tasks of ideal semantics.

3.3.3 Algorithms for SETAFs

The field of algorithms for SETAFs is rather under-explored with the exception of [82]. The former studies algorithmic ideas for preferred semantics. We recapitulate

σ	Cred_σ	Skept_σ	Ver_σ	Exists_σ	$\mathit{Exists}_\sigma^{\neg\emptyset}$
Conflict-free	in L	trivial	in L	trivial	in L
Naive	in L	in L	in L	trivial	in L
Grounded	P-c	P-c	P-c	trivial	in L
Stable	NP-c	coNP-c	in L	NP-c	NP-c
Admissible	NP-c	trivial	in L	trivial	NP-c
Complete	NP-c	P-c	in L	trivial	NP-c
Ideal	Θ_2^P-c	Θ_2^P-c	Θ_2^P-c	trivial	Θ_2^P-c
Eager	Π_2^P-c	Π_2^P-c	D_2^P-c	trivial	Π_2^P-c
Preferred	NP-c	Π_2^P-c	coNP-c	trivial	NP-c
Semi-stable	Σ_2^P-c	Π_2^P-c	coNP-c	trivial	NP-c
Stage	Σ_2^P-c	Π_2^P-c	coNP-c	trivial	in L

Table 4: Complexity of SETAFs ($\mathcal{C}$-c denotes completeness for class $\mathcal{C}$).

their main observations in terms of a simple algorithm (see Algorithm 1) in the style of today's labelling-based algorithms ([36; 35]).

The rough idea of labelling-based algorithms is to start with all arguments unlabelled, in each step pick an argument and then consider two branches: one where we add the argument to the extension, i.e., labelled `in`; and one where we decide that the argument is excluded from the extension, i.e., labelled `out` or `undec` (cf. Section 3.1.5). When all arguments are labelled, one tests whether the labelling is valid w.r.t. the considered semantics and, if so, it is added to the output. By that procedure we would consider all possible candidates for valid labellings and thus also obtain all the extensions. In order to design an efficient algorithm one aims to cut off branches that do not lead to valid labellings as soon as possible. One approach are the so-called label propagation rules, i.e., one uses the already fixed labels of the arguments to conclude that other arguments have to obtain a certain label and by that avoids unnecessary branching in the algorithm. For instance, for preferred semantics, given the set of arguments $\mathcal{L}ab_{\mathtt{in}}$ labelled `in` by a partial labelling $\mathcal{L}ab$ we can conclude that all arguments in the set $\mathcal{L}ab_{\mathtt{in}}^{+}$, i.e., arguments a with $\mathcal{L}ab_{\mathtt{in}} \blacktriangleright a$, must be labelled `out`. Moreover, for attacks that target $\mathcal{L}ab_{\mathtt{in}}$ and have only one argument outside of $\mathcal{L}ab_{\mathtt{in}}^{+}$ we have that this argument has to be labelled `out`. This is captured by the set $\mathcal{L}ab_{\mathtt{in}}^{\leftarrow}$ defined as $\mathcal{L}ab_{\mathtt{in}}^{\leftarrow} = \{a \in \mathit{Ar} \mid \mathcal{L}ab_{\mathtt{in}} \cup \{a\} \blacktriangleright \mathcal{L}ab_{\mathtt{in}}\}$. This propagation of `out` labels is implemented in Line 8 of Algorithm 1 and triggered whenever a new argument is labelled in. Another observation is that we cannot label an argument in if this would cause a conflict in the set $\mathcal{L}ab_{\texttt{in}}$. Many cases where this could happen are already covered by the propagation rules for out labels, but these rules do not cover attacks $(S \cup \{a\}, a) \in \rhd$ with $S \subseteq \mathcal{L}ab_{\texttt{in}}$. This propagation is implemented by the if condition on Line 4, which prevents the algorithm from starting the branch where the argument a is added to the extension. Finally, when an argument a is already defended by $\mathcal{L}ab_{\texttt{in}}$ then, due to the maximality of preferred extensions, we know that this argument is in each preferred extension containing $\mathcal{L}ab_{\texttt{in}}$ and thus we must label a by in. This propagation is implemented by the if condition on Line 12, which prevents the algorithm from starting the branch where the argument a is excluded from the extension.

We obtain that Algorithm 1 returns the preferred labellings of a given SETAF $AF^{\mathcal{S}}$. Notice that the algorithm can be easily adapted to compute complete labellings, by removing the maximality check on Line 16, or admissible sets, by removing the maximality check on Line 16 and the if condition on Line 12. We can roughly estimate the running time of these algorithms by $O(\exp(|Ar|) \cdot \mathsf{poly}(|Ar|, |\rhd|))$. Notably only the polynomial part depends on the number of attacks while the exponential part solely depends on the number of arguments. Finally, recent work [53] suggests to not just label arguments but also label the attacks of a SETAF. It then studies possible label propagation-rules for stable and complete semantics and provides a linear time algorithm (linear w.r.t. the representation of the SETAF) for grounded semantics.

3.3.4 Systems and Reduction-based Approaches

Reduction-based approaches have been successfully applied in the design of argumentation systems, most prominently by systems that are based on modern SAT-solver technology or answer-set programming [35]. For SETAFs the only system discussed in the literature, i.e., the SETAF module of the ASPARTIX[10] system [49], is based on answer-set programming.

Reduction to Answer-set Programming. Answer-set programming (ASP) [79; 85] is a declarative problem solving paradigm with its roots in logic programming and non-monotonic reasoning. Today's answer-set systems [66; 75] support a rich language and are capable of solving hard problems efficiently. Thus, ASP is a convenient formalism to implement argumentation systems. The ASPARTIX approach [57] to argumentation problems relies on a query-based implementation

[10] https://www.dbai.tuwien.ac.at/research/argumentation/aspartix

Algorithm 1 pref-lab($AF^{\mathcal{S}}$)

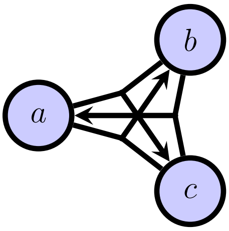
```
Require: SETAF $AF^{\mathcal{S}} = \langle Ar, \triangleright \rangle$, global variable $\mathcal{L}$
Ensure: $\mathcal{L}$ is the set of preferred labellings
$\mathcal{L} = \emptyset$, $\mathcal{L}ab = \langle \emptyset, \emptyset, \emptyset \rangle$
pref-lab($AF^{\mathcal{S}}, \mathcal{L}ab$)
function pref-lab($F, \mathcal{L}ab$)
Require: SETAF $F = \langle A, R \rangle$, partial labelling $\mathcal{L}ab$, global variable $\mathcal{L}$
if there is an argument $a \in A$ not labeled by $\mathcal{L}ab$ then
    $a \leftarrow$ pick some unlabeled argument
    if $\mathcal{L}ab_{\mathtt{in}} \cup \{a\} \in \mathcal{CF}(F)$ then
        $\mathcal{L}ab'_{\mathtt{in}} = \mathcal{L}ab_{\mathtt{in}} \cup \{a\}$,
        $\mathcal{L}ab'_{\mathtt{out}} = \mathcal{L}ab_{\mathtt{out}} \cup \mathcal{L}ab'^{+}_{\mathtt{in}} \cup \mathcal{L}ab'^{\leftarrow}_{\mathtt{in}}$
        $\mathcal{L}ab'_{\mathtt{undec}} = \mathcal{L}ab_{\mathtt{undec}} \setminus \mathcal{L}ab'_{\mathtt{out}}$
        pref-lab($AF^{\mathcal{S}}, \langle \mathcal{L}ab'_{\mathtt{in}}, \mathcal{L}ab'_{\mathtt{out}}, \mathcal{L}ab'_{\mathtt{undec}} \rangle$)
    end if
    if $\{a\} \notin \mathcal{F}_F(\mathcal{L}ab_{\mathtt{in}})$ then
        pref-lab($AF^{\mathcal{S}}, \langle \mathcal{L}ab_{\mathtt{in}}, \mathcal{L}ab_{\mathtt{out}}, \mathcal{L}ab_{\mathtt{undec}} \cup \{a\} \rangle$)
    end if
else
    if $\mathcal{L}ab_{\mathtt{in}} \in \mathcal{AD}(F)$ and $\mathcal{L}ab_{\mathtt{in}}$ $\subseteq$-max among $\{\mathcal{L}ab_{\mathtt{in}} \mid \mathcal{L}ab \in \mathcal{L}\}$ then
        $\mathcal{L} = \mathcal{L} \cup \{\mathcal{L}ab\}$
    end if
end if
endFunction
```

where the argumentation framework is provided as an input database, and one provides fixed queries encoding the different argumentation semantics and reasoning tasks.

Here we briefly highlight the main differences between the ASP encodings of Dung AFs [57] and SETAFs [49]. To this end, we first briefly recall the basic terminology for logic programs (for rigorous definitions see [35] or [36]). A logic program (under the answer-set semantics) is a set of disjunctive rules r of the form

$$a_1 \vee \cdots \vee a_n \ \leftarrow b_1, \ldots, b_k, \ \mathit{not}\ b_{k+1}, \ldots, \ \mathit{not}\ b_m$$

where $a_1, \ldots, a_n$ and $b_1, \ldots, b_m$ are atoms, and *not* stands for *default negation.* We refer to a as a positive literal, while we refer to *not* a as a default negated literal. The *head* of r is the set $\{a_1, \ldots, a_n\}$ and the *body* of r is $\{b_1, \ldots, b_m\}$, and a rule



$\pi_{setaf}(AF^{\mathcal{S}})$

$$\begin{gathered}\mathbf{arg}(a).\ \mathbf{arg}(b).\ \mathbf{arg}(c).\\ \mathbf{att}(r1,c).\ \mathbf{mem}(r1,a).\ \mathbf{mem}(r1,b).\\ \mathbf{att}(r2,b).\ \mathbf{mem}(r2,a).\ \mathbf{mem}(r2,c).\\ \mathbf{att}(r3,a).\ \mathbf{mem}(r3,b).\ \mathbf{mem}(r3,c).\end{gathered}$$

Figure 5: The SETAF $AF^{\mathcal{S}} = \langle\{a,b,c\},\{(\{a,b\},c),(\{a,c\},b),(\{b,c\},a)\}\rangle$ and its ASP-encoding $\pi_{setaf}(AF^{\mathcal{S}})$.

$\pi_{\mathcal{CF}}$

$$\begin{gathered}\mathbf{in}(Y) \leftarrow \mathbf{arg}(Y), \mathit{not}\ \mathbf{out}(Y).\\ \mathbf{out}(Y) \leftarrow \mathbf{arg}(Y), \mathit{not}\ \mathbf{in}(Y).\\ \mathbf{blocked}(R) \leftarrow \mathbf{mem}(R,X), \mathbf{out}(X).\\ \leftarrow \mathbf{in}(X), \mathbf{att}(R,X), \mathit{not}\ \mathbf{blocked}(R).\end{gathered}$$

$\pi_{\mathcal{AD}}$

$$\begin{gathered}\mathbf{in}(Y) \leftarrow \mathbf{arg}(Y), \mathit{not}\ \mathbf{out}(Y).\\ \mathbf{out}(Y) \leftarrow \mathbf{arg}(Y), \mathit{not}\ \mathbf{in}(Y).\\ \mathbf{blocked}(R) \leftarrow \mathbf{mem}(R,X), \mathbf{out}(X).\\ \leftarrow \mathbf{in}(X), \mathbf{att}(R,X), \mathit{not}\ \mathbf{blocked}(R).\\ \mathbf{defeated}(R) \leftarrow \mathbf{att}(R,X), \mathbf{mem}(R,Y),\\ \mathbf{att}(R2,Y), \mathit{not}\ \mathbf{blocked}(R2).\\ \leftarrow \mathbf{in}(X), \mathbf{att}(R,X), \mathit{not}\ \mathbf{defeated}(R).\end{gathered}$$

Figure 6: ASP Encodings $\pi_{\mathcal{CF}}$, $\pi_{\mathcal{AD}}$ for $\mathcal{CF}$ and $\mathcal{AD}$ semantics of SETAFs.

r is a *constraint* if $n = 0$. A *fact* is a ground rule without disjunction ($n = 1$) and with an empty body. An *input database* is a set of facts.

In order to evaluate SETAFs with ASP, in a first step, we have to encode SETAFs as an input database for the ASP-program. We introduce three predicates **arg**, **att**, and **mem** to encode a SETAF $AF^{\mathcal{S}} = \langle Ar, \rhd\rangle$. The predicate **arg** is used to encode arguments, the latter two to encode the set attacks, i.e., **att** encodes which argument is attacked by an attack and **mem** encode which arguments are required to attack that argument. Notice that, this encoding uses a unique identifier for each attack in $\rhd$. The encoding of a SETAF $AF^{\mathcal{S}} = \langle Ar, \rhd\rangle$ is then given by $\pi_{setaf}(AF^{\mathcal{S}}) = \{\mathbf{arg}(a).\mid \text{for } a \in Ar\} \cup \{\mathbf{att}(r,x).\mid \text{for } r \in \rhd \text{ and } r = (S,x)\} \cup \{\mathbf{mem}(r,y).\mid \text{for } r \in \rhd, r = (S,x), \text{ and } y \in S\}$ (cf. Figure 5). While arguments are represented in the same way as in Dung AFs, Dung AFs allow for a simpler representation of attacks. That is, the encoding of AFs ([35]) only uses one binary predicate **att** to encode the attacks, containing the attacker and the attacked argument of each attack, and does not use identifiers for attacks.

When it comes to the encoding of semantics one uses predicates $\mathbf{in}(\cdot)$, $\mathbf{out}(\cdot)$

to guess whether an argument is in the extension or not (in the same way as for AFs). Notice that the predicate $\mathbf{out}(\cdot)$ encodes that an argument is not in the extension and does not correspond to the label `out`. This guess builds up all possible subsets of arguments which are then filtered by adding constraints that reflect the specific semantics. Here the SETAF encodings differ from the AF encodings as they explicitly define statuses of attacks. First, we call an attack $(T, a) \in \triangleright$ blocked w.r.t. a set $E \subseteq Ar$ if $T \not\subseteq E$. Second, we consider an attack $(T, a) \in \triangleright$ to be defeated by a set E iff $E \blacktriangleright T$. We will exemplary discuss the encodings $\pi_{\mathcal{CF}}$, $\pi_{\mathcal{AD}}$ for conflict-free sets and admissible sets respectively (cf. Figure 6). In the encoding of the conflict-freeness, with the first two rules one guesses a subset of arguments, the third rule computes the blocked attacks, and the constraint in the fourth line rules out all sets that contain an argument X and have a non-blocked rule attacking X. That is, if we compute the answer-sets of the combined program $\pi_{setaf}(AF^{\mathcal{S}}) \cup \pi_{\mathcal{CF}}$ the answer-sets correspond to the conflict-free sets, i.e., the conflict-free sets are given by the $\mathbf{in}(\cdot)$ predicate in the answer-sets. Next, we further extend $\pi_{\mathcal{CF}}$ to an encoding $\pi_{\mathcal{AD}}$ for admissible semantics. That is, we add a rule that computes the defeated attacks and a constraint that rules out sets where an argument of the set is attacked by an undefeated attack. Thus, if we compute the answer-sets of the combined program $\pi_{setaf}(AF^{\mathcal{S}}) \cup \pi_{\mathcal{AD}}$ the answer-sets correspond to the admissible sets.

Other Reduction-based Approaches. For Dung AFs and their generalisations, several reduction-based approaches have been studied in the literature and often resulted in argumentation systems [36]. In particular, systems based on modern SAT-solving systems have been successful [101; 64]. Beside ASP, none of these approaches have been considered in the literature on SETAFs so far. However, very recently a first version of the SAT-based SETAF system *joukko* appeared online[11]. Thus, one approach towards an efficient SETAF system would be to extend existing approaches that have been successful for AFs to SETAFs. Another approach is to translate SETAFs to AFs or ADFs and use one of the existing systems for these formalisms to evaluate SETAFs. Translations from SETAFs to AFs have been presented in [94] and [60](see also Section 3.2 in this chapter). However, when using these translations one is faced with an exponential blow-up in the arguments and thus these translations are not well-suited for computational matters. Recall, that algorithms for SETAFs scale polynomially w.r.t. the number of attacks and exponentially w.r.t. the number of arguments. Thus translating attacks to arguments and using AFs tools can results in a serious computational overhead. Con-

[11] `https://bitbucket.org/andreasniskanen/joukko`

cerning the latter, there are rather simple translations of SETAFs into ADFs [77; 93] (see Section 3.2.4) which do not increase the number of arguments. That is, one can efficiently encode a SETAF as an ADF and then use one of the existing systems for ADFs, e.g., k++ADF[12] [76], YADF[13] [21], or DIAMOND[14] [58], to evaluate the SETAF. The attentive reader may argue that the computational complexity of ADFs is higher than that of SETAFs and thus such a reduction might result in significant overheads. However, modern ADF systems are sensitive to the actual complexity of the acceptance conditions in the processed ADF and thus the overheads when processing ADFs with acceptance conditions generated from SETAFs probably will not be as high as one would expect from the worst-case complexity gap.

3.4 Alternative models for attacks involving sets of arguments

SETAFs have not been the only attempt to formalise *collective* attacks[15] in argumentation systems. There have been earlier or more recent related approaches, both in abstract and structured argumentation, each of which captures a slightly different notion of collective attack and with a different aim.

One of the earliest approaches to formalise collective attacks in abstract argumentation was the *collective argumentation theories* proposed by [18]. These are generalisations of Dung's abstract argumentation frameworks aimed at the representation of the semantics of disjunctive logic programs, but also, more generally, at the description of "reasoning situations in which the conflict between incompatible views or theories is global and cannot be reduced to particular claims made by these theories". [18] proposes a four-valued semantics, i.e., each argument is assigned a subset of $2^{\{t,f\}}$ and attacks occur among sets of arguments (e.g. $S \hookrightarrow T$) and are interpreted as "at least one of the arguments in the attacked set (T) should be rejected whenever all the arguments from the attacking set (S) are accepted".

[33] introduced the notion of coalitions of arguments to represent sets of non-conflicting arguments that are related via the support relation in a bipolar argumentation framework (BAF). Using this notion, a bipolar argumentation framework $AF^{\mathcal{B}}$ can be translated into a Dung-style meta-argumentation framework $C(AF^{\mathcal{B}})$, called "Coalition AF", in which the arguments represent coalitions of arguments of $AF^{\mathcal{B}}$ and the attacks among arguments (called *c-attacks*) correspond to attacks among elements of the corresponding coalitions: S c-attacks T in $C(AF^{\mathcal{B}})$, iff there

[12] `https://www.cs.helsinki.fi/group/coreo/k++adf/`

[13] `http://www.dbai.tuwien.ac.at/proj/adf/yadf/`

[14] `http://diamond-adf.sourceforge.net/`

[15] We use the term collective to refer to any kind of attack relation that involves sets of arguments, either as attackers or as targets of an attack, or both.

exist arguments $a, b \in AF^{\mathcal{B}}$ such that $a \in S, b \in T$ and a attacks b in $AF^{\mathcal{B}}$. All arguments belonging to a coalition are then treated in the same way when computing the acceptable arguments: an argument a is acceptable (under the preferred, stable or grounded semantics) in $AF^{\mathcal{B}}$ iff it is a member of a coalition S, which is acceptable (under the same semantics) in $C(AF^{\mathcal{B}})$.

The framework proposed in [63] also considers sets of arguments, but as recipients of disjunctive attacks from single arguments. In this framework, the result of an attack from an argument a that is labelled `in`, to a set of arguments S, is that at least one of the arguments in S must be labelled `out`. Definition 2.8 and Theorem 2.9 of the same paper show how a finite disjunctive framework can be converted to a Dung-style AF with the same set of extensions, which, combined with the results on the relationship between SETAF and AF that we present in Section 3.2, provide a way to associate SETAF with disjunctive argumentation frameworks. Note also that [84] also provides a way to model disjunctive attacks using SETAFs, using the notion of "indeterministic defeat" [104] (see Section 3.1 for details).

CumulA [103; 104] is an example of a structured argumentation model that supports collective attacks. In this model, arguments are tree-like structures that represent how a conclusion is supported. In order to support situations where a set of arguments should be collectively defeated (*collective defeat*) or at least one of the arguments in a set should be defeated (*indeterministic defeat*), it uses *compound defeaters*, i.e., attack relations where either the source or the target of the attack (or both) are sets of arguments. The meaning of a compound defeater is different than that of joint attacks in SETAFs: if all arguments in the attacking set are undefeated, the arguments in the attacked set are defeated as a group unless one of the arguments in the attacked set has already been defeated by another defeater. In the latter case the compound defeater becomes *inactive*.

Another structured argumentation formalism that incorporates the notion of collective attacks is the Abstract Argumentation Systems (AAS) from [106]. An AAS is defined as a triple $(\mathcal{L}, \mathcal{R}, \leq)$, where $\mathcal{L}$ is a language containing the symbol $\perp$, which represents a contradictory proposition, $\mathcal{R}$ is a set of (strict and defeasible) inference rules, and $\leq$ is a preorder on the set of arguments, called *order of conclusive force*, and determining "the relative difference in strength among arguments". Arguments are defined as chains of rules organised as trees. The notion of defeat in AAS is used to capture and resolve conflicts among groupwise incompatible arguments: a set of arguments X defeats an argument a if $X \cup \{a\}$ is incompatible (there is a strict argument b that is based on the conclusions of $X \cup \{a\}$ and has conclusion $\perp$) and X is not undermined by a (there is no $c \in X$ such that $a < c$).

Defeasible Logic [89], which, as shown in [69] has an argumentation-theoretic semantics, also supports a type of collective attacks, called *team defeat*. This logic includes a rule priority relation, which is used to resolve conflicts between rules with contradictory conclusions. An attack on a rule r with conclusion p from a rule r' with conclusion $\neg p$ can be invalidated by another rule r'' also with conclusion p that is superior to r'. In this case, we say that r and r'' team defeat r'. Using this feature, we conclude that p is true if for every applicable rule that supports $\neg p$, there is a superior rule for p; in other words, if the rules for $\neg p$ are team defeated by the rules for p[16]. In order to support this feature, the argumentation-theoretic characterisation of Defeasible Logic defines arguments as sets of proof trees supporting the same conclusion and team defeat as a relation between two arguments with opposite conclusions, and requires that an argument team defeats all its attacking arguments to become acceptable. Team defeat is also supported by other rule-based non-monotonic logics, which use preferences on rules, such as Courteous Logic Programs [70] and Order Logic [73]. An interesting problem is to study the possibility of mapping Defeasible Logic, or any of the other rule-based non-monotonic logic that supports team defeat, to SETAF by defining arguments as proof trees and by representing team defeat, between a set of rules R supporting the same conclusion and a rule s supporting the opposite conclusion, as a joint attack from the set of arguments that have a top rule in R to each argument that has s as its top rule.

[10] recently introduced a semi-structured formalism for argumentation, called LAF-ensembles, capturing a set of essential features of structured arguments, such as their conclusion, their "attackable elements" and their subarguments. They also defined a family of abstract argumentation frameworks, called *set-based* (as their nodes correspond to sets of arguments instead of individual arguments), which are appropriate for representing LAF-ensembles at the abstract level. In set-based argumentation frameworks, the attacks occur at the set level. The main differences between set-based frameworks and SETAFs are that the former allow attacks on sets of arguments and attacks where the source is the empty set; the latter are useful to capture inconsistencies of the theory at the language level (e.g., incompatible subsets of the language in Vreeswijk's AAS [106]).

Finally, it should be noted that, as also explained in Section 3.2 and shown in [93], ADFs are generalisations of SETAFs and can therefore model the type of collective attacks used in SETAFs. This is done by setting the following acceptance condition for each argument a: at least one argument from each of the sets of arguments attacking a should be rejected.

[16] For a more detailed discussion on team defeat, see [17]

4 Applications of joint attacks and models for joint supports

The ideas behind the characterisation of abstract argumentation frameworks with joint attacks, as those described in Section 3, have also been applied in other contexts. In this section we will focus on applications of joint attacks in Bipolar Argumentation Frameworks (BAFs) and argumentation frameworks with higher-order interactions[17].

Briefly, BAFs extend Dung's AF by incorporating a support relation intended to model a positive interaction between the elements it relates. The first works accounting for bipolarity in abstract argumentation conceived the support relation as a binary relation over the set of arguments in the framework (see [34; 37] for an overview on BAFs). However, later approaches adopted a different view of the support relation, to also account for *joint supports* (i.e., support relations whose source is a set of arguments) or, more generally, *higher-order supports* (i.e., support relations that can target other interactions, either attacks or supports), in addition to arguments.

In this section we will consider approaches to bipolar abstract argumentation that make use of joint attacks, joint supports or both. Finally, we will discuss the possibility of using joint attacks for modelling higher-order attacks and supports (i.e., interactions whose target is another interaction) and the generalised necessary support relation proposed in [88] and also accounted for in [31].

4.1 Flat bipolar argumentation frameworks with joint attacks or joint supports

In [91] the authors used the SETAF as the underlying framework for representing evidence against an argument in order to allow for evidence-based reasoning. They introduced the Evidential Argumentation System (EAS) which further extended the definition of SETAF by incorporating a specialised support relation to capture the notion of *evidential support*. The support relation in the EAS enables to distinguish between *prima-facie* and *standard* arguments; the former arguments do not require support from other arguments to stand, whereas the latter must be linked to at least one *prima-facie* argument through a chain of supports. Moreover, the prima-facie arguments are supported by a special argument η denoting support from the environment or the existence of supporting evidence. Also, analogously to the attack relation, the support relation in an EAS allows for supports to be originated on sets

[17] The latter are the subject of study in Chapter 1 of this handbook [29].

of arguments. Formally:

Definition 4.1. *An EAS is a tuple $\langle Ar, att, sup\rangle$, where Ar is a set of arguments, $att \subseteq (2^{Ar}\backslash\emptyset) \times Ar$ is the attack relation, and $sup \subseteq (2^{Ar}\backslash\emptyset) \times Ar$ is the support relation. A special argument $\eta \in Ar$ is distinguished, such that $\nexists(X, y) \in att$ where $\eta \in X$; and $\nexists X$ where $(X, \eta) \in att$ or $(X, \eta) \in sup$.*

The attack relation in an EAS is interpreted in the same way as the attack relation in the SETAF. Given $X \subseteq Ar$ and $a \in Ar$, $(X, a) \in att$ reads as follows: if all the arguments in X are accepted, then a cannot be accepted. In contrast, the evidential support relation is interpreted as follows. Given $X \subseteq Ar$ and $a \in Ar$, $(X, a) \in sup$ reads as: "the acceptance of a requires the acceptance of every argument in X".

Since the core idea of the EAS is that valid arguments (in particular, those originating attacks) need to trace back to the environment, the authors define the notion of evidence supported attack (e-supported attack). Then, based on this notion, semantics for the EAS have been characterized in [91] and then reformulated in [95], following Dung's methodology.

The Generalised Argumentation Frameworks with Necessities (GAFNs) [88] (directly referred to as AFNs in [87]) are another kind of bipolar argumentation frameworks that account for interactions between single arguments and sets of arguments but in a different way: a necessity relation between a set of arguments S and an argument a means that the acceptance of a requires the acceptance of *at least one* argument in S.

To illustrate the support relation of GAFNs, let us consider the following example. Suppose that in order to be awarded with a scholarship (s) a student is required to obtain a Bachelor's degree with honours (bh) or justify modest income (mi). In addition, suppose that the student has a bad mark (bm), and that having a bad mark prevents the student from obtaining the honours (regardless of the average of marks). We can represent this scenario by a GAFN with arguments s, bh, mi and bm. On the other hand, there exists an attack from bm to bh, and there exists a necessary support from the set $\{bh, mi\}$ to argument s. It is important to note that, even though the attack from bm to bh will result in bh not being accepted, this does not prevent s from being accepted (in other words, the student will obtain the scholarship). This is because the support towards s is originated in the set $\{bh, mi\}$, where each argument within this set provides an alternative condition for obtaining the scholarship.

Generalised Argumentation Frameworks with Necessities are formally defined as follows:

Definition 4.2. *A Generalised Argumentation Framework with Necessities (GAFN) is defined by a tuple* $\langle Ar, att, sup\rangle$ *where* Ar *is a set of arguments,* $att \subseteq Ar \times Ar$ *is an attack relation and* $sup \subseteq ((2^{Ar} \backslash \emptyset) \times Ar)$ *is a necessity relation.*

In [87] the author proposed a characterisation of semantics for the GAFN, in addition to those given in [88]. Finally, it should be noted that in [95] the authors provided a translation allowing the transformation of a GAFN into an EAS. Briefly, this translation is such that unsupported arguments in the GAFN will be arguments supported by η in the EAS; on the other hand, all sets of supporting arguments in the GAFN are combined into different sets of supporting arguments in the EAS by accounting for their Cartesian product. Finally, for the attack relation it suffices to map the attacking arguments in the GAFN into singleton sets of attacking arguments in the EAS. Then, [95] formally established a correspondence between the EAS and the GAFN in terms of their semantics, and identified correspondences between the properties of both frameworks and properties of Dung's AF.

Example 4.3. *Consider the GAFN* $\langle Ar, att, sup\rangle$*, where:*

- $Ar = \{a, b, c, d, e, f\}$
- $att = \{(b, a), (e, a), (c, d)\}$
- $sup = \{(\{b\}, e), (\{d, f\}, e), (\{a\}, d)\}$

This AFN could be translated into the EAS $\langle Ar', att', sup'\rangle$*, where:*

- $Ar' = Ar \cup \{\eta\}$
- $att' = \{(\{b\}, a), (\{e\}, a), (\{c\}, d)\}$
- $sup' = \{(\{b, d\}, e), (\{b, f\}, e), (\{a\}, d), (\{\eta\}, a), (\{\eta\}, b), (\{\eta\}, c), (\{\eta\}, f)\}$

For details about the characterisation of semantics for EAS and GAFN we refer the reader to [91] and [88; 87], respectively.

4.2 Bipolar argumentation frameworks with joint attacks or joint supports and higher-order interactions

The ideas adopted by the EAS and the GAFN described in the previous section were further exploited in [30] and [31], where the authors introduced the Recursive Evidence-Based Argumentation Framework (REBAF) and the Recursive Argumentation Framework with Necessity (RAFN). Briefly, these frameworks extend Dung's

AF by accounting for attack and support relations that can target not only arguments, but also attacks or supports at any level[18]. As a result, the REBAF adopts the evidential interpretation for the support relation of [91], whereas the RAFN adopts the generalised necessity interpretation of support proposed in [88]. The formal definitions of these frameworks are included below:

Definition 4.4. *A Recursive Evidence-Based Argumentation Framework (REBAF) is a tuple $\langle Ar, att, sup, \mathbf{s}, \mathbf{t}, PF\rangle$ where Ar, att and sup are pairwise disjoint sets respectively representing the names of arguments, attacks and supports, and $PF \subseteq Ar \cup att \cup sup$ is a set representing the* prima-facie *elements of the framework that do not need to be supported. The functions $\mathbf{s} : (att \cup sup) \mapsto 2^{Ar} \setminus \emptyset$ and $\mathbf{t} : (att \cup sup) \mapsto (Ar \cup att \cup sup)$ respectively map each attack and support to its source and its target.*

Definition 4.5. *A Recursive Argumentation Framework with Necessity (RAFN) is a tuple $\langle Ar, att, sup, \mathbf{s}, \mathbf{t}\rangle$, where Ar, att and sup are pairwise disjoint sets respectively representing the names of arguments, attacks and supports. The function $\mathbf{s} : (att \cup sup) \mapsto 2^{Ar} \setminus \emptyset$ and $\mathbf{t} : (att \cup sup) \mapsto (Ar \cup att \cup sup)$ respectively map each attack and support to its source and its target. It is assumed that $\forall \alpha \in att$, $\mathbf{s}(\alpha)$ is a singleton.*

Note that, according to Definition 4.4, attacks and supports in a REBAF can have a set of arguments as their source. In contrast, by Definition 4.5, the attack relation of a RAFN is restricted to only allow for arguments as the source of attacks. Then, in both cases, an attack or a support can also be the target of an interaction. Consequently, since these frameworks allow to reason about interactions in addition to arguments, the attacks and supports are also accounted for in the acceptability calculus.[19]

Semantics of REBAF and RAFN are defined using a notion of *structure*, defined as a triple $U = (S, \Gamma, \Delta)$ such that $S \subseteq Ar$, $\Gamma \subseteq att$ and $\Delta \subseteq sup$. Then, the notions of conflict-freeness, acceptability and admissibility as well as the subsequent semantics are defined over these structures, with the idea that the set S represents the set of "acceptable" arguments w.r.t. the structure U, and the sets Γ and Δ respectively represent the sets of "valid attacks" and "valid supports" w.r.t. U. For details about the definition of semantics for REBAF and RAFN, we refer the reader to [30] and [31], respectively, or to Chapter 1 of this handbook [29].

[18] The REBAF and the RAFN are studied in more detail in Chapter 1 of this handbook [29].

[19] This feature is also shared by other frameworks such as the AFRA and the ASAF, discussed in Section 4.3.

4.3 Using joint attacks to model higher-order interactions and generalised necessary supports

Gabbay [62] proposed the Higher-Level Argumentation Frames (HLAFs), which extend Dung's framework by allowing for attacks from arguments targeting not only arguments, but also attacks at any level. A HLAF can be defined as follows:

Definition 4.6. *Let Ar be a set of arguments.* Level $(0, n)$ argumentation frames *are defined as follows:*

1. *A pair $(a, b) \in Ar \times Ar$ is called a level $(0, 0)$ attack.*
2. *If $c \in Ar$ and α is a level $(0, n)$ attack then (c, α) is a level $(0, n+1)$ attack.*
3. *A level $(0, n)$ argumentation frame is the pair $\langle Ar, att\rangle$ where att contains only level $(0, m)$ attacks for $0 \leq m \leq n$.*

Note that, although the level of HLAFs is expressed in terms of pairs $(0, n)$ with possibly different values for n, the first component of the pair denoting the level is always 0 (the part of the level associated with the set of arguments). In particular, [62] proposed different kinds of approaches in order to define the semantics of HLAFs: the first option consists in translating a HLAF into a Dung's AF; the second alternative corresponds to the characterisation of labellings for HLAF, similarly to the labellings for AFs [6]; finally, in the third approach Gabbay proposed to translate a HLAF into a logic program. In the following, we will consider the first translation approach, which consists of obtaining a Dung's AF corresponding to a HLAF. Specifically, a HLAF $\langle Ar, att\rangle$ can be translated into an AF $\langle Ar^*, att^*\rangle$, where:

- $Ar^* = Ar \cup \{x_\beta, y_\beta \mid \beta = (a, \alpha) \in att\}$.
- $att^* = \{(a, x_\beta), (x_\beta, y_\beta), (y_\beta, \alpha) \mid \beta = (a, \alpha) \in att\}$.

The new arguments $x_{(a,\alpha)}$ and $y_{(a,\alpha)}$ associated with an attack from a to α respectively represent that the attack is 'live' or 'dead'; moreover, Gabbay argued that the translation of attacks as in the second bullet above is sufficient for attacks which are under attack. This translation is illustrated in Figure 7, where two attacks $\alpha = (a, b) \in att$ and $\beta = (c, \alpha) \in att$ are considered.

Then, given a set of extensions $E_1^+, E_2^+, \ldots, E_n^+$ of the associated AF, the corresponding extensions of the HLAF are $E_i^+ \cap Ar$, $(i = 1, \ldots, n)$.

In spite of proposing the translation described above, [62] argued that an attack $(a, b) \in att$ should be viewed as an independent unit, the attack of a on b, which can

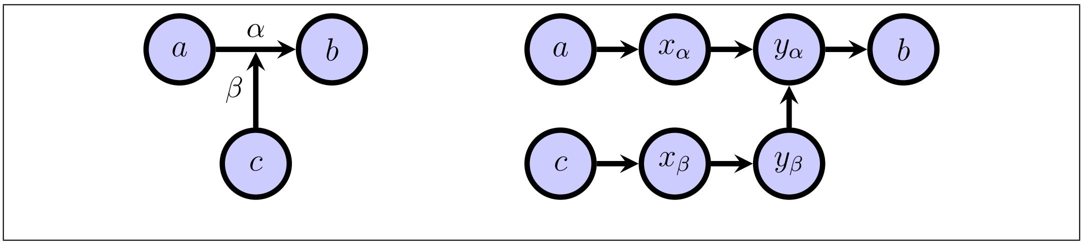


Figure 7: (LHS) HLAF with attacks $\alpha = (a, b)$ and $\beta = (c, \alpha)$; and (RHS) the translation into a series of AF attacks

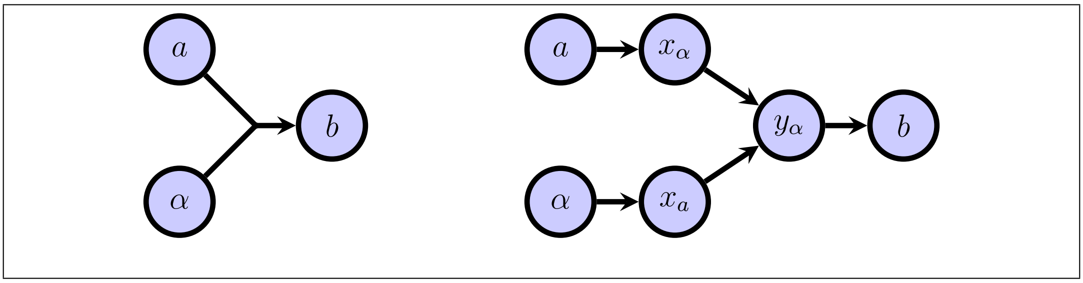


Figure 8: (LHS) Graphical representation of the joint attack by a and α on b, corresponding to an attack $\alpha = (a, b)$; and (RHS) its translation into a sequence of AF attacks.

be itself attacked. In particular, he stated that the preceding translation does not serve its purpose for modelling more general situations, such as attacks originated in other attacks (although the latter are not allowed in the frameworks of Definition 4.6). In that way, the author suggested that an attack $(a, b) \in att$ should be a unit kept 'live' unless attacked itself. Consequently, he proposed an alternative solution making use of *joint attacks*: an attack $\alpha = (a, b)$ is translated in a way such that the argument b is jointly attacked by two arguments a and α Then, both a and α must be 'live' in order for b to be 'dead'. The graphical representation of a joint attack by a and α on b, corresponding to an attack $\alpha = (a, b)$ is shown in Figure 8 on the left.

Given this notion of joint attack, [62] proposed a further translation of joint attacks into attacks in a Dung's AF. This translation has some similarities with the one introduced before for directly translating a HLAF into an AF, and is illustrated in Figure 8 on the right for the case of a joint attack by a and α on b.

Alternatively to the translation of joint attacks into attacks at the argument level in a Dung's AF, [62] introduced the *frames with joint attacks*:

Definition 4.7. *A* frame with joint attacks *has the form $\langle Ar, att\rangle$, where Ar is the set of arguments and $att \subseteq Ar \times Ar \times Ar$ is a ternary relation. We understand $(x, y, z) \in att$ as saying that the two arguments x and y are mounting a joint attack on z.*

The author remarked that single attacks can still appear in a frame with joint attacks; these would be attacks of the form $(x, x, y) \in att$. It is important to note that, following Definition 4.7, the frames with joint attacks are a particular case of the SETAFs, where the set of arguments originating an attack is restricted to a maximum of two elements. Consequently, the algorithms and reduction-based approaches for SETAFs discussed in Section 3.3 could also be applied to the frames with joint attacks.

Then, Gabbay introduced definitions analogous to those of [84], characterising the extensional semantics of these frameworks. Finally, he proposed a translation from HLAFs into frames with joint attacks, so that extensions of the former correspond to extensions of the latter.

Definition 4.8. *Let $\langle Ar, att\rangle$ be a HLAF. The corresponding frame with joint attacks $\langle Ar', att'\rangle$ is defined as follows:*

- $Ar' = Ar \cup att$
- $att' = \{(a, \alpha, \beta) \mid \alpha = (a, \beta) \in att\}$

Following this approach, for instance, the HLAF illustrated in Figure 9 on the top can be translated into a frame with joint attacks (or a SETAF) like the one depicted in Figure 9 at the bottom.

Next, we will discuss the possibility of using joint attacks for modelling attacks, including higher-order attacks, through Gabbay's Frames with Joint Attacks (a particular case of SETAFs) in frameworks such as the AFRA [7] or the ASAF[20] [68]. We will start by briefly recalling the definition of these frameworks, as proposed by their authors. As mentioned before, these frameworks are studied in another chapter of this book. Thus, for more details, we refer the interested reader to Chapter 1 of this handbook [29].

The Argumentation Framework with Recursive Attacks (AFRA) [7] generalises Dung's AF by incorporating a *recursive attack relation* where attacks are allowed to target other attacks as well as arguments, and the attacks can occur at any level.

Definition 4.9. *An* Argumentation Framework with Recursive Attacks (AFRA) *is a pair $\langle Ar, att\rangle$ where:*

[20] In the case of the ASAF, initially, without supports (where such an ASAF would be an AFRA). The means for modelling supports through joint attacks will be discussed later.

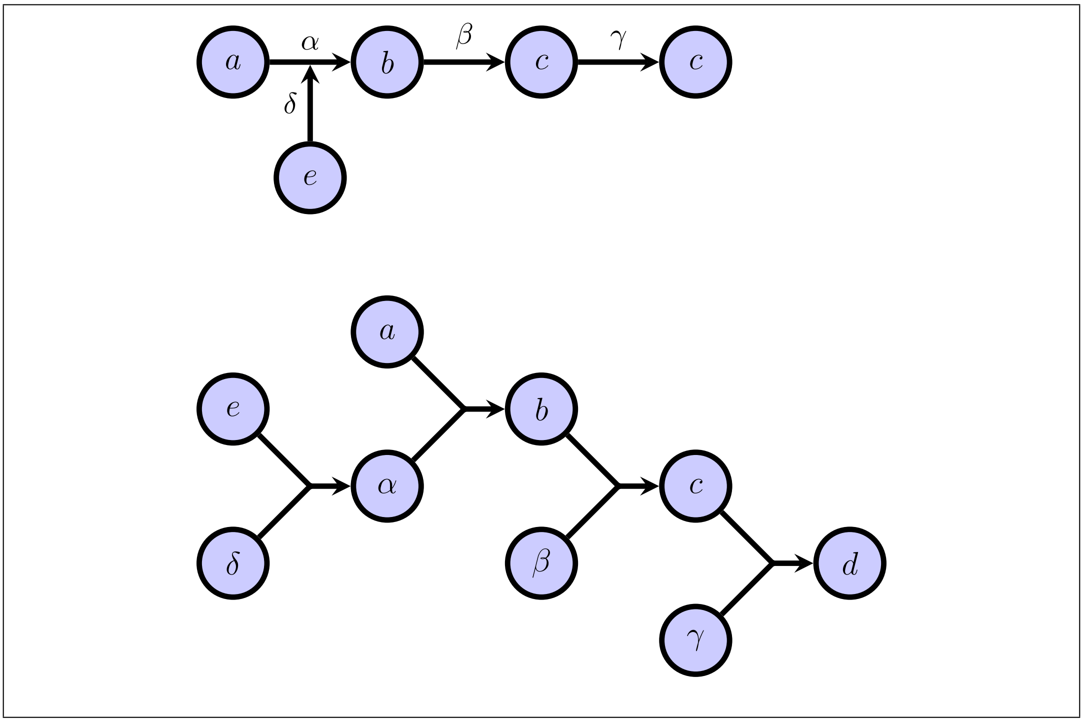


Figure 9: (Top) A HLAF with a higher-order attacks, where Greek letters denote the labels of the attacks; and (Bottom) its corresponding frame with joint attacks

- *Ar is a set of arguments;*
- *att is a set of attacks, namely pairs (a, X) such that $a \in Ar$ and ($X \in Ar$ or $X \in att$).*

Given an attack $\alpha = (a, X) \in att$, a is said to be the source of α, denoted as $\mathbf{s}(\alpha) = a$, and X is the target of α, denoted as $\mathbf{t}(\alpha) = X$. Moreover, similarly to the notation used before for Gabbay's HLAF, [7] introduces an abbreviated notation for recursive attacks in the AFRA; for instance, an attack $(c, (a, b))$ can be expressed as (c, α), where $\alpha = (a, b)$.

Then, [7] establishes the different kinds of defeat that can occur between the elements of an AFRA. A key aspect of their formalisation is that they regard attacks (not their source arguments) as the subjects able to defeat arguments and other attacks. Then, an attack can be made ineffective (in other words, defeated) either by attacking the attack itself or by attacking its source. The notions of *direct defeat* and *indirect defeat* are introduced in [7] as follows:

Definition 4.10. *Let $\langle Ar, att\rangle$ be an AFRA, $\alpha \in att$ and $X \in Ar \cup att$. It is said that α defeats X, denoted $\alpha \rightarrow^R X$ if one of the following conditions holds:*

- $\mathbf{t}(\alpha) = X$ *(direct defeat); or*
- $X = \beta \in att$ *and* $\mathbf{t}(\alpha) = \mathbf{s}(\beta)$ *(indirect defeat).*

Then, based on this notion of defeat, the notions of conflict-freeness, acceptability, admissibility and extensions under different semantics are introduced following Dung's methodology. Consequently, the extensions of an AFRA will not only contain the accepted arguments under the corresponding semantics, but also the accepted attacks.

Example 4.11. *The arguments and attacks depicted at the top in Figure 9 correspond to the AFRA $\langle Ar, att\rangle$, where $Ar = \{a, b, c, d, e\}$ and $att = \{\alpha, \beta, \gamma, \delta\}$, with $\mathbf{s}(\alpha) = a$, $\mathbf{t}(\alpha) = b$, $\mathbf{s}(\beta) = b$, $\mathbf{t}(\beta) = c$, $\mathbf{s}(\gamma) = c$, $\mathbf{t}(\gamma) = d$, $\mathbf{s}(\delta) = e$, $\mathbf{t}(\delta) = \alpha$.*

Here, the direct defeats are: $\alpha \rightarrow^R b$, $\beta \rightarrow^R c$, $\gamma \rightarrow^R d$ and $\delta \rightarrow^R \alpha$. On the other hand, the indirect defeats are: $\alpha \rightarrow^R \beta$ and $\beta \rightarrow^R \gamma$. Consequently, β reinstates d, α reinstates c and γ, and δ reinstates α and b. As a result, for instance, the AFRA has only one complete extension (which is also its grounded and only preferred and stable extension), namely $\{a, e, \delta, b, \beta, d\}$. In contrast, if we apply the SETAF semantics on the framework depicted at the top on Figure 9[21]*, we have that the only complete extension is $\{a, e, \delta, b, \beta, \gamma, d\}$.*

The difference in the result obtained by applying the AFRA semantics, compared to the one obtained by translating the AFRA into a SETAF and then applying the SETAF semantics, has to do with the fact that the translation proposed in [62] does not take into account the indirect defeats. In particular, in the above example, the indirect defeat by β on γ is not captured, leaving γ as an accepted attack[22]. This suggests that we need to establish a different translation of AFRAs into SETAFs, in order to account for the effect of indirect defeats. An alternative translation of an AFRA into a SETAF could be:

Definition 4.12. *Let $\langle Ar, att\rangle$ be an AFRA. The corresponding SETAF $\langle Ar, \triangleright\rangle$ is defined as follows:*

$$\begin{aligned} Ar = \quad & Ar \cup att \\ \triangleright = \quad & \{(\{a, \alpha\}, X) \mid \alpha = (a, X) \in att\} \cup \\ & \{(\{a, \alpha\}, \alpha') \mid \alpha = (a, X) \in att,\ \alpha' \in att,\ \mathbf{s}(\alpha') = X\} \end{aligned}$$

[21] As stated before, the frames with joint attacks are a particular case of SETAFs.

[22] The indirect defeat by α on β is not captured either; however, since α is not accepted (because it is directly defeated by δ) it does not affect the outcome.

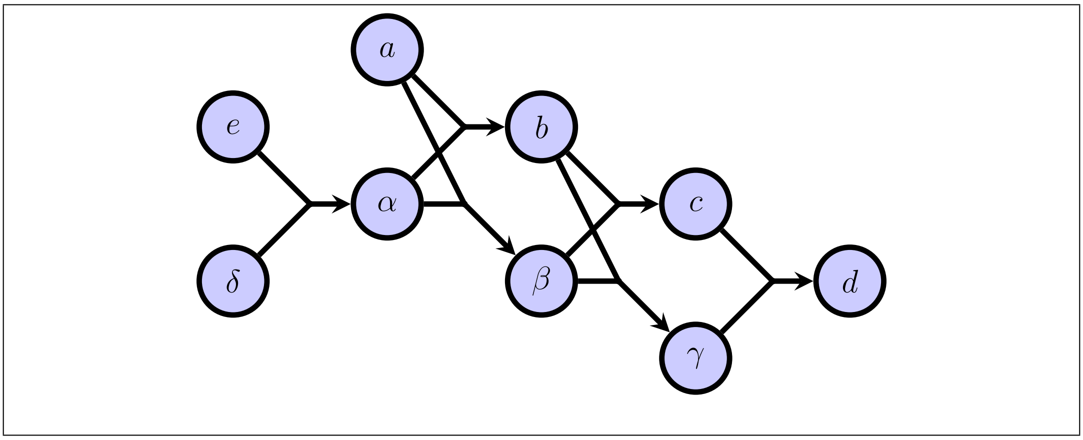


Figure 10: SETAF corresponding to the AFRA from Example 4.11

The AFRA from Example 4.11, corresponding to the framework depicted at the top of Figure 9, can be translated following Definition 4.12 to obtain the SETAF depicted in Figure 10. Then, applying the SETAF semantics on that framework, the only complete extension coincides with the one obtained with the AFRA semantics in Example 4.11.

Let us now consider the formalization of the Attack-Support Argumentation Framework (ASAF) [68]. Briefly, the ASAF extends Dung's AF by incorporating bipolar higher-order interactions. In that way, the ASAF allows for the representation and reasoning with attack and support relations not only between arguments, but also targeting the attack and support relations themselves. In particular, the support relation of the ASAF is interpreted as necessity [88]. That is, the necessary support relation in the ASAF imposes the following acceptability constraints on the elements it relates: if a supports b, then the acceptance of b implies the acceptance of a; equivalently, the non-acceptance of a implies the non-acceptance of b. Note that the support relation in the ASAF is set to be binary, differently from the necessary support relation of the GAFN introduced in Section 4.1. Some of the following definitions are taken from [2], where the background for the ASAF was succinctly introduced.

Definition 4.13. *An* Attack-Support Argumentation Framework (ASAF) *is a tuple* $\langle Ar, att, sup \rangle$ *where* Ar *is a set of arguments,* $att \subseteq W$ *is the attack relation, and* $sup \subseteq W$ *is the support relation, with* W *being the set iteratively defined as follows:* $W = Ar \times Ar$ *(basic step) and* $W = Ar \times W$ *(iterative step). It is assumed that* sup *is acyclic and* $att \cap sup = \emptyset$.

Similarly to the case of the AFRA, an attack $(a, b) \in \mathit{att}$ will be denoted as $\alpha_1 = (a, b)$; analogously, a support $(b, c) \in \mathit{sup}$ will be denoted as $\beta_1 = (b, c)$. Then, for instance, an attack from d to α_1 will be denoted as $\alpha_2 = (d, \alpha_1)$. In general, given an attack $\alpha = (a, X) \in \mathit{att}$, a is called the source of α, denoted $\mathbf{s}(\alpha) = a$, and X is called the target of α, denoted $\mathbf{t}(\alpha) = X$. Analogously, given a support $\beta = (b, Y) \in \mathit{sup}$, b is called the source of β, denoted $\mathbf{s}(\beta) = b$, and Y is called the target of β, denoted $\mathbf{t}(\beta) = Y$.

Like in the AFRA, different kinds of defeat that can occur between the elements of an ASAF. Specifically, they correspond to the two kinds of defeat identified for the AFRA, plus two additional kinds of defeat that arise from the coexistence of the attack and support relations.

Definition 4.14. *Let $\Delta = \langle Ar, \mathit{att}, \mathit{sup}\rangle$ be an ASAF, $\alpha \in \mathit{att}$, $X \in (Ar \cup \mathit{att} \cup \mathit{sup})$ and $\mathbf{S} \subseteq \mathit{sup}$. We say that α defeats X (given $\mathbf{S}$), denoted α def X given $\mathbf{S}$ (or simply α def X whenever $\mathbf{S} = \emptyset$) iff one of the following conditions holds:*

- *there exists a (possibly empty) support path from $\mathbf{t}(\alpha)$ to X, whose corresponding set of supports is $\mathbf{S}$; or*
- *$X \in \mathit{att}$ and there exists a (possibly empty) support path from $\mathbf{t}(\alpha)$ to $\mathbf{s}(X)$, whose corresponding set of supports is $\mathbf{S}$.*

To illustrate these notions, let us consider the following example. Similarly to Dung's AF or the AFRA, an ASAF can be graphically represented using a graph-like notation where two kinds of edges are considered: $\rightarrow$ for the attack relation and $\Rightarrow$ for the support relation. In addition, attacks and supports are labelled with greek letters, following the convention that attacks are labelled with α (possibly with subscripts) and supports are labelled with β (again, possibly with subscripts).

Example 4.15. *Consider the ASAF $\langle Ar, \mathit{att}, \mathit{sup}\rangle$, where $Ar = \{a, b, c, d, e, f, g, h\}$, $\mathit{att} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ and $\mathit{sup} = \{\beta_1, \beta_2, \beta_3\}$, with $\alpha_1 = (a, b)$, $\alpha_2 = (c, e)$, $\alpha_3 = (g, f)$, $\alpha_4 = (h, \alpha_3)$, $\beta_1 = (b, c)$, $\beta_2 = (c, d)$ and $\beta_3 = (f, \beta_1)$. This framework is depicted in Figure 11, and the following defeats occur: α_1 def b, α_2 def e, α_3 def f, α_4 def α_3, α_1 def c given $\{\beta_1\}$, α_1 def α_2 given $\{\beta_1\}$, α_1 def d given $\{\beta_1, \beta_2\}$, α_3 def β_1 given $\{\beta_3\}$.*

The semantics of the ASAF are also defined following Dung's methodology, accounting for the notions of conflict-freeness, acceptability and admissibility, to later characterise the complete, preferred, stable and grounded semantics of the framework. It should be noted that, since the defeats in the ASAF may involve a set of supports, these notions cannot be directly defined by considering the definitions for AFs (see Section 3.1 and Definition 4.14).

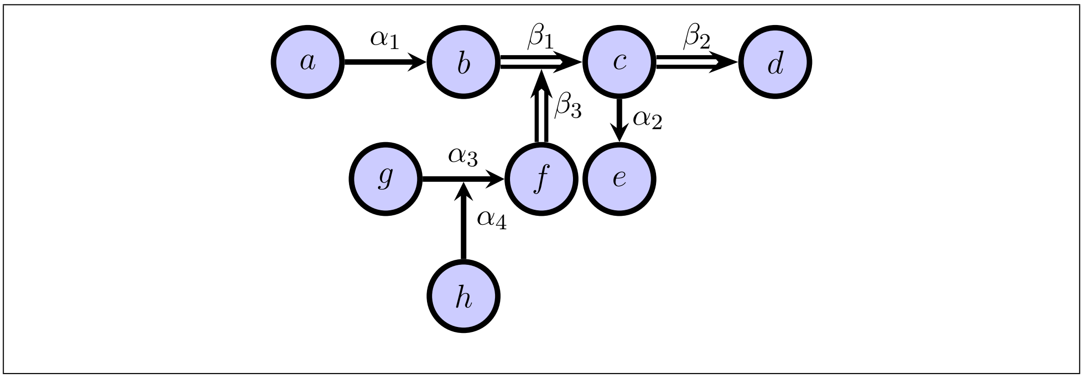


Figure 11: ASAF from Example 4.15

Definition 4.16. *Let $\Delta = \langle Ar, att, sup\rangle$ be an ASAF and $\mathbf{S} \subseteq (Ar \cup att \cup sup)$.*

- $\mathbf{S}$ *is* conflict-free *iff* $\nexists \alpha, X \in \mathbf{S}$, $\nexists \mathbf{S}' \subseteq (\mathbf{S} \cap sup)$ *such that* α def X given $\mathbf{S}'$.
- $X \in (Ar \cup att \cup sup)$ *is* acceptable w.r.t. $\mathbf{S}$ *iff* $\forall \alpha \in att$, $\forall \mathbf{T} \subseteq sup$ *such that* α def X given $\mathbf{T}$: $\exists Y \in (\{\alpha\} \cup \mathbf{T})$, $\exists \alpha' \in \mathbf{S}$, $\exists \mathbf{S}' \subseteq (\mathbf{S} \cap sup)$ *such that* α' def Y given $\mathbf{S}'$.
- $\mathbf{S}$ *is* admissible *iff it is conflict-free and for all* $X \in \mathbf{S}$, X *is acceptable w.r.t.* $\mathbf{S}$.

Definition 4.17. *Let $\Delta = \langle Ar, att, sup\rangle$ be an ASAF and $\mathbf{S} \subseteq (Ar \cup att \cup sup)$.*

- $\mathbf{S}$ *is a* complete extension *of* Δ *iff it is an admissible set and* $\forall X \in (Ar \cup att \cup sup)$, *if* X *is acceptable w.r.t.* $\mathbf{S}$, *then* $X \in \mathbf{S}$.
- $\mathbf{S}$ *is a* preferred extension *of* Δ *iff it is a maximal (w.r.t.* $\subseteq$*) complete extension of* Δ.
- $\mathbf{S}$ *is a* stable extension *of* Δ *iff it is a complete extension of* Δ *and* $\forall X \in (Ar \cup att \cup sup) \backslash \mathbf{S}$, $\exists\, \alpha \in \mathbf{S}$, $\exists\, \mathbf{S}' \subseteq (\mathbf{S} \cap sup)$ *such that* α def X given $\mathbf{S}'$.
- $\mathbf{S}$ *is the* grounded extension *of* Δ *iff it is the smallest (w.r.t.* $\subseteq$*) complete extension of* Δ.

The ASAF from Example 4.15 has only one complete extension, which is also the grounded extension and the only preferred and stable extension of the framework: $\{a, e, f, g, h, \alpha_1, \alpha_4, \beta_1, \beta_2, \beta_3\}$. In particular, it can be noted that whereas α_3 def β_1 given $\{\beta_3\}$, the support β_1 is reinstated by α_4, since α_4 def α_3. Then, the defeats from α_1 on c and α_2 given $\{\beta_1\}$ are also reinstated, as well as the defeat from α_1 on d given $\{\beta_1, \beta_2\}$.

As shown in [68], an ASAF without support is an AFRA. So, when applying the AFRA semantics on ASAFs without supports we obtain the same outcome as the one obtained under the ASAF semantics. Consequently, the translation of an AFRA into a SETAF could also be applied to the ASAF; nevertheless, some adjustments need to be made in order to account for the defeats involving a set of supports. A possible translation of an ASAF into a SETAF is given below.

Definition 4.18. *Let $\langle Ar, att, sup\rangle$ be an ASAF. The corresponding SETAF $\langle Ar, \triangleright\rangle$ is defined as follows:*

$$\begin{array}{rl} Ar = & Ar \cup att \cup sup \cup \{\beta^* \mid \beta \in sup\} \\ \triangleright = & \{(\{a,\alpha\}, X) \mid \alpha = (a, X) \in att\} \cup \\ & \{(\{a,\alpha\}, \alpha') \mid \alpha = (a, X) \in att,\ \alpha' \in att,\ \mathbf{s}(\alpha') = X\} \cup \\ & \{(\{a,\alpha\}, X^*) \mid \alpha = (a, X) \in att,\ X \in sup\} \cup \\ & \{(\{a,\beta\}, \beta^*), (\{\beta^*\}, X) \mid \beta = (a, X) \in sup\} \cup \\ & \{(\{\beta^*\}, X^*) \mid \beta \in sup,\ X \in sup, \mathbf{t}(\beta) = X\} \cup \\ & \{(\{\beta^*\}, \alpha) \mid \beta \in sup,\ \alpha \in att, \mathbf{t}(\beta) = \mathbf{s}(\alpha)\} \end{array}$$

The ASAF from Example 4.15, corresponding to the framework depicted in Figure 11, can be translated following Definition 4.18 to obtain the SETAF depicted in Figure 12. Then, applying the SETAF semantics on that framework, the only complete extension coincides with the one obtained with the ASAF semantics, namely $\{a, e, f, g, h, \alpha_1, \alpha_4, \beta_1, \beta_2, \beta_3\}$.

Finally, we will briefly discuss the possibility of using joint attacks to model the generalised necessity relation proposed in [88] adopted in frameworks such as the RAFN (see Section 4.2).

Let us recall the example introduced in Section 2, represented using the SETAF from Figure 2. There, we can think of the argument NP as providing a context under which $A18$ attacks M; that is, a person aged under 18 is not allowed to marry whenever parent permission is not provided. So, we could think of this situation as corresponding to the existence of an attack $\alpha_1 = (A18, M)$ and a generalised necessary support $\beta_1 = (\{NP\}, \alpha_1)$. Similarly, given the restriction to drink alcohol, arguments NA and NM can be considered as providing alternative contexts under which $A18$ attacks Alc. Therefore, we could think of representing this situation through an attack $\alpha_2 = (A18, Alc)$ and a generalised necessary support $\beta_2 = (\{NA, NM\}, \alpha_2)$. This is because, in this situation, it suffices to have either NA or NM accepted in order to be able to accept α_2 (i.e., in order for the attack from $A18$ to Alc to hold).

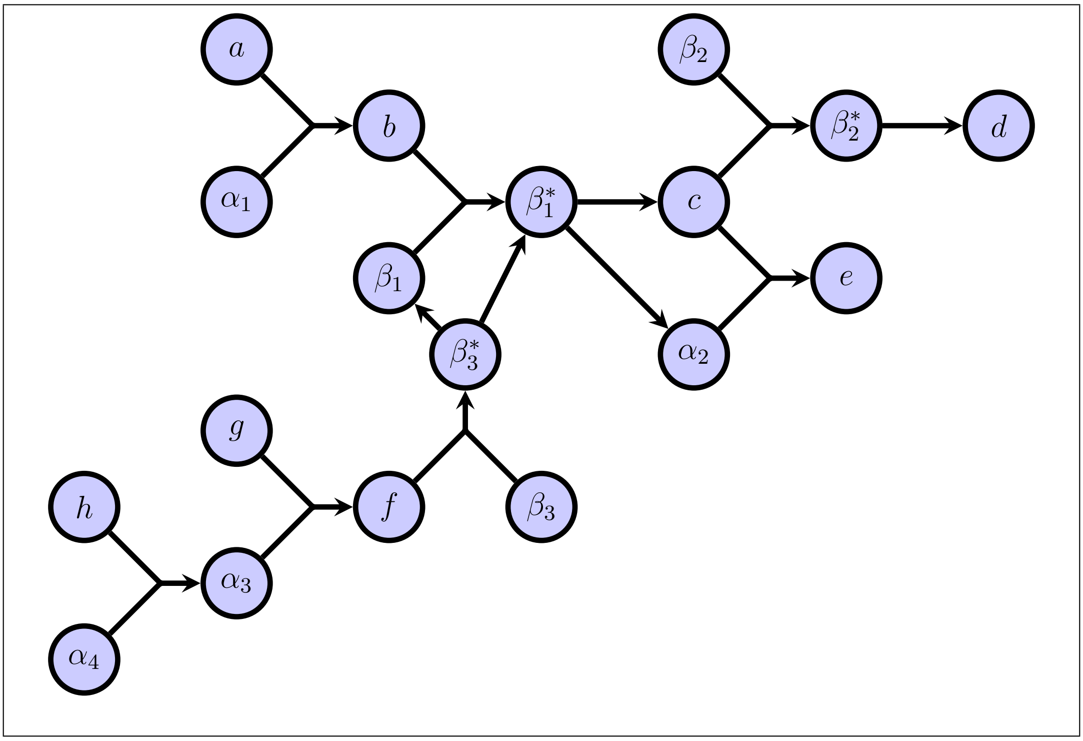


Figure 12: SETAF corresponding to the ASAF from Example 4.15

The preceding example suggests that joint attacks (as those in the SETAF from Figure 2) may be suitable for modelling the generalised necessary support relation in the case of higher-order supports targeting an attack. However, for instance, if there exists another interaction (say, an attack α_3) targeting β_1, we should be able to model on the SETAF the fact that if α_3 is accepted then β_1 no longer holds and, consequently, that NP does not provide a context under which the attack α_1 from $A18$ to M holds. Nevertheless, if the support β_1 from NP to α_1 is modeled by a joint attack from NP and $A18$ as in Figure 2, we cannot model the attack from α_3 towards β_1 in the SETAF, since β_1 is not made explicit in this representation.

5 Accrual

A parallel line of research in computational argumentation studies the accrual of arguments, i.e., how arguments supporting or refuting the same claim can be combined. The main differences between accrual and joint attacks (at least under the type of joint attacks used in SETAFs) is that, in joint attacks, the strength of an argument or a combination of arguments is not considered when evaluating the effectiveness of attacks, and each argument participating in a joint attack is an essential element of it (in other words if an argument is missing then the attack is ineffective), while in accrual the strength of each argument is taken into account, and adding an argument to an accrual makes the accrual stronger, or more generally it changes its strength and the effectiveness of its attacks (or supports).

A seminal study on the accrual of arguments [96] set out three principles for accrual:

1. An accrual is sometimes weaker than its accruing elements. This is due to the possibility that the accruing reasons are not independent.

2. An accrual makes its elements inapplicable. More generally, any ‘larger’ accrual that applies makes all its ‘lesser’ versions inapplicable. This is because an accrual is meant to consider all available information, while the individual arguments it consists of take only part of the information into account.

3. Flawed reasons or arguments may not accrue. Any treatment of accrual should capture that when an individual reason or argument turns out to be flawed, it does not take part in the accrual.

Prakken also described two general ways to formalise accrual: (a) the *knowledge representation* (or else *KR*) approach, which requires formulating a separate rule for each possible combination of the accruing reasons; (b) the *inference* approach, where the accrual is part of the inference process, i.e., after all individual reasons have been constructed, those that attack or support the same claim are somehow aggregated and some mechanism is then used to resolve any conflicts between the conflicting sets of reasons. He also proposed a formalisation of accrual using the inference approach, according to which the conclusion of each individual defeasible inference step is labelled with the premises of the applied defeasible inference rule:

$$\phi, \phi \Rightarrow \psi \mathrel{|\!\!\sim} \psi^{\{\phi, \phi \Rightarrow \psi\}}$$

and a new defeasible inference rule is introduced that takes any set of labelled versions of a certain formula and produces the unlabelled version:

$$\phi^{l_1}, \cdots, \phi^{l_n} \mathrel{|\!\!\sim} \phi$$

The attack relationships among arguments are adjusted as follows: rebuttal requires that the two arguments support opposite conclusions that are labelled in the same way, while undercut requires that the attacking arguments have unlabelled conclusions. Finally, the following rules ensure that when a set of reasons accrues, any subset of it is inapplicable:

$$\phi^{l_1}, \cdots, \phi^{l_n} \mathrel{|\!\!\sim} \neg\lceil \phi^{l_1}, \cdots, \phi^{l_{n-1}} \rceil$$

The proposed formalisation satisfies all principles of accrual, but has a computational drawback: it requires considering all possible accruals for every conclusion, which may lead to an exponential increase in the number of arguments.

The idea of combining arguments for and against a claim, albeit under the name "aggregation" rather than "accrual" was studied by argumentation researchers before [96]. One line of work, that of Fox and colleagues, goes back at least as far as [90], where the idea of symbolically weighing evidence is formalised in what recognisably is a structured argumentation framework, and arguably as far back as [61] where the idea was first applied. The formal development of that work came to a conclusion with [72] and [59]. The former paper describes a model that links the simple form of accrual from [90], which effectively just looks at the numbers or arguments for and against a claim[23], with forms of accrual which connect to probabilistic models. The latter uses the same model to develop a hierarchy of notions of acceptability, coming close to Dung's work at about the same time that work was first published [39].

CumulA [103; 104] was another structured argumentation model that dealt with accrual. Additionally to the notion of *compound defeaters*, which we discussed in Section 3.4, it also includes the notions of *coordination* and *narrowings* of arguments. Different arguments supporting the same conclusion can be combined in a *coordinated* argument, while the *narrowing* of a coordinated argument a is an argument b supporting the same conclusion as a but containing a subset of the arguments combined in a (or narrowings of them). CumulA deals with accrual using compound defeaters and the following acceptability condition for arguments: if the narrowing of an argument a is `in` (meaning that the argument is accepted), then a should be `in` too. As shown in [96], CumulA satisfies all three principles of accrual but the second one (i.e. that an accrual makes its elements inapplicable) is satisfied in a way that is too strong. Because of the acceptability condition described above, which implies that if an accrual is `out` then all its narrowings are `out`, it cannot capture a situation where an accrual is defeated because of subargument defeat so that some of its narrowings can be undefeated.

More recently, [16] developed another account of accrual, based on their logic-based approach to argumentation, though again they do not describe it as such. In

[23] Before dismissing such a simple model, consider how effective such simple models can be [38].

[16], lines of discussion about a particular claim — the argument for it, the arguments against it, the arguments against those arguments, and so on — are brought together into an argument tree. Then, all argument trees for or against a claim are assembled into an argument structure. An argument structure thus gathers everything that is relevant to whether or not a claim should be accepted. This, of course, is not much different to what one would get from assembling all of the arguments in a structured framework like ASPIC+ or DeLP that bear on a specific formula into some super-structure. However, whereas most structured frameworks summarise this higher level structure in a notion of acceptability, [16] defines a "categoriser" which maps a structure to a number, and this can be thought of as the accrued value of the set of arguments in the structure.

[78] proposed an approach for formalising the accrual of arguments in Defeasible Logic Programming using the notion of *a-structure*, a special kind of argument which subsumes different chains of reasoning that provide support for the same conclusion, and *partial attacks* among a-structures, where the attacking a-structure generally affects only the narrowing of the attacked a-structure containing exactly the arguments affected by the conflict. A binary preference relation on a-structures is used to determine the relevant strength of conflicting a-structures and whether an attack succeeds (in which case it constitutes a *defeat*). To deal with combined attacks (situations where two or more a-structures simultaneously attack the same a-structure), they define a process, called *bottom-up sequential degradation*, according to which the defeats are applied in sequence with the "deeper" ones applied first. The described framework satisfies all three principles of accrual. Its main difference with the formalisation proposed in [96] is that when analysing a theory to determine the accepted (undefeated) a-structures, it only considers maximal accruals (a-structures) and not all possible accruals for a conclusion.

[67] proposed the use of argument weighing functions as a way to model different types of argument schemes, including some types of argument accrual, in Carneades, a structured argumentation framework. In this framework, an argument is defined as a tuple (s, P, c, u) where s is the scheme that the argument instantiates; P, the premises of the argument, is a finite subset of the underlying logical language $\mathcal{L}$; and c, its conclusion, and u, its undercutter, are elements of $\mathcal{L}$. Its semantics is defined in terms of a labelling, which assigns a value from $\{\texttt{in}, \texttt{out}, \texttt{undec}\}$ to each element of $\mathcal{L}$ and a weighing function, which assigns a value from $[0, 1]$ to each argument and 0 to all arguments such that their undercutter is `in`. Gordon also provided several examples of weighing functions, some of which are appropriate for modelling different types of accrual. To simulate *convergent arguments*, i.e., arguments that at least one of its premises must be `in` to support their conclusion, he defined a weighing function that assigns 1 to an argument if at least one of its premises is `in`

and its undercutter is not in, and 0 otherwise. A weighing function that simulates *cumulative arguments*, i.e., arguments whose strength increases with the number of their acceptable premises, assigns the percentage of the premises of the argument that are in to every argument whose undercutter is not in. Cumulative arguments are a special type of accrual that does not satisfy Prakken's second principle, since cumulation can only increase the strength of an argument. Another weighing function that simulates accrual takes into account all *factors* (statements) that need to be considered when evaluating the arguments for a certain issue. It does so by assigning to each argument the proportion of its factors that are premises of the argument and are labelled in. One limitation of this framework with respect to accrual is that although it handles various forms of accrual at the level of statements, e.g., premises or factors of an argument, it does not provide a way to handle accrual of multiple arguments.

[97] proposed a formalisation of accrual for ASPIC+, a structured argumentation framework where arguments are tree-like structures constructed from a knowledge base, which is a subset of an underlying logical language $\mathcal{L}$, and a set of inference, strict or defeasible, rules. In this framework, there are two ways to attack an argument a: either at the top inference rule r of a (*undercut*) or at the conclusion of r (*rebuttal*)[24] - in both cases r must be defeasible, otherwise a cannot be attacked. [97] extended ASPIC+ with the notion of *accrual sets*, which are defined relative to a labelling of the set of arguments S. An accrual set for a literal $\phi \in \mathcal{L}$, denoted as $s_l(\phi)$, is the set of arguments with conclusion ϕ satisfying the following two conditions: (*i*) for any argument in $s_l(\phi)$ no immediate subargument of a is out and no undercutter of a is in; (*ii*) any argument with conclusion ϕ whose undercutters are out and its immediate subarguments are in must be in $s_l(\phi)$. The extended framework also includes a preference relation $\leq$ on the power set of S, such that any set of arguments containing a strict argument is at least as preferred as every other subset of S. It also includes a new defeat relation on arguments, called *l-defeat*, which takes into account accruals: an argument a l-defeats an argument b iff a undercuts b; or a rebuts b, and for some accrual sets for the conclusions of a, $s_l(Conc(a))$, and b, $s_l(Conc(b))$, it holds that $s_l(Conc(a)) \not< s_l(Conc(b))$. A *characteristic function* F is used to compute the labelling of a framework, which satisfies the following conditions: an argument a is in iff all arguments that l-defeat a are out and all immediate subarguments of a are in; a is out iff it is defeated by an argument that is labelled in or one of its immediate subarguments are out.

[24] Note that these definitions are different from the standard ASPIC+ where arguments can also be attacked on their subarguments. As explained in [97], in the version of ASPIC+ considered in this paper, arguments are constructed recursively and the recursion takes care of subargument attacks.

The proposed framework satisfies all principles of accrual and preserves some of the properties of Dung's AFs, such as the existence of complete and preferred labellings and the relations between grounded, complete, stable and preferred semantics.

In the field of abstract argumentation, the most relevant approaches are the frameworks with graded semantics (e.g., see [11] for an overview and a study of their properties) or ranking semantics and social argumentation frameworks (e.g., see [74; 12; 92]). Such frameworks provide methods for assessing the strength of an argument based on the aggregate strength of its attackers and the aggregate strength of its supporters (and in some cases the initial valuation of the argument), capturing the main idea of accrual. Some of their general properties are: (*i*) the larger the set of the attackers on an argument, the lower the strength of the argument under attack; (*ii*) the larger the set of supporters or defenders of an argument, the higher the strength of the argument they support or defend; and (*iii*) an argument with 0 strength does not have an effect on the strength of the arguments it attacks or supports. The last property satisfies the third principle of accrual (i.e., that flawed arguments do not accrue), while by considering the aggregate strength of the attackers or supporters of an argument, they essentially satisfy the second principle, i.e., that an accrual makes its elements inapplicable. Properties (*i*) and (*ii*), however, violate the second principle, since they imply that an accrual is always stronger than the individual accrued arguments.

6 Proposals for future work on joint attacks

In this section we highlight some emerging topics for future research on joint attacks and accrual.

There are several interesting directions for further research concerning semantics of SETAFs. Standard semantics of AFs have been generalised to SETAFs and their basic properties and relations are settled. However, several prominent semantics have not yet been generalised and analysed on SETAFs, e.g., cf2 [8], strong admissibility [9; 26] and weak admissibility [14]. Recently, a first approach to transfer also ranking-based semantics to SETAFs has been undertaken [108]. Properties of AF semantics have been studied in versatile aspects [102; 13] beyond the existing analysis for SETAFs. For instance, generalising the principle-based approach for analysing and comparing semantics to SETAFs would be valuable for the selection of the right argumentation semantics, and understanding the different notions of equivalence also on SETAFs is fundamental for using SETAFs in dynamic settings. Concerning the latter, a first investigation of strong equivalence notions for SETAFs has been done in [54].

As another research direction one could consider enhancing the expressiveness of SETAF by extending its basic model with features similar to the ones used in extensions of the AF model, such as the introduction of a joint support relation, weights on (joint) attacks, values promoted by (sets of) arguments, or a preference relation among (sets of) arguments. This would allow associating SETAFs with the corresponding AF extensions, i.e., frameworks for bipolar argumentation [3; 32], graded [71] or weighted argumentation [46], value-based [15], or preference-based argumentation [4] respectively. A related but somehow orthogonal research direction is the investigation of the relations of SETAFs and other extensions of AFs concerning their expressiveness. Existing investigations in that direction are the embedding of SETAFs in ADFs [77] and translations between SETAFs and claim-augmented AFs [56].

The translations from SETAF to AFs discussed in Section 3.2 either had the weakness that they might increase the size exponentially or only supported a selection of the semantics. For future research one could investigate alternative translation schemes in order to avoid this pitfall. Ideally, we would like to have a transformation that applies for all semantics and causes a polynomial increase in the size of the framework (in the sense of [20]), while at the same time resulting in elegant correspondences for all the semantics (unlike [60], where this is true only for some of the semantics). Recall, that [94] provide a translation that satisfies the latter two properties but only for a selection of the semantics, i.e., the semantics based on complete extensions, with the exception of semi-stable semantics. Also notice that a polynomial increase in the size of the framework can still result in an exponential increase in the number of arguments. Thus, another open question is whether such a potential exponential increase in the number of arguments can be avoided.

On the computational side there are several open challenges. From the theoretical perspective one would be interested in identifying classes of instances that provide milder complexity than general SETAFs. One approach that has been extensively studied for AFs are the so called tractable fragments [42], i.e., special graph classes like acyclic or bipartite, that allow for efficient reasoning procedures. A more general approach are graph parameters and techniques for parametrised complexity theory that allow for algorithms which are only exponential w.r.t. a graph parameter but polynomial in the size of the AF [51; 52]. From a more practical view one would be interested in efficient labelling-based algorithms [86; 36] for SETAFs as well as systems that extend methods that have been successfully applied for AFs [35]. An important step to boost the development of such systems would be to establish standard formats to share SETAF instances and standard benchmark sets.

Regarding the application of joint attacks, the ideas discussed in Section 4.3 could be further explored. As shown in Section 4.3, the translations from an AFRA [7] or an ASAF [68] into a SETAF yield the same outcomes as those obtained directly by applying the AFRA or ASAF semantics, respectively. However, this was only shown for the examples illustrated in that section. A formal analysis of this correspondence for the general case of an arbitrary ASAF or AFRA is left for future work. In addition, the brief discussion at the end of Section 4.3 can also be the subject of future work, also considering the translations discussed in Section 3.2. Specifically, studying the possibility of using the SETAF for modelling the generalised support relation of frameworks like the RAFN [30; 31], accounting for all cases: first-order supports, higher-order supports targeting attacks and supports, and higher-order supports which can be themselves attacked.

Finally, the potential of collective attacks in structured models of argumentation is rather unexplored. Consider an instantiation scheme like ASPIC+ [97], or instantiations for logic programs [28] or assumption-based argumentation [1] that construct AFs from a knowledge base. Using SETAFs instead of AFs as target formalism can significantly reduce the number of arguments and, in certain cases one can even ensure that each statement has a unique argument supporting that statement [56]. A first investigation in that direction is [107; 109] where SETAFs are instantiated from Datalog knowledge bases. There is also scope for relating this kind of approach to work on accrual and the other models that collect related arguments such as the argument trees of [16] and the coalitions of [33].

The accrual of arguments is a less studied problem compared to joint attacks. As we discussed in Section 5 most existing approaches are focused on structured argumentation and only few of them satisfy all three principles proposed in [96]. There are a lot of interesting future research directions in this area such as the systematic comparison and evaluation of the frameworks that support accrual and the development of methods for handling accrual in abstract argumentation. The latter could rely on the recently proposed graded semantics for abstract argumentation or may require the development of a new abstract argumentation framework that explicitly models accrual. Another interesting direction, which could also lead to a solution for this problem, is to study the relation between current approaches for accrual and collective attacks and the mapping between the frameworks that deal with these two different problems.

7 Conclusions

In this chapter we have studied different formalisms that account for joint attacks (or more generally, collective attacks) in abstract argumentation. Also, we discussed the consideration of joint attacks in the literature of structured argumentation as well as the application of joint attacks (and joint supports) in other frameworks such as bipolar argumentation frameworks or argumentation frameworks with higher-order interactions. We also touched upon works on argument accrual which, although not strictly related to the existing models of joint attacks, can be considered as a related topic. In particular, the SETAF [84] framework along with its computational complexity, algorithms and applications, was the main subject of study in this chapter.

In Section 3.1 the basic definitions of the framework were provided, followed by the presentation of extension-based semantics of the SETAF and the relationships between them, as well as the introduction of labelling-based semantics for the framework. Then, in Section 3.2 the expressive power of the SETAF was compared against that of Dung's AF [40]. For this purpose, the results and analyses reported in [48; 60; 94; 20] were accounted for. On the one hand, the characterisation of the expressive power using signatures was discussed, to then consider exponential and compact translations of SETAFs into AFs, and later discuss an indirect translation path consisting of a translation of a SETAF into an ADF [22] and a translation of the latter into an AF. The main conclusion here is that, although SETAFs could be represented by means of Dung's AFs, the SETAF indeed increases the expressive power of the AF. Moreover, as discussed in Section 3.3, the translations from a SETAF into an AF lead to an exponential blow-up in the arguments, making them not well-suited for computational matters.

In Section 3.3 different computational problems for SETAFs were characterised, following the definition of function problems and decision problems for Dung's AFs (cf. [36; 47; 35]). In particular, the decision problems include determining the credulous or skeptical acceptance of an argument under a given semantics, the verification of an extension, or determining the existence of a (non-empty) extension. Then, the computational complexity of these decision problems is addressed, and the results are linked to the existing results for decision problems in a Dung's AF. The conclusion here is that the complexity of decision problems for SETAFs is the same as the complexity of the corresponding problems for Dung's AFs. Notwithstanding this, we should note that although in both cases the complexity is stated w.r.t. the size of the input framework, the size is interpreted differently for AFs and SETAFs. On the one hand, the size of an AF is often interpreted in terms of the number of arguments of the input framework. On the other hand, since the number of attacks in a SETAF may be exponentially larger than the number of arguments in the framework (due to the existence of attacks by sets of arguments), the size of a SETAF should be interpreted in terms of the number of arguments plus the number of attacks.

Also, Section 3.3 briefly discussed the ideas behind labelling-based algorithms for

SETAFs, illustrating the algorithm for labelling enumeration under the preferred semantics. Moreover, as mentioned before, different reduction-based approaches for computing the extensions of a SETAF were presented. The former consists of encoding the SETAF and its semantics in Answer-set programming [79; 85], whereas the latter rely on translations of a SETAF into a Dung's AF or an ADF. While the drawbacks of the translations into AFs were pointed out above, we should note that the translation into an ADF offers the possibility of using existing systems for ADFs [76] without incurring significant overheads.

Section 3.4 recalled alternative models of abstract and structured argumentation which account for attacks involving sets of arguments. While in SETAF, a set of arguments can only be the source of an attack, in other models sets of arguments are only considered as a potential target of an attack (e.g. see [63]); or as both potential sources or targets of an attack (e.g. see [18; 33; 103]). The different models also differ in how the arguments within a set are treated. For example, while in the framework of [33], all arguments in a coalition are treated in the same way, i.e. they are all either accepted or rejected, in other frameworks, such as the ones proposed in [18; 63], a successful attack on a set of arguments has as a result that at least one of the arguments in the attacked set is rejected. Although the different approaches have different aims, an interesting problem is to study the extent to which they can be mapped to each other and whether there is a more general model that captures their different features.

Section 4 addresses the application of models for joint attacks in the context of Bipolar Argumentation Frameworks (BAFs) and argumentation frameworks with higher-order interactions such as those addressed in Chapter 1 [29]. First, BAFs that make use of joint attacks, joint supports, or both are recalled, highlighting the constraints they impose on the attack and support relations, as well as the adopted interpretations for the notion of support. Then, generalisations of these BAFs are presented, which incorporate higher-order interactions in order to allow for attacks and supports targeting other attacks or supports. Later, an analysis of the possibility of using joint attacks to model higher-order interactions is performed.

On the one hand, Section 4 considered the work by Gabbay on Higher-Level Argumentation Frames [62], as well as the proposed translations of HLAFs into Frames with Joint Attacks (a particular case of SETAF). Then, Gabbay's ideas are taken in the context of the AFRA [7] and the ASAF [68], two abstract argumentation frameworks allowing for binary higher-order interactions. Our findings are that, when applying the translation proposed by Gabbay to obtain the SETAF associated with an AFRA or an ASAF without supports, and then applying the SETAF semantics, the corresponding extensions might not be as expected (since the translation does not account for the existence of indirect defeats). Then, translations for obtaining a SETAF corresponding to an AFRA or an ASAF are proposed, and illustrated through examples; their formalisations for the general case of an AFRA or an ASAF are left for future research. Moreover, the possibility of using the SETAF to model generalised necessary supports is briefly analysed in Section 4, leaving an in-depth discussion for future work.

In Section 5 different works addressing the topic of argument accrual were discussed, both at the abstract and structured levels of argumentation. As discussed there, the main difference between the approaches studying argument accrual and those accounting for joint attacks (e.g., as in SETAFs) is that the strength of the arguments combined to originate a joint attack is not accounted for when evaluating the effectiveness of attacks; notwithstanding this, each argument originating a joint attack is an essential element in the sense that the attack becomes ineffective whenever one of its source arguments is missing. In contrast, in accrual, the strength of each argument is taken into account, and adding an argument to an accrual causes changes in the strength and the effectiveness of its attacks (or supports).

Finally, as stated in Section 6, many open challenges remain for research on joint attacks, in addition to those mentioned above.

Acknowledgements

This work was partially supported by EPSRC grant EP/P010105/1 and by Universidad Nacional del Sur grant 24/N046. The opinions expressed in this paper are those of the authors and do not necessarily reflect the opinions of the funders. The authors are grateful to the reviewers of this chapter for their helpful comments.

References

[1] João F. L. Alcântara, Samy Sá, and Juan Carlos Acosta Guadarrama. On the equivalence between abstract dialectical frameworks and logic programs. *Theory and Practice of Logic Programming*, 19(5-6):941–956, 2019.

[2] Gianvincenzo Alfano, Andrea Cohen, Sebastian Gottifredi, Sergio Greco, Francesco Parisi, and Guillermo R. Simari. Dynamics in abstract argumentation frameworks with recursive attack and support relations. In *Proceedings of the 24th European Conference on Artificial Intelligence*, 2020.

[3] Leila Amgoud, Claudette Cayrol, Marie-Christine Lagasquie-Schiex, and Pierre Livet. On bipolarity in argumentation frameworks. *International Journal of Intelligent Systems, Bipolar Representations of Information and Preference (Part 2: reasoning and learning)*, 23(10):1062–1093, 2008.

[4] Leila Amgoud and Srdjan Vesic. Rich preference-based argumentation frameworks. *International Journal of Approximate Reasoning*, 55(2):585–606, 2014.

[5] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011.

[6] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 4, pages 159–236. College Publications, 2018.

[7] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52:19–37, 2011.

[8] Pietro Baroni and Massimiliano Giacomin. Solving semantic problems with odd-length cycles in argumentation. In Thomas D. Nielsen and Nevin Lianwen Zhang, editors, *Proceedings of the 7th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 440–451. Springer, 2003.

[9] Pietro Baroni and Massimiliano Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.

[10] Pietro Baroni, Massimiliano Giacomin, and Beishui Liao. A general semi-structured formalism for computational argumentation: Definition, properties, and examples of application. *Artificial Intelligence*, 257:158 – 207, 2018.

[11] Pietro Baroni, Antonio Rago, and Francesca Toni. From fine-grained properties to broad principles for gradual argumentation: A principled spectrum. *International Journal of Approximate Reasoning*, 105:252–286, 2019.

[12] Pietro Baroni, Marco Romano, Francesca Toni, Marco Aurisicchio, and Giorgio Bertanza. Automatic evaluation of design alternatives with quantitative argumentation. *Argument & Computation*, 6(1):24–49, 2015.

[13] Ringo Baumann. On the nature of argumentation semantics: Existence and uniqueness, expressibility, and replaceability. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 17, pages 839–936. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2779–2886.

[14] Ringo Baumann, Gerhard Brewka, and Markus Ulbricht. Revisiting the foundations of abstract argumentation — semantics based on weak admissibility and weak defense. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 2742–2749. AAAI Press, 2020.

[15] Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[16] Philippe Besnard and Anthony Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128:203–235, 2001.

[17] David Billington, Grigoris Antoniou, Guido Governatori, and Michael Maher. An inclusion theorem for defeasible logics. *ACM Transactions on Computational Logic*,

12(1), November 2010.

[18] Alexander Bochman. Collective argumentation and disjunctive logic programming. *Journal of Logic and Computation*, 13(3):405–428, 2003.

[19] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1-2):63–101, 1997.

[20] Gerd Brewka, Paul E. Dunne, and Stefan Woltran. Relating the semantics of abstract dialectical frameworks and standard AFs. In *Proceedings of the 22nd International Joint Conference on Articial Intellignce*, 2011.

[21] Gerhard Brewka, Martin Diller, Georg Heissenberger, Thomas Linsbichler, and Stefan Woltran. Solving advanced argumentation problems with answer-set programming. In Satinder P. Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 1077–1083. AAAI Press, 2017.

[22] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract dialectical frameworks. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 5, pages 237–285. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2263–2318.

[23] Gerhard Brewka, Hannes Strass, Stefan Ellmauthaler, Johannes Peter Wallner, and Stefan Woltran. Abstract dialectical frameworks revisited. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 803–809, 2013.

[24] Martin Caminada. Semi-stable semantics. In *Proceedings of the 1st Conference on Computational Models of Argument*, pages 121–130, 2006.

[25] Martin Caminada. Comparing two unique extension semantics for formal argumentation: Ideal and eager. In *Proceedings of the 19th Belgian-Dutch Conference on Artificial Intelligence*, pages 81–87, 2007.

[26] Martin Caminada and Paul E. Dunne. Strong admissibility revisited: Theory and applications. *Argument & Computation*, 10(3):277–300, 2019.

[27] Martin Caminada and Dov M. Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2/3):109–145, 2009.

[28] Martin Caminada, Samy Sá, João F. L. Alcântara, and Wolfgang Dvořák. On the equivalence between logic programming semantics and argumentation semantics. *International Journal of Approximate Reasoning*, 58:87–111, 2015.

[29] Claudette Cayrol, Andrea Cohen, and Marie-Christine Lagasquie-Schiex. Higher-order interactions (bipolar or not) in abstract argumentation: A state of the art. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 1. College Publications, 2021.

[30] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Argumentation frameworks with recursive attacks and evidence-based support. In F. Ferrarotti and S. Woltran, editors, *Proceedings of the 10th International Symposium on Foundations of Information and Knowledge Systems*, pages

150–169. Springer-Verlag, 2018.

[31] Claudette Cayrol, Jorge Fandinno, Luis Fariñas del Cerro, and Marie-Christine Lagasquie-Schiex. Structure-based semantics of argumentation frameworks with higher-order attacks and supports. In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *International Conference on Computational Models of Argument*, pages 29–36. IOS Press, septembre 2018.

[32] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty*, pages 378–389, 2005.

[33] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Coalitions of arguments: A tool for handling bipolar argumentation frameworks. *International Journal of Intelligent Systems*, 25(1):83–109, 2010.

[34] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolarity in argumentation graphs: Towards a better understanding. *International Journal of Approximate Reasoning*, 54(7):876–899, 2013.

[35] Federico Cerutti, Sarah A. Gaggl, Matthias Thimm, and Johannes P. Wallner. Foundations of implementations for formal argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 15, pages 688–767. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2623–2706.

[36] Günther Charwat, Wolfgang Dvořák, Sarah Alice Gaggl, Johannes Peter Wallner, and Stefan Woltran. Methods for solving reasoning problems in abstract argumentation - A survey. *Artificial Intelligence*, 220:28–63, 2015.

[37] Andrea Cohen, Sebastian Gottifredi, Alejandro Javier García, and Guillermo Ricardo Simari. A survey of different approaches to support in argumentation systems. *Knowledge Engineering Review*, 29(5):513–550, 2014.

[38] Robyn M. Dawes. The robust beauty of improper linear models in decision making. *American Psychologist*, 34(7):571, 1979.

[39] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning and logic programming. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, pages 852–857, Chambéry, France, 1993.

[40] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, September 1995.

[41] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10-15):642–674, July 2007.

[42] Paul E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artificial Intelligence*, 171(10-15):701–729, 2007.

[43] Paul E. Dunne. The computational complexity of ideal semantics. *Artificial Intelligence*, 173(18):1559–1591, 2009.

[44] Paul E. Dunne, Wolfgang Dvořák, Thomas Linsbichler, and Stefan Woltran. Characteristics of multiple viewpoints in abstract argumentation. *Artificial Intelligence*, 228:153–178, 2015.

[45] Paul E. Dunne, Wolfgang Dvořák, and Stefan Woltran. Parametric properties of ideal semantics. *Artificial Intelligence*, 202:1–28, 2013.

[46] Paul E. Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457 – 486, 2011.

[47] Wolfgang Dvořák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 14, pages 631–687. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2557–2622.

[48] Wolfgang Dvořák, Jorge Fandinno, and Stefan Woltran. On the expressive power of collective attacks. *Argument & Computation*, 10(2):191–230, 2019.

[49] Wolfgang Dvořák, Alexander Greßler, and Stefan Woltran. Evaluating SETAFs via Answer-Set Programming. In *Proceedings of the Second International Workshop on Systems and Algorithms for Formal Argumentation*, pages 10–21. CEUR-WS.org, 2018.

[50] Wolfgang Dvořák, Atefeh Keshavarzi Zafarghandi, and Stefan Woltran. Expressiveness of SETAFs and support-free ADFs under 3-valued semantics. In *Proceedings of the 8th Conference on Computational Models of Argument*, pages 191–202. IOS Press, 2020.

[51] Wolfgang Dvořák, Sebastian Ordyniak, and Stefan Szeider. Augmenting tractable fragments of abstract argumentation. *Artificial Intelligence*, 186:157–173, 2012.

[52] Wolfgang Dvořák, Reinhard Pichler, and Stefan Woltran. Towards fixed-parameter tractable algorithms for abstract argumentation. *Artificial Intelligence*, 186:1–37, 2012.

[53] Wolfgang Dvořák, Anna Rapberger, and Johannes Peter Wallner. Labelling-based algorithms for SETAFs. In *Proceedings of the Third International Workshop on Systems and Algorithms for Formal Argumentation*, pages 34–46. CEUR-WS.org, 2020.

[54] Wolfgang Dvořák, Anna Rapberger, and Stefan Woltran. Strong equivalence for argumentation frameworks with collective attacks. In *Proceedings of the 42nd German Conference on Artificial Intelligence*, pages 131–145. Springer, 2019.

[55] Wolfgang Dvořák, Anna Rapberger, and Stefan Woltran. On the different types of collective attacks in abstract argumentation: equivalence results for SETAFs. *Journal of Logic and Computation*, 30(5):1063–1107, 2020.

[56] Wolfgang Dvořák, Anna Rapberger, and Stefan Woltran. On the relation between claim-augmented argumentation frameworks and collective attacks. In *Proceedings of the 24th European Conference on Artificial Intelligence*, pages 721–728. IOS Press, 2020.

[57] Uwe Egly, Sarah Alice Gaggl, and Stefan Woltran. Answer-set programming encodings

for argumentation frameworks. *Argument & Computation*, 1(2):147–177, 2010.

[58] Stefan Ellmauthaler and Hannes Strass. The DIAMOND system for computing with abstract dialectical frameworks. In *Proceedings of the 5th Conference on Computational Models of Argumen*, volume 266, pages 233–240. IOS Press, 2014.

[59] Morten Elvang-Gøransson, Paul J. Krause, and John Fox. Acceptability of arguments as 'logical uncertainty'. In M. Clarke, R. Kruse, and S. Moral, editors, *Proceedings of the 2nd European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 85–90. Springer, 1993.

[60] Giorgos Flouris and Antonis Bikakis. A comprehensive study of argumentation frameworks with sets of attacking arguments. *International Journal of Approximate Reasoning*, 109, 03 2019.

[61] J. Fox, D. Barber, and K. D. Bardhan. Alternatives to Bayes? a quantitative comparison with rule-based diagnostic inference. *Methods of Information in Medicine*, 19:210–215, 1980.

[62] Dov M. Gabbay. Semantics for higher level attacks in extended argumentation frames. *Studia Logica*, 93:357–381, 2009.

[63] Dov M. Gabbay and Michael Gabbay. Theory of disjunctive attacks, Part I. *Logic Journal of the IGPL*, 24(2):186–218, 2016.

[64] Sarah Alice Gaggl, Thomas Linsbichler, Marco Maratea, and Stefan Woltran. Design and results of the second international competition on computational models of argumentation. *Artificial Intelligence*, 279, 2020.

[65] Alejandro Javier García and Guillermo Ricardo Simari. Argumentation based on logic programming. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 8, pages 409–435. College Publications, 2018.

[66] Martin Gebser, Benjamin Kaufmann, Roland Kaminski, Max Ostrowski, Torsten Schaub, and Marius T. Schneider. Potassco: The Potsdam answer set solving collection. *AI Communications*, 24(2):107–124, 2011.

[67] Thomas F. Gordon. Defining argument weighing functions. *Journal of Applied Logics*, 5(3):747–773, 2018.

[68] Sebastian Gottifredi, Andrea Cohen, Alejandro Javier García, and Guillermo Ricardo Simari. Characterizing acceptability semantics of argumentation frameworks with recursive attack and support relations. *Artificial Intelligence*, 262:336–368, 2018.

[69] Guido Governatori and Michael J. Maher. An Argumentation-Theoretic Characterization of Defeasible Logic. In *Proceedings of the 14th European Conference on Artificial Intelligence*, pages 469–473, 2000.

[70] Benjamin N. Grosof. Prioritized conflict handling for logic programs. In Jan Maluszynski, editor, *Proceedings of the 1997 International Symposium on LOgic Programming*, pages 197–211. MIT Press, 1997.

[71] Davide Grossi and Sanjay Modgil. On the graded acceptability of arguments. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages

868–874. AAAI Press, 2015.

[72] Paul J. Krause, Simon Ambler, Morten Elvang-Gøransson, and John Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11 (1):113–131, 1995.

[73] E. Laenens and D. Vermeir. A Fixpoint Semantics for Ordered Logic. *Journal of Logic and Computation*, 1(2):159–185, 12 1990.

[74] João Leite and João G. Martins. Social abstract argumentation. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona*, pages 2287–2292. IJCAI/AAAI, 2011.

[75] Nicola Leone, Gerald Pfeifer, Wolfgang Faber, Thomas Eiter, Georg Gottlob, Simona Perri, and Francesco Scarcello. The DLV system for knowledge representation and reasoning. *ACM Transactions on Computational Logic*, 7(3):499–562, 2006.

[76] Thomas Linsbichler, Marco Maratea, Andreas Niskanen, Johannes Peter Wallner, and Stefan Woltran. Novel algorithms for abstract dialectical frameworks based on complexity analysis of subclasses and SAT solving. In Jérôme Lang, editor, *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 1905–1911, 2018.

[77] Thomas Linsbichler, Jörg Pührer, and Hannes Strass. A uniform account of realizability in abstract argumentation. In *Proceedings of the 22nd European Conference on Artificial Intelligence*, pages 252–260. IOS Press, 2016.

[78] Mauro J. Gómez Lucero, Carlos I. Chesñevar, and Guillermo R. Simari. On the accrual of arguments in defeasible logic programming. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 804–809. Morgan Kaufmann Publishers Inc., 2009.

[79] Victor W. Marek and Mirosław Truszczyński. Stable models and an alternative logic programming paradigm. In *The Logic Programming Paradigm – A 25-Year Perspective*, pages 375–398. Springer, 1999.

[80] Sanjay Modgil and Trevor J. M. Bench-Capon. Metalevel argumentation. *Journal of Logic and Computation*, 21(6):959–1003, 09 2010.

[81] Sanjay Modgil and Henry Prakken. Abstract rule-based argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 6, pages 287–364. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2319–2406.

[82] Søren Holbech Nielsen and Simon Parsons. Computing preferred extensions for argumentation systems with sets of attacking arguments. In *Proceedings of the 1st Conference on Computational Models of Argument*, pages 97–108. IOS Press, 2006.

[83] Søren Holbech Nielsen and Simon Parsons. An application of formal argumentation: Fusing Bayesian networks in multi-agent systems. *Artificial Intelligence*, 171(10–15):754–775, 2007.

[84] Søren Holbech Nielsen and Simon Parsons. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Proceedings*

of the 3^{rd} International Workshop on Argumentation in Multi-Agent Systems, pages 54–73, 2007.

[85] Ilkka Niemelä. Logic programming with stable model semantics as a constraint programming paradigm. *Annals of Mathematics and Artificial Intelligence*, 25(3–4):241–273, 1999.

[86] Samer Nofal, Katie Atkinson, and Paul E. Dunne. Algorithms for decision problems in argument systems under preferred semantics. *Artificial Intelligence*, 207:23–51, 2014.

[87] Farid Nouioua. AFs with necessities: Further semantics and labelling characterization. In *7th International Conference on Scalable Uncertainty Management*, pages 120–133, 2013.

[88] Farid Nouioua and Vincent Risch. Argumentation frameworks with necessities. In *Proceedings of the 5th International Conference on Scalable Uncertainty Management*, pages 163–176, 2011.

[89] Donald Nute. *Defeasible Logic*, page 353?395. Oxford University Press, Inc., USA, 1994.

[90] Mike O'Neil, Andrzej J. Glowinski, and John Fox. A symbolic theory of decision-making applied to several medical tasks. In *AIME 89: Proceedings of the Second European Conference on Artificial Intelligence in Medicine*, pages 62–71. Springer, 1989.

[91] Nir Oren and Timothy J. Norman. Semantics for evidence-based argumentation. In *Proceedings of the 2nd Conference on Computational Models of Argument*, pages 276–284, 2008.

[92] Theodore Patkos, Giorgos Flouris, and Antonis Bikakis. Symmetric multi-aspect evaluation of comments - extended abstract. In *Proceedings of the 22nd European Conference on Artificial Intelligence*, pages 1672–1673. IOS Press, 2016.

[93] Sylwia Polberg. Understanding the abstract dialectical framework. In *Proceedings of the 15th European Conference on Logics in Artificial Intelligence*, pages 430–446, 2016.

[94] Sylwia Polberg. *Developing the abstract dialectical framework.* PhD thesis, TU Wien, Institute of Information Systems, 2017. available at `http://katalog.ub.tuwien.ac.at/AC13773888`.

[95] Sylwia Polberg and Nir Oren. Revisiting support in abstract argumentation systems. In *Proceedings of the 5th Conference on Computational Models of Argument*, pages 369–376, 2014.

[96] Henry Prakken. A study of accrual of arguments, with applications to evidential reasoning. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, pages 85–94, New York, NY, USA, 2005. ACM.

[97] Henry Prakken. Modelling accrual of arguments in ASPIC+. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, pages 103–112, New York, NY, USA, 2019. Association for Computing Machinery.

[98] Jörg Pührer. Realizability of three-valued semantics for abstract dialectical frame-

works. *Artificial Intelligence*, 278, 2020.

[99] Hannes Strass. Expressiveness of two-valued semantics for abstract dialectical frameworks. *Journal of Artificial Intelligence Research*, 54:193–231, 2015.

[100] Hannes Strass. The relative expressiveness of abstract argumentation and logic programming. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 1625–1631, 2015.

[101] Matthias Thimm and Serena Villata. The first international competition on computational models of argumentation: Results and analysis. *Artificial Intelligence*, 252:267–294, 2017.

[102] Leendert van der Torre and Srdjan Vesic. The principle-based approach to abstract argumentation semantics. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 16, pages 797–838. College Publications, 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2735–2778.

[103] Bart Verheij. Accrual of arguments in defeasible argumentation. In *Proceedings of the Second Dutch/German Workshop on Nonmonotonic Reasoning*, pages 217–224, 1995.

[104] Bart Verheij. *Rules, Reasons, Arguments. Formal studies of argumentation and defeat.* PhD thesis, Universiteit Maastricht, 1996.

[105] Bart Verheij. Two approaches to dialectical argumentation: Admissible sets and argumentation stages. In *In Proceedings of the International Conference on Formal and Applied Practical Reasoning*, pages 357–368. Universiteit, 1996.

[106] Gerard A. W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90(1–2):225–279, February 1997.

[107] Bruno Yun, Madalina Croitoru, Srdjan Vesic, and Pierre Bisquert. NAKED: n-ary graphs from knowledge bases expressed in Datalog$^{\pm}$. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2390–2392, 2019.

[108] Bruno Yun, Srdjan Vesic, and Madalina Croitoru. Ranking-based semantics for sets of attacking arguments. In *Proceedings of the 34th Conference on Artificial Intelligence*, pages 3033–3040. AAAI Press, 2020.

[109] Bruno Yun, Srdjan Vesic, and Madalina Croitoru. Sets of attacking arguments for inconsistent Datalog knowledge bases. In *Proceedings of the 8th Conference on Computational Models of Argument*, pages 419–430. IOS Press, 2020.

Received May 2021

Collective Acceptability in Abstract Argumentation

Dorothea Baumeister, Daniel Neugebauer, Jörg Rothe
Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, Germany
{d.baumeister,daniel.neugebauer,rothe}@hhu.de

Abstract

This chapter highlights the collective acceptability problem in multiagent argumentation, which is related to the problem of collective decision making in the field of computational social choice at the intersection of social choice theory, theoretical computer science, and artificial intelligence. Specifically, the chapter surveys various approaches to collective acceptability and showcases useful methods for structural aggregation of argumentation frameworks and their properties.

1 Introduction

In this chapter, we will be concerned with collective decision making in multiagent scenarios, specifically focusing on how methods from computational social choice (COMSOC) can be employed to solve the problem of collective acceptability in multiagent argumentation. Computational social choice (see the textbooks edited by [23] and [74]) is an interdisciplinary area at the interface of social choice theory, computer science, and (distributed) artificial intelligence. A core research stream in COMSOC is the study of voting. While voters have individual preferences over alternatives among which they seek to determine the best choices, called the *winners* of the election, in argumentation theory we are faced with agents who have individual views on the arguments and on the attack relation among arguments and who seek to determine the best choices of arguments, i.e., *acceptable sets of arguments*, according to certain semantics. Just as preference aggregation through voting rules has been studied in COMSOC, various ways of aggregating argumentation frameworks (AFs) of different types have been proposed.

We will survey the latter through the COMSOC lens, paying particular attention to how methods and concepts originally designed for COMSOC have been applied in the context of argumentation. For example, axiomatic properties of voting mechanisms have been thoroughly studied in social choice theory and also in COMSOC, and in Section 3 we will present what is known about the axiomatic properties of methods used to aggregate AFs. In Section 4, we will turn to specific aggregation methods for AFs. In particular, we will present existing aggregation operators and their properties that make use of partial, incomplete, and value-based AFs, the former two being related to uncertainty and incomplete information about the argumentation at hand and the latter being related to the impact a ranking of values assigned to arguments may have on the argumentation process.

Uncertainty can occur both in voting and in argumentation, even though due to distinct sources of incomplete information. In voting, we may be faced with "noisy" elections [72; 82; 25], for example due to incorrect vote counts, either by accident or with malicious intent, or with voters who are simply too lazy to rank all the alternatives [10; 66]. In argumentation, dynamic changes in a given AF [27], different and changing individual views and beliefs of the involved agents [26], or uncertainty in the underlying knowledge base used to instantiate the AF [67] all may lead to uncertainty about the status of an AF [31; 13; 69; 77].

Central concepts used for voting with uncertainty are the notions of *possible* and *necessary winner*, proposed by [58] and studied in more depth by, e.g., [83], [29], and 14 [14, 15]. Intuitively, assuming that the voters' preferences over the alternatives are incomplete, a *possible winner* is an alternative that can be made win for *some* complete extension of the voters' preferences, whereas a *necessary winner* is an alternative that must win for *every* complete extension of the voters' preferences. These notions are so important that they could be beneficially applied in other areas as well, such as in fair division by [21], [7], and [61]; in algorithmic game theory by [57] and [75]; and in judgment aggregation by 8 [8, 9].

As mentioned earlier, at the intersection of computational social choice and formal argumentation lies the problem of *collective acceptability.* Acceptability in the standard argumentation model means that we are looking at a single representation of an argumentation and try to determine which arguments are acceptable under a certain semantics. Collective acceptability, on the other hand, is concerned with acceptability in a set of several related, but potentially different representations of a single argumentation. Problems of collective acceptability can arise in various applications. When individual agents each create their personal representation of the argumentation from a private knowledge base, different or missing information in these knowledge bases will have the effect that the argumentation representations will be different. Determining argument acceptability in the individual views will have limited significance, while collective acceptability with regard to all individual views can incorporate all information available to the agents and produce more

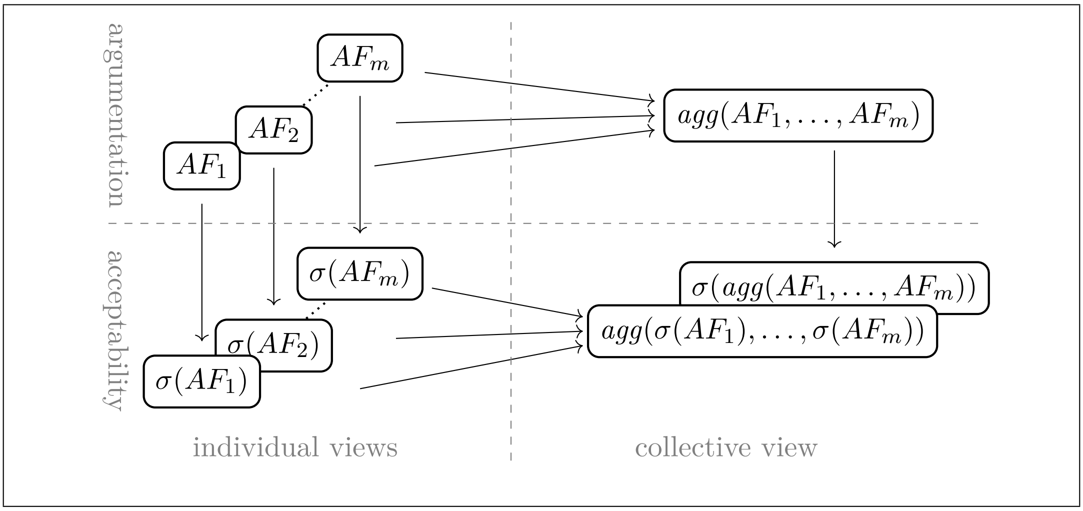


Figure 1: Schematic overview of different approaches to collective acceptability in formal argumentation, where σ is a general placeholder for some kind of acceptability criterion on AFs, and *agg* is a general placeholder for some aggregation operator on AFs.

meaningful results. In strategic scenarios, arguments known to all agents (and thus present in all individual views) may represent arguments that were already given in a shared discussion, while arguments known only to some, but not all agents may represent arguments that these agents could still bring forward. Here, collective acceptability may help anticipate which arguments might become acceptable in the further progress of the discussion, and which of the available arguments should be brought forward to improve an agent's position in the argumentation.

There are two fundamentally different approaches to collective acceptability in the literature. In the survey by [20], these are called the argument-wise and the framework-wise approach. The *argument-wise* approach determines acceptability in the individual views using standard methods, and then defines *semantic aggregation* methods—e.g., voting procedures—to aggregate the individually accepted arguments into a single collectively acceptable set of arguments. The *framework-based* approach defines *structural aggregation* methods to aggregate individual views into a collective representation first, and then determines acceptability in the collective representation, either by standard methods or by dedicated methods for that representation.

Figure 1 illustrates these two approaches to the problem of collective acceptability. The elements in the top-left corner represent the individual views of m agents.

Semantic evaluation on the individual views (left arrows going down) produces individually acceptable outcomes (bottom left), which can be aggregated using semantic aggregation (bottom arrows going right) into a collectively acceptable outcome. Alternatively, structural aggregation on the individual views (top arrows going right) produces a single collective representation of the argumentation (top right), which can be semantically evaluated (right arrow going down) for a collectively acceptable outcome.

Semantic aggregation is suitable for applications where the outcome of the individual views (i.e., the accepted arguments) is considered to be the most important part of the views, overshadowing the importance of the underlying structure of the argumentation. This might be the case when agents have created their individual argumentations with the specific purpose to support the acceptability of certain key arguments. The agents might not be very interested in finding a collective view of the argumentation that is close to their individual view, but they are highly interested in having collectively accepted arguments that are close to the accepted arguments in their view. Semantic aggregation is applied by [36] via merging extensions, and by [24] and [5] via merging labelings. A related research problem to the argument-wise approach to collective acceptability is the problem of *realizability* [44] or *synthesis* [70], where sets of accepted arguments are given as input, and the goal is to find an AF whose sets of accepted arguments are the same as, or as close as possible to, the given sets. These methods can be used to augment the results of semantic aggregation methods to not only obtain collectively acceptable arguments as a result, but also an AF that produces these. Additional constraints can be used to make sure that the AF created is as close as possible to the input AFs. Relatedly, the *enforcement* problem asks whether a given set of arguments can be enforced as an acceptable set by a finite number of elementary changes to a given AF. The original definition by [6] allows the addition of single arguments along with incident attacks, while subsequent work allows the addition and deletion of attacks [81], or both simultaneously [33].

The second approach of structural aggregation is suitable when the structure in the individual views is more important than their accepted arguments. For example, this might be the case when each agent has access to a limited part of the information on a subject matter, so the significance of which arguments are accepted in the individual views is rather limited. Merging the individual fragments of the information into an aggregated representation of the argumentation creates a better basis for semantic evaluation in such situations. In this chapter, we focus on structural aggregation.

a	"Everybody needs access to medical supplies for personal protection."
b	"Medical supplies are not sufficient, so hospitals must have priority access to medical supplies."
c	"Medical supplies are sufficient for all."
d	"The disease problem must be solved by government health officials, the population should stay out of it."
e	"Everybody has a right to know how dangerous the disease really is."
f	"Information about the dangers of the disease may cause a panic in the population, leading to hoarding of medical supplies and thus a shortage of these."
g	"If people know enough about the disease, they can effectively protect themselves without the need of medical supplies."

Table 1: Arguments used in the AF shown in Figure 2

2 Preliminaries

We recall the model of abstract argumentation frameworks due to [42].

Definition 2.1. *An* argumentation framework (AF) *is a pair $AF = \langle Ar, att \rangle$ consisting of a finite set Ar of arguments and a binary attack relation $att \subseteq Ar \times Ar$.*

As an example, consider the set of arguments, $Ar = \{a, b, c, d, e, f, g\}$, displayed in Table 1. These arguments might be given in the context of a virus disease that is spreading in a population, and they make different suggestions on how to react to the disease. We will use this argumentation as a running example throughout this chapter.

Arguments a and b mutually attack each other, and so do d and e, as well as f and g. Further, argument c attacks b, argument f attacks c and e, and argument g attacks b and d. This is formally captured by the attack relation

$$att = \{(a, b), (b, a), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (f, g), (g, b), (g, f)\}.$$

Figure 2 gives a graph representation of this AF, $\langle Ar, att \rangle$.

An argumentation *semantics* σ maps a given argumentation framework AF to the set of σ*-extensions* of AF, which are the sets of arguments that are acceptable in AF with respect to σ. A set S of arguments is called *conflict-free* ($\mathcal{CF}$) if there are no

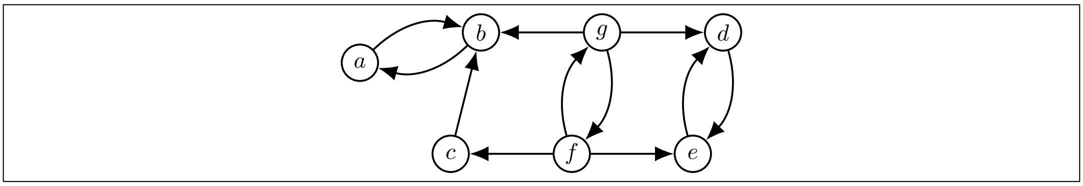


Figure 2: Graph representation of the AF used in our running example, where arguments are displayed as nodes and attacks as directed edges

attacks among arguments in S. A conflict-free set S is *admissible* ($\mathcal{AD}$) if it defends each of its arguments, where we say $S \subseteq Ar$ *defends* $a \in Ar$ if for each argument $b \in Ar$ attacking a, there is an argument $c \in S$ attacking b. A maximal (with respect to set inclusion) admissible set S is said to be *preferred* ($\mathcal{PR}$). A conflict-free set S is *stable* ($\mathcal{ST}$) if every argument outside of S is attacked by some argument in S. An admissible set that is closed under defense—i.e., that includes every argument that it defends—is called *complete* ($\mathcal{CO}$). The unique minimal complete set is the *grounded* extension ($\mathcal{GR}$).

In this chapter, we consider aggregation operators on argumentation frameworks, which are mappings that aggregate a set of argumentation frameworks into a collective representation. We use the following general notation. Let $\mathbb{AF}$ denote the universe of all possible argumentation frameworks. An *aggregation operator* $agg : \mathbb{AF}^m \to 2^{\mathbb{AF}}$ for argumentation frameworks is a mapping from a set of m input argumentation frameworks (which may represent the individual views of m agents) to a set of aggregate argumentation frameworks. An aggregation operator is called *resolute* if it always outputs a singleton, i.e., if $|agg(P)| = 1$ for all profiles $P \in \mathbb{AF}^m$, and *irresolute* otherwise. Generalized aggregation operators may use an extended target format that goes beyond standard argumentation frameworks.

3 Axiomatic Properties of Aggregation Methods in Argumentation

In social choice theory, mechanisms for collective decision making are studied with respect to various axioms. Such axioms express desirable behavior of these mechanisms. Unfortunately, there is a number of impossibility results such as Arrow's Theorem ([3]) and the Gibbard–Satterthwaite Theorem ([53]; [76]) showing that it is impossible to fulfill some of the most basic criteria simultaneously. Similar questions arise in collective argumentation, especially when different views of agents should be aggregated. This is, for example, the case when agents have different opinions about the attack relation. An important question is then which properties of the individual attack relations will be preserved by a given aggregation rule. In the case of argumentation, these properties will be specifically related to the various semantics. In this section, we will review results by [28] about the preservation of semantic properties in the aggregation of abstract argumentation frameworks. This work focuses on the case where all agents consider the same set of arguments but have different opinions on the attacks among them. However, a generalization where the argument sets may differ for each agent would also be possible.

The general setting we consider is a common finite set of arguments Ar and a set $N = \{1, \dots, n\}$ of agents. The individual view of agent $i \in N$ is represented as an argumentation framework $AF_i = \langle Ar, att_i \rangle$. The profile $P = (att_1, \dots, att_n)$ consist of all individual attack relations. Additionally, $N_r^P = \{i \in N \mid r \in att_i\}$ is the set of supporters of attack r in profile P. To aggregate these individual views, we use aggregation rules. For a fixed number of n agents, they are formally defined as a function

$$F : (Ar \times Ar)^n \to Ar \times Ar.$$

One class of aggregation rules often used in the context of judgment aggregation are quota rules that have been introduced by [38] and further studied by, e.g., 8 [8, 9]. The idea is that an element will be included in the aggregated outcome if the agreement exceeds some given quota. According to [28], the definition is as follows in the context of argumentation frameworks.

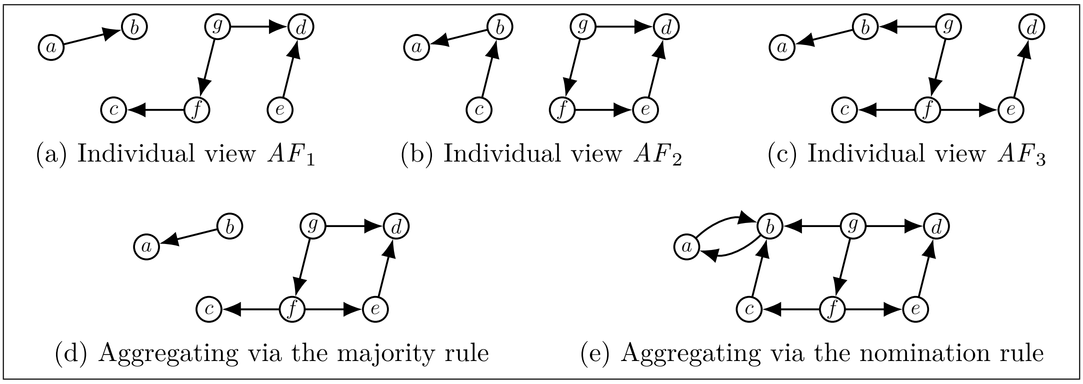
Definition 3.1 (Quota Rule). *For $q \in \{1, \dots, n\}$ and a profile P, the* quota rule *F_q is defined as*

$$F_q(P) = \{r \in Ar \times Ar \mid |N_r^P| \geq q\}.$$

Hence, all attacks that are supported by at least q agents are accepted.

Prominent examples are the *majority rule* F_q with quota $q = \lceil \frac{n}{2} \rceil$ and the *nomination rule* F_1 with quota 1. The latter rule requires only one nomination of every attack that is included in the aggregated argumentation framework, which is a reasonable choice especially in argumentation where conflicts should be taken seriously. Another example of a rule is the *dictatorship* of a specific agent. The outcome for the dictatorship of agent $i \in N$ is always the individual AF of this agent. In contrast to the quota rules defined above, this rule does not take into account the attack relations of all agents.

Example 3.2. *Recall our running example from Table 1 and Figure 2 in Section 2. We consider a profile consisting of the attack relations (as shown in Figures 3a–3c) of the three individual AFs AF_1, AF_2, and AF_3 representing the individual views*



(a) Individual view AF_1 (b) Individual view AF_2 (c) Individual view AF_3

(d) Aggregating via the majority rule (e) Aggregating via the nomination rule

Figure 3: Graph visualizations for Example 3.2

of three agents over the same set of arguments. When using the majority rule, all attacks with at least two supporters will be included in the aggregated AF. The result is shown in Figure 3d. Under the nomination rule, all attacks that are present in at least one individual AF are contained in the aggregated outcome, as shown in Figure 3e.

Other rules that are used in voting or judgment aggregation may also be transferred to abstract argumentation. For example, rules that minimize the distance to the individual votes or the individual judgments (for one of the common types of distance between preferences or judgment sets), like the Kemeny rule in voting [56; 54] and in judgment aggregation [35; 34], can also be used for the aggregation of argumentation frameworks. They will not be considered here, though.

An important property of aggregation rules is the existence of agents with veto power. These are agents which may not be ignored, and hence only attacks that exist in their individual attack relation may be included in the aggregated outcome.

Definition 3.3 (Veto Power)**.** *Agent $i \in N$ has* veto powers *under aggregation rule F if for every profile P, it holds that*

$$F(P) \subseteq att_i.$$

It is obvious that under the majority and nomination rules, no agent has veto powers. In a dicatorship, the dictator has veto powers.

The most basic axioms in social choice are anonymity and neutrality. Their intuitive meaning in the context of voting rules is that all voters and all candidates, respectively, are treated equally, which are quite basic fairness criteria. This can be directly transferred to agents and attacks for AF aggregation rules.

Definition 3.4 (Anonymity and Neutrality).

- *An aggregation rule F is* anonymous *if for all profiles P and all permutations $\pi : N \to N$, it holds that*

$$F(P) = F(att_{\pi(1)}, \dots, att_{\pi(n)}).$$

- *An aggregation rule F is* neutral *if for all profiles P and all attacks a, a' with $N^P_a = N^P_{a'}$, it holds that*

$$a \in F(P) \Leftrightarrow a' \in F(P).$$

From the definition of quota rules it follows that they are anonymous and neutral. However, there may also be reasons for AF aggregation rules that are not anonymous. This is for example the case when some agents are considered to be experts and their view (maybe on a subset of the arguments) should have more weight in the aggregated outcome. An example for such rules are so-called *qualified majority rules*, where a subset of the agents has veto powers. The acceptance of an attack depends on acceptance by a weak majority and by the agents that have veto powers. Similar reasons can justify aggregation rules that are not neutral, since some arguments may be more important than others. Properties of qualified majority rules, which are not anonymous, have been studied by [79]. A dictatorship is obviously not anonymous and not neutral.

In social choice theory, there are many different formulations for independence axioms. The common idea is that the choice between two alternatives should only depend on their relation and not on other ("irrelevant") alternatives. For the aggregation of AFs, we require that the acceptance of an attack only depends on the supporters of this attack.

Definition 3.5 (Independence). *An aggregation rule F is* independent *if for all profiles P, P' and all attacks a with $N^P_a = N^{P'}_a$, it holds that*

$$a \in F(P) \Leftrightarrow a \in F(P').$$

As the definition of quota rules relies only on the number of supporters for the attacks, they satisfy independence. Again, dictatorships violate independence. As for anonymity and neutrality, there may be reasons to consider AF aggregation rules that are not independent. If additional structural relations between the arguments are considered that have to be respected in the aggregated outcome, it may be useful to consider non-independent aggregation rules. The situation in judgment aggregation is similar, but here the relation between the different issues are present through the given formulas. However, independence is a key axiom for several impossibility results in judgment aggregation. [78] provide a recent study of alternative formulations for independence in judgment aggregation.

A very intuitive property for aggregation mechanisms is that additional support should not be harmful. This is captured by the monotonicity axiom. Violation of this axiom is considered to be a serious disadvantage, as stated, for example, by [49]. A prominent example of a voting rule violating monotonicity is the Dodgson rule (see [41]), as shown by [49] for five alternatives and by [50] even for four alternatives; see also [22].

For the aggregation of argumentation frameworks, monotonicity requires that a selected attack will never be rejected if it receives more support from the agents.

Definition 3.6 (Monotonicity)**.** *An aggregation rule F is* monotonic *if for all profiles P, P' and all attacks a with $N_a^P \subseteq N_a^{P'}$ and $N_{a'}^P = N_{a'}^{P'}$ for all attacks $a' \neq a$, it holds that*

$$a \in F(P) \Rightarrow a \in F(P').$$

Similarly to the other axioms, monotonicity is satisfied by quota rules but violated by dictatorships.

The last two properties we consider are unanimity and groundedness. We will follow [28] who write: "Note that, in line with the existing literature in argumentation theory on the one hand and social choice theory on the other, we use the term 'grounded' in two unrelated ways (grounded extension vs. grounded aggregation rules)." It will always be clear from the context whether we use the term *grounded* in the sense of argumentation theory or social choice theory.

While unanimity requires that an aggregation rule has to follow a unanimous decision on an attack, groundedness requires that at least one supporter must exist for an attack to be selected.

Definition 3.7 (Unanimity and Groundedness)**.**

- *An aggregation rule F is* unanimous *if for all profiles P, it holds that*

$$F(P) \supseteq att_1 \cap \cdots \cap att_n.$$

- *An aggregation rule F is* grounded *if for all profiles P, it holds that*

$$F(P) \subseteq att_1 \cup \cdots \cup att_n.$$

The quota lies between 1 and the number of agents. Hence all quota rules are unanimous and grounded. Dictatorships are also grounded, since all attacks are contained in the individual AF of the dictator, but unanimity is obviously violated. To summarize, quota rules satisfy all introduced basic axioms: anonymity, neutrality, independence, monotonicity, unanimity and groundedness.

A very important concept in social choice theory is *collective rationality* (see [3]) with respect to some given property. In the case of transitive preferences of individuals, collective rationality would imply that the aggregated preference is also transitive. The Condorcet paradox [30] shows that this is not the case for the pairwise majority comparison between alternatives. Collective rationality has also been studied in judgment aggregation by [64] and in graph aggregation by [47]. For AF-aggregation, [28] define collective rationality with respect to some *AF-property* $Prop \subseteq 2^{Ar \times Ar}$, which is the set of all attack relations that satisfy *Prop*.

Definition 3.8 (Preserving a Property)**.** *Given an aggregation rule F and some AF-property Prop, we say that F* preserves *Prop if for every profile P with $att_i \in Prop$, $i \in N$, it holds $F(P) \in Prop$.*

Two basic AF-properties are *acyclicity* and *coherence.* They are very attractive because they reduce the number of possible extensions. If the attack relation of an AF is acyclic, the grounded, stable, preferred, and complete extension coincide, and hence there is exactly one. Coherent AFs are those where preferred and stable semantics coincide. An aggregation rule F *preserves acyclicity (coherence)* if for every profile consisting of acyclic (coherent) individual AFs the outcome of F is also an acyclic (coherent) AF. Obviously, acyclicity is stronger than coherence; however, the results show that preserving acyclicity is easier than preserving coherence.

Theorem 3.9. [**[28]**]

- *Let $|Ar| \geq |N|$. Then for any neutral and independent aggregation rule F that preserves acyclicity, there is at least one agent that has veto powers.*
- *Let $|Ar| \geq 4$. Then any unanimous, grounded, and independent aggregation rule F that preserves coherence is a dictatorship.*

[79] showed that already qualified majority rules preserve acyclicity. Since these rules include an agent with veto powers, this is a special case of the result above. In quota rules, which are independent, no agent has veto powers, hence they do not preserve acyclicity. As an example for the majority rule, consider the simple case of three arguments $\{a, b, c\}$ and three agents with the attack relations $att_1 = \{(a, b), (b, c)\}$, $att_2 = \{(b, c), (c, a)\}$, and $att_3 = \{(c, a), (a, b)\}$. It holds that all

individual AFs are acyclic, but the outcome of the majority rule includes the three attacks $\{(a,b),(b,c),(c,a)\}$ which actually form a cycle. Since all quota rules satisfy the basic axioms, they do not preserve coherence.

When considering the basic semantics, the grounded semantics is the only one that always has a unique solution. In this case, an interesting question is also the preservation of the membership property under the grounded semantics, i.e., whether any argument that is contained in the grounded extension of each individual AF is also contained in the grounded extension of the aggregated AF. For the study of semantics with nonunique extensions, there are different ways to formulate argument acceptability appropriately.

Let σ be the stable, preferred, or complete semantics, F an aggregation rule, and $P = (att_1, \ldots, att_n)$ a profile. We say F *preserves credulous argument acceptability under* σ if for all arguments $a \in Ar$ that belong to some extension under σ for every AF_i, $i \in N$, there is some σ-extension of $F(P)$ that contains a. On the other hand, F is said to *preserve sceptical argument acceptability under* σ if for all arguments $a \in Ar$ that belong to all extensions under σ for every AF_i, $i \in N$, all σ-extensions of $F(P)$ contain a. Since the grounded extension is unique, both notions coincide for this extension.

Unfortunately, argument acceptability in either form is not compatible with a set of very basic axioms, unless we allow dictatorships.

Theorem 3.10. [[**28**]] *Let* $|Ar| \geq 4$ *and* σ *be the stable, preferred, complete, or grounded semantics. Then any unanimous, grounded, and independent aggregation rule* F *that preserves either credulous or sceptical argument acceptability under* σ *is a dictatorship.*

The proof of this theorem relies on similar results on graph aggregation obtained by [47]. This result shows that no quota rule preserves argument acceptability.

Example 3.11. *Consider again the individual attack relations as shown in Figures 3a–3c. The grounded extension for* AF_1*,* AF_2*, and* AF_3 *is* $\{a,c,e,g\}$*. Thus argument* a *is contained in the grounded extension for every individual agent. However, for the argumentation framework resulting from the majority rule (see Figure 3d) the grounded extension is* $\{b,c,e,g\}$ *and thus does not contain argument* a*. This shows that the majority rule does not preserve argument acceptability under grounded semantics.*

The next properties do no longer focus on single arguments but on sets of arguments. Let P be a profile. An aggregation rule F *preserves conflict-freeness (admissibility)* if for all sets of arguments $A \subseteq Ar$ that are conflict-free (admissible) for all AF_i, $i \in N$, it holds that A is also conflict-free (admissible) in $F(P)$. In contrast to the previous impossibility result (Theorem 3.10), the results here are more positive. For conflict-freeness there is a very general result, whereas for admissibility there is at least one reasonable rule that satisfies some basic criteria.

Theorem 3.12. [[**28**]]

- *Every aggregation rule that is grounded preserves conflict-freeness.*
- *Let* $|Ar| \geq 4$*. The nomination rule is the only unanimous, grounded, anonymous, neutral, independent, and monotonic aggregation rule that preserves admissibility.*

A counter-example for the majority rule and the preservation of admissibility rule is given in the following example.

Example 3.13. *Recall the example shown in Figures 3a–3c. It holds that* $\{a, c, e, g\}$ *is admissible for* AF_1*,* AF_2*, and* AF_3*. However, for the framework that results from a majority aggregation (see Figure 3d)* $\{a, c, e, g\}$ *is not admissible since* a *is not defended. This means, the majority rule does not preserve admissibility.*

In addition to preservation of admissibility and conflict-freeness, the question of preservation under a given semantics is interesting. Formally, an aggregation rule F *preserves extensions under the stable (complete, grounded, preferred) semantics* if for all sets $A \subseteq Ar$ that are stable (complete, grounded, preferred) for all AF_i, $i \in N$, it holds that A is also stable (complete, grounded, preferred) in $F(P)$. The results here differ, depending on the semantics considered. For the case of the complete, preferred, and grounded semantics, there is again a negative result that builds up on known results by [47] on graph aggregation. On the other hand, the result for the stable semantics is more positive, as the nomination rule indeed preserves stable extensions.

Theorem 3.14. [[**28**]]

- *Let* $|Ar| \geq 5$*. Then any unanimous, grounded, and independent aggregation rule* F *that preserves either complete or preferred extensions is a dictatorship.*
- *Let* $|Ar| \geq 4$*. Then any unanimous, grounded, and independent aggregation rule* F *that preserves grounded extensions is a dictatorship.*
- *The nomination rule preserves stable extensions.*

The following example shows that the majority rule does not preserve extensions under the grounded, stable, preferred, or complete semantics.

Example 3.15. *Consider again the individual argumentation frameworks shown in Figures 3a–3c. Note that all three attack relations are acyclic, and hence the grounded, stable, preferred, and complete semantics coincide. As mentioned before, the grounded extension for all three argumentation frameworks is* $\{a, c, e, g\}$*. However this is not preserved under majority aggregation, since in the resulting argumentation framework the grounded extension is* $\{b, c, e, g\}$*. Since all attack relations are acyclic, the same example shows that the majority rule does not preserve grounded, stable, preferred, or complete extensions.*

A different view on the preservation of extensions has been taken by [46], who consider different variants of these problems from the point of view of computational complexity. Following their work, [37] propose specific AF aggregation rules.

4 Specific Aggregation Methods in Argumentation

In this section, we survey different specific structural aggregation operators for abstract argumentation frameworks from the literature, starting with the pioneering work on AF aggregation by [31], who use *partial* argumentation frameworks as a supporting notion for a framework of parameterized aggregation operators. Next, we show how the model of *incomplete* argumentation frameworks due to [13] is used to implement a simple, very general structural aggregation operator. Finally, we present the work of [1], who employ *value-based* argumentation frameworks due to [18] as a target formalism of structural aggregation operators, using different values associated with arguments as a possible explanation for the differences in the input AFs. A less in-detail, but broader overview of aggregation operators in formal argumentation can be found in the survey by [20].

4.1 Partial Argumentation Frameworks

[31] introduced the notion of partial argumentation framework specifically as an intermediate format for the implementation of aggregation operators on argumentation frameworks.

Definition 4.1 (Partial Argumentation Framework (PAF))**.** *A* partial argumentation framework (PAF) *is a triple* $PAF = \langle Ar, att, ign \rangle$ *and consists of*

- *a set* Ar *of* arguments*,*
- *an* attack relation $att \subseteq (Ar \times Ar)$ *specifying attacks known to exist,*

- *and an* ignorance relation $ign \subseteq (Ar \times Ar)$ *specifying attacks whose existence is not known.*

It is assumed that $att \cap ign = \emptyset$. *A third relation is the* non-attack relation $non = (Ar \times Ar) \setminus (att \cup ign)$, *which is implicitly given by the other two.*

Aggregating several individual AFs via PAFs is a two-step process. First, each AF is expanded to incorporate the information that is present in the other AFs. An expansion must include the union of all arguments from all AFs. On the other hand, an expansion must respect the information that is present in the original AF; in particular, all arguments and all attacks of the original AF must be present in the expanded AF, too, and also every attack that does not exist in the original AF must be non-existent in the expanded AF. However, this still leaves a lot of freedom, allowing for many different expansion operators. Expansion operators use PAFs as their target format in order to be able to represent ignorance about the existence of attacks.

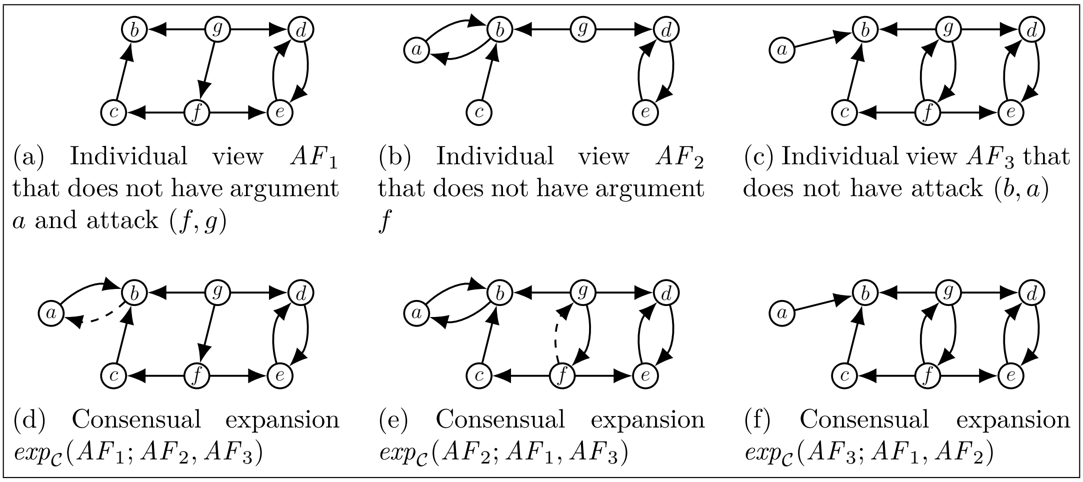
Definition 4.2 (Expansion Operator)**.** *Let* $\mathbb{PAF}$ *denote the universe of all partial AFs. We consider mappings* $exp : \mathbb{AF}^m \to \mathbb{PAF}$ *that map an argumentation framework* $AF = \langle Ar, att \rangle$ *and a profile* $P = (AF_1, \ldots, AF_{m-1})$ *with* $AF_i = \langle Ar_i, att_i \rangle$ *of other individual AFs to an expanded PAF representation* $exp(AF; P) = \langle Ar', att', ign' \rangle$ *of* AF, *that incorporates the information given by the other individual AFs* AF_i. *More formally, exp is called an* expansion operator *if it satisfies the following conditions:*

- $Ar' = Ar \cup \bigcup_{i=1}^{m-1} Ar_i$, *i.e., all arguments from all individual views are included;*
- $att' \supseteq att$, *i.e., all known attacks from* AF *are preserved; and*
- $non' \supseteq (Ar \times Ar) \setminus att$, *i.e., all non-attacks from* AF *are preserved.*

Every agent may choose their own expansion operator, it is not required that all agents use the same one in an aggregation process. In their paper, [31] focus on the *consensual expansion operator*, which is defined as follows.

Definition 4.3 (Consensual Expansion)**.** *The* consensual expansion operator $exp_{\mathcal{C}}$ *maps an* $AF = \langle Ar, att \rangle$ *and a profile* $P = (AF_1, \ldots, AF_{m-1})$ *with* $AF_i = \langle Ar_i, att_i \rangle$ *to a PAF* $\langle Ar', att', ign' \rangle$ *with*

- $Ar' = Ar \cup \bigcup_{i=1}^{m-1} Ar_i$ *(as required),*
- $att' = att \cup \Big(\big(\bigcup_{i=1}^{m-1} att_i \setminus conf(AF, P) \big) \setminus non \Big)$, *and*



(a) Individual view AF_1 that does not have argument a and attack (f, g)

(b) Individual view AF_2 that does not have argument f

(c) Individual view AF_3 that does not have attack (b, a)

(d) Consensual expansion $exp_{\mathcal{C}}(AF_1; AF_2, AF_3)$

(e) Consensual expansion $exp_{\mathcal{C}}(AF_2; AF_1, AF_3)$

(f) Consensual expansion $exp_{\mathcal{C}}(AF_3; AF_1, AF_2)$

Figure 4: Graph visualizations for the consensual expansion in Example 4.4, where attacks in an ignorance relation ign_i are drawn as dashed arrows

- $ign' = conf(AF, P) \setminus (att \cup non)$.

The helper function $conf(AF, P) = (att \cup \bigcup_{i=1}^{m-1} att_i) \cap (non \cup \bigcup_{i=1}^{m-1} non_i)$ *identifies those attacks for which there is a conflicting opinion in the input AFs about whether or not they exist.*

An agent that uses the consensual expansion operator is confident to include new attacks in their set att' of attacks known to exist, when all other agents that have both incident arguments in their individual view agree on the existence of the attack. Likewise, when all other agents that know both incident arguments of an attack agree that it does *not* exist, the agent is confident to include it in their set non' of attacks known to not exist. All other new attacks are included in the ignorance relation ign'.

Example 4.4. *Recall our running example from Table 1 and Figure 2 in Section 2. Consider three individual AFs* $(AF_i = \langle Ar_i, att_i \rangle)_{i \in \{1,2,3\}}$ *that each represent the subjective views of a participant in the example discussion, where*

$$\begin{aligned}
Ar_1 &= \{b, c, d, e, f, g\}, & att_1 &= \{(c, b), (d, e), (e, d), (f, e), (f, c),\\
& & &\quad (g, d), (g, b), (g, f)\},\\
Ar_2 &= \{a, b, c, d, e, g\}, & att_2 &= \{(a, b), (b, a), (c, b), (d, e), (e, d),\\
& & &\quad (g, d), (g, b)\},\\
Ar_3 &= \{a, b, c, d, e, f, g\}, & att_3 &= \{(a, b), (c, b), (d, e), (e, d), (f, e),\\
& & &\quad (f, c), (g, d), (f, g), (g, b), (g, f)\}.
\end{aligned}$$

These individual AFs are illustrated in Figures 4a, 4b, and 4c, respectively. The stable extensions are $\{c, e, g\}$ for AF_1, $\{a, c, e, g\}$ for AF_2, and $\{a, c, e, g\}$ and $\{a, d, f\}$ for AF_3.

When all individual views are expanded using consensual expansion, we obtain the PAFs $PAF_1 = exp_{\mathcal{C}}(AF_1; AF_2, AF_3)$, $PAF_2 = exp_{\mathcal{C}}(AF_2; AF_1, AF_3)$, and $PAF_3 = exp_{\mathcal{C}}(AF_3; AF_1, AF_2)$ that are visualized in Figures 4d, 4e, and 4f.

A profile of expanded individual views can now be aggregated using the *fusion* step. Fusion operators identify a single AF or a set of AFs that is "as close as possible" to all expanded individual views. To implement the notion of "closeness," a fusion operator is parameterized by a pseudo-distance $d : \mathbb{PAF}^2 \to \mathbb{R}^+$ on PAFs—satisfying $d(PAF_1, PAF_2) = d(PAF_2, PAF_1)$ and $d(PAF_1, PAF_2) = 0 \Leftrightarrow PAF_1 = PAF_2$—and an aggregation function $\otimes : \mathbb{R}^m \to \mathbb{R}$ that is used to aggregate m distances into a single score.

As an example, [31] define an edit distance on PAFs, which penalizes outright disagreement about the existence of a relation $r \in (Ar \times Ar)$ with an increase of the distance by 1, while if one of the two PAFs has r in its ignorance relation and the other does not, this is penalized with an increase of 0.5.

Definition 4.5 (Edit Distance on PAFs)**.** *The* edit distance $d_e : \mathbb{PAF}^2 \to \mathbb{R}^+$ on PAFs *is defined as follows:*

$$\begin{aligned}
&d_e(\langle Ar, att_1, ign_1\rangle, \langle Ar, att_2, ign_2\rangle) =\\
&\quad \sum_{r\in(Ar\times Ar)} 1 \cdot (\mathbb{1}_{att_1\cap non_2}(r) + \mathbb{1}_{non_1\cap att_2}(r))\\
&\qquad\qquad + 0.5 \cdot \left(\mathbb{1}_{ign_1\cap att_2}(r) + \mathbb{1}_{ign_1\cap non_2}(r) + \mathbb{1}_{att_1\cap ign_2}(r) + \mathbb{1}_{non_1\cap ign_2}(r)\right),
\end{aligned}$$

where $\mathbb{1}_X$ denotes the indicator function for a set X*, defined by $\mathbb{1}_X(x) = 1$ if $x \in X$, and $\mathbb{1}_X(x) = 0$ otherwise.*

We are now ready to define fusion operators.

Definition 4.6 (Fusion Operator)**.** *Let $m \in \mathbb{N}$, d be a pseudo-distance on PAFs, and $\otimes$ be an aggregation function on $\mathbb{R}^m$. The* fusion operator $\mathit{fusion}_{d,\otimes} : \mathbb{PAF}^m \to 2^{\mathbb{AF}}$ *maps a profile $(PAF_1, \dots, PAF_m)$ of partial AFs obtained through expansion to the set of AFs that minimize the aggregated distance (with respect to d and $\otimes$) to all input PAFs, i.e.,*

$$\begin{aligned}
&\mathit{fusion}_{d,\otimes}(PAF_1, \dots, PAF_m) = \\
&\qquad \{\langle Ar^*, att^* \rangle \mid Ar^* = \bigcup_{i=1}^{m} Ar_i, att^* \subseteq (Ar^* \times Ar^*), \\
&\qquad \textit{and } \langle Ar^*, att^* \rangle \textit{ minimizes } \otimes_{i=1}^{m} \left(d(\langle Ar^*, att^* \rangle, PAF_i)\right)\}.
\end{aligned}$$

An aggregation operator is now obtained by chaining m expansion operators and a fusion operator together.

Definition 4.7 (PAF Aggregation Operator)**.** *Let $m \in \mathbb{N}$. For a profile* $\mathtt{exp} = (\mathit{exp}_1, \dots, \mathit{exp}_m)$ *of m expansion operators, for a given pseudo-distance d on PAFs, and for a given aggregation function $\otimes$ on $\mathbb{R}^m$, the* aggregation operator $\mathit{agg}^{\mathit{pafs}}_{\mathtt{exp},d,\otimes} : \mathbb{AF}^m \to 2^{\mathbb{AF}}$ *is defined as*

$$\begin{aligned}
\mathit{agg}^{\mathit{pafs}}_{\mathtt{exp},d,\otimes}(AF_1, \dots, AF_m) = \mathit{fusion}_{d,\otimes}(&\mathit{exp}_1(AF_1; AF_2, \dots, AF_m), \\
&\dots, \mathit{exp}_m(AF_m; AF_1, \dots, AF_{m-1})).
\end{aligned}$$

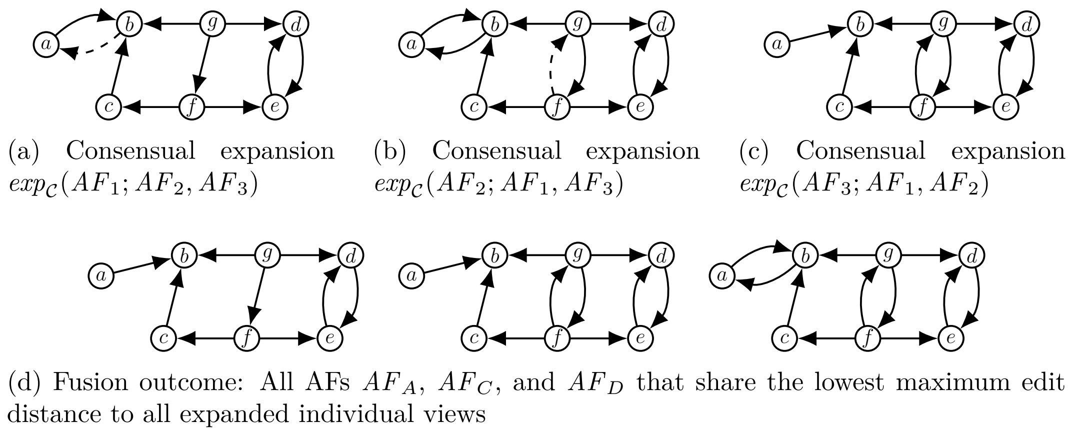
In general, PAF aggregation operators are irresolute, i.e., they may return a set of aggregates instead of a single aggregate, since the fusion operator may find several AFs that share the lowest aggregated distance to the expanded individual views.

Example 4.8. *We continue our example. The expanded views PAF_1, PAF_2, and PAF_3 that we obtained through consensual expansion all share the same set $\{a, b, c, d, e, f, g\}$ of arguments, have ignorance relations $ign_1 = \{(b, a)\}$, $ign_2 = \{(f, g)\}$, and $ign_3 = \emptyset$, and the following attack relations:*

$$\begin{aligned}
att_1 &= \{(a, b), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (g, b), (g, f)\}, \\
att_2 &= \{(a, b), (b, a), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (g, b), (g, f)\}, \\
att_3 &= \{(a, b), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (f, g), (g, b), (g, f)\}.
\end{aligned}$$

For better readability, we display the expanded individual views again in Figures 5a, 5b, and 5c.

For the fusion operator, we use the edit distance d_e, and as an aggregation function we use the maximum function max. *The operator $\mathit{fusion}_{d_e,\max}$ identifies all*



(a) Consensual expansion $exp_{\mathcal{C}}(AF_1; AF_2, AF_3)$

(b) Consensual expansion $exp_{\mathcal{C}}(AF_2; AF_1, AF_3)$

(c) Consensual expansion $exp_{\mathcal{C}}(AF_3; AF_1, AF_2)$

(d) Fusion outcome: All AFs AF_A, AF_C, and AF_D that share the lowest maximum edit distance to all expanded individual views

Figure 5: Graph visualizations for the fusion in Example 4.4, where attacks in an ignorance relation ign_i are drawn as dashed arrows

argumentation frameworks $\langle\{a, b, c, d, e, f, g\}, att\rangle$ for which the maximum of the edit distances to each expanded individual view PAF_1, PAF_2, or PAF_3 is minimal. Among all candidates $\langle\{a, b, c, d, e, f, g\}, att\rangle$, we only need to consider those that share the attacks and non-attacks that all PAF_i agree on, because any deviation would increase the distance to every individual view. The only relations for which there is disagreement among the PAF_i are (b, a) and (f, g). This leaves four candidates AF_A, AF_B, AF_C, and AF_B with attack relations:

$$
\begin{aligned}
att_A &= \{(a, b), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (g, b), (g, f)\},\\
att_B &= \{(a, b), (b, a), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (g, b), (g, f)\},\\
att_C &= \{(a, b), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (f, g), (g, b), (g, f)\},\\
att_D &= \{(a, b), (b, a), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (f, g), (g, b), (g, f)\}.
\end{aligned}
$$

The edit distances are as follows:

- $d_e(AF_A, PAF_1) = 0.5,\ d_e(AF_A, PAF_2) = 1.5,\ d_e(AF_A, PAF_3) = 1,$
- $d_e(AF_B, PAF_1) = 0.5,\ d_e(AF_B, PAF_2) = 0.5,\ d_e(AF_B, PAF_3) = 2,$
- $d_e(AF_C, PAF_1) = 1.5,\ d_e(AF_C, PAF_2) = 1.5,\ d_e(AF_C, PAF_3) = 0,$
- $d_e(AF_D, PAF_1) = 1.5,\ d_e(AF_D, PAF_2) = 0.5,\ d_e(AF_D, PAF_3) = 1,$

AF_B has an edit distance $d_e(AF_B, PAF_3) = 2$ to PAF_3, while the other three AFs share a maximum distance of 1.5 *to any of the PAFs. Therefore, the result of*

the fusion step is

$$\mathit{fusion}_{d_e,\max}(PAF_1, PAF_2, PAF_3) = \{AF_A, AF_C, AF_D\}.$$

These AFs are visualized in Figure 5d. AF_A has $\{a, c, e, g\}$ as its only stable extension, AF_C has stable extensions $\{a, c, e, g\}$ and $\{a, d, f\}$, and AF_D has stable extensions $\{a, c, e, g\}$, $\{a, d, f\}$, and $\{b, d, f\}$. In particular, we can observe that the aggregate has an extension where argument b is accepted, although b is not accepted in any of the individual views.

The fact that the aggregation results can have acceptable (sets of) arguments that are not acceptable in any individual view bears some similarities with the discursive dilemma from judgment aggregation theory,[1] where a profile of individually consistent judgment sets can be aggregated into a single judgment via some majority criterion such that this aggregated judgment contains a conclusion that is not accepted in any of the individual judgments. Since we are concerned with framework-based structural aggregation, we regard the information provided by the individual views (i.e., the existing arguments and attacks) as primarily relevant, overshadowing the outcome (i.e., extensions and accepted arguments) of the individual views. Like premise-based aggregation rules in judgment aggregation (see [60], [39]), structural aggregation operators for AFs allow the outcome of the aggregate to have precedence over the aggregation of the outcomes.

4.2 Incomplete Argumentation Frameworks

Incomplete Argumentation Frameworks (IAFs) further generalize PAFs and can represent uncertainty about the existence of individual arguments [17], uncertainty about the existence of individual attacks [11], or both simultaneously [13].

Definition 4.9 (Incomplete Argumentation Framework)**.** *An* incomplete argumentation framework (IAF) *is a quadruple $IAF = \langle Ar, Ar^?, att, att^? \rangle$ and consists of*

- *a set Ar of* definite arguments *known to exist,*
- *a set $Ar^?$ of* uncertain arguments *of which each may or may not exist,*
- *a set $att \subseteq (Ar \cup Ar^?) \times (Ar \cup Ar^?)$ of* (conditionally) definite attacks *that exist if and only if both incident arguments exist, and*

[1] Generalizing the *doctrinal paradox* due to [59], [71] introduced the *discursive dilemma*; both are discussed by [68] in more detail.

- *a set* $att^{?} \subseteq (Ar \cup Ar^{?}) \times (Ar \cup Ar^{?})$ *of* uncertain attacks *of which each may or may not exist, but only if both incident arguments exist.*

We assume that $Ar \cap Ar^{?} = \emptyset$ *and* $att \cap att^{?} = \emptyset$*.*

An IAF is a compact representation of a set of possible worlds—namely, all the standard AFs that can be obtained from it by deciding for each uncertain element whether or not it should be included. Each such AF is called a *completion* of the IAF.

Definition 4.10 (Completion)**.** *Let* $IAF = \langle Ar, Ar^{?}, att, att^{?}\rangle$ *be an incomplete AF. An AF* $AF^{*} = \langle Ar^{*}, att^{*}\rangle$ *is called a* completion of *IAF if it satisfies*

$$Ar \subseteq Ar^{*} \subseteq Ar \cup Ar^{?} \text{ and}$$
$$att|_{Ar^{*}} \subseteq att^{*} \subseteq \left(att \cup att^{?}\right)|_{Ar^{*}}.$$

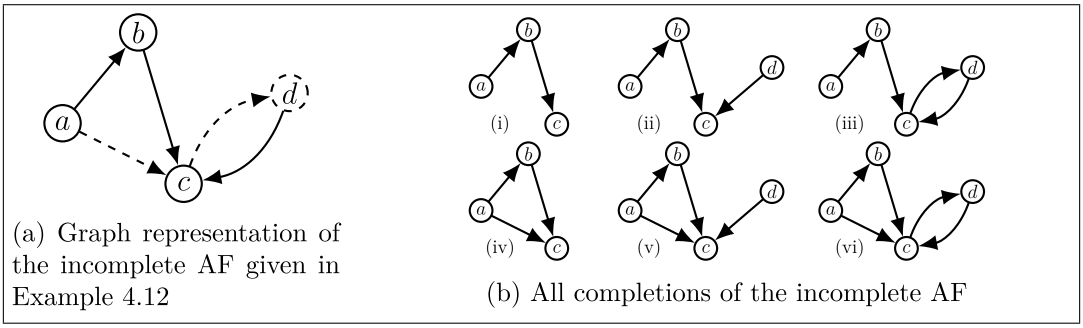
That is, every completion of IAF must include each of its definite arguments and may include any of its uncertain arguments. Attacks can only be included if both incident arguments are included (indicated by the restriction operator $att'|_{Ar'} = att' \cap (Ar' \times Ar')$*). Every conditionally definite attack that has both incident arguments present in a completion, must be included in that completion. Uncertain attacks may be included, but only if both incident arguments are included.*

Incomplete AFs define acceptability criteria for sets of arguments or for individual arguments, which are derived from the standard AF criteria of extensions and credulous/skeptical acceptance in the completions of an IAF.

Definition 4.11 (Acceptability in IAFs)**.** *Let* $IAF = \langle Ar, Ar^{?}, att, att^{?}\rangle$ *be an incomplete AF and let* σ *be a semantics.*

- *A set* $S \subseteq Ar \cup Ar^{?}$ *of arguments in IAF is* accepted possibly *(respectively,* necessarily*) for semantics* σ *if* S *is a* σ *extension in some completion of IAF (respectively, in all completions of IAF)*[2]*. This notion of acceptability is derived from the verification problem (*σ-Ver*) for standard AFs and formalized via the problems of possible verification (*σ-PV*) and necessary verification (*σ-NV*) by [13] and [48].*

[2] This is a revised definition due to [48], who point out some problematic behavior of the initial definition by [13], where it was only required that $S \cap Ar^{*}$ (instead of S) is a σ extension in some completion of IAF in order for S to be accepted in that completion. In this chapter, we focus on the revised definition.



(a) Graph representation of the incomplete AF given in Example 4.12

(b) All completions of the incomplete AF

Figure 6: Incomplete AF and corresponding completions for Example 4.12

- *A single argument $a \in Ar \cup Ar^?$ in IAF is* possibly credulously accepted *(respectively,* necessarily credulously accepted*) for semantics σ if a is a member of some σ extension S in some completion of IAF (respectively, in all completions of IAF). Similarly to verification, this notion is derived from the credulous acceptance problem (σ-*CA*) for standard AFs and formalized via the problems of possible credulous acceptance (σ-*PCA*) and necessary credulous acceptance (σ-*NCA*) by [12].*

- *A single argument $a \in Ar \cup Ar^?$ in IAF is* possibly skeptically accepted *(respectively,* necessarily skeptically accepted*) for semantics σ if a is a member of all σ extensions S in some completion of IAF (respectively, in all completions of IAF). Again, this notion is derived from the skeptical acceptance problem (σ-*SA*) for standard AFs and formalized via the problems of possible skeptical acceptance (σ-*PSA*) and necessary skeptical acceptance (σ-*NSA*) by [12].*

Note that the problems of possible/necessary acceptability in IAFs can be restricted to definite targets (i.e., $S \subseteq Ar$ for sets of arguments or $a \in Ar$ for single arguments) without changing their complexity: For "necessary" problem variants, we get a trivial "no" answer if (part of) the target is uncertain, since then it cannot be present in all completions. For "possible" problem variants, we can disregard all completions that do not contain all target arguments, since they cannot produce "yes" answers. The remaining cases constitute exactly the original problem where all target arguments are definite.

Example 4.12. *Consider an incomplete AF $IAF = \langle Ar, Ar^?, att, att^? \rangle$ with three definite arguments in $Ar = \{a, b, c\}$, an uncertain argument in $Ar^? = \{d\}$, three (conditionally) definite attacks in $att = \{(a, b), (b, c), (d, c)\}$, and two uncertain attacks in $att^? = \{(a, c), (c, d)\}$. Its graph representation is given in Figure 6a, where* *uncertain elements are distinguished by dashed lines, and all its completions are given in Figure 6b.*

When we determine the stable extensions of the completions, we see that completion (i) *has* $\{a, c\}$ *as its only stable extension, completion* (iii) *has* $\{a, c\}$ *and* $\{a, d\}$ *as its stable extensions, completion* (iv) *has* $\{a\}$ *as its only stable extension, and all other completions—namely,* (ii)*,* (v)*, and* (vi)*—have* $\{a, d\}$ *as their only stable extension. For the stable semantics, this means that:*

- $\{a\}$*,* $\{a, c\}$*, and* $\{a, d\}$ *are possibly accepted, while no other set of arguments is possibly accepted, and no set of arguments is necessarily accepted.*

- *Argument* a *is necessarily skeptically accepted, since it is a member of every stable extension in every completion.*

- *Argument* b *is not possibly credulously accepted, since it is not a member of any stable extension in any completion.*

- *Argument* c *is possibly skeptically accepted, since in completion* (i)*,* c *is a member of all stable extensions. However,* c *is not necessarily credulously accepted, since there are completions—e.g.,* (iv)*—where* c *is not included in any stable extension.*

The computational complexity of decision problems that capture these notions of acceptability was analyzed by [13] and [48] for sets of arguments and by [12] and [69] for individual arguments.[3] Table 2 gives an overview of the complexities compared to the respective complexity of the base problem for standard AFs without uncertainty. Results marked with $^{\spadesuit}$ are by [42], marked with $^{\clubsuit}$ are by [13], marked with $^{\bullet}$ are by [48], marked with $^{\bigstar}$ are by [40], marked with $^{\blacklozenge}$ are by [12], marked with $^{\blacktriangle}$ are by [32], marked with $^{\blacktriangledown}$ are by [69], marked with † are by [73], and marked with ‡ are by [43]. Results marked with an asterisk (*) are straight-forward and are not formally proven. We can observe various cases where the computational hardness is not increased by the introduction of uncertainty—in particular, for necessary verification (σ-NV). Further, experimental results by [69] indicate that even the hard cases (i.e., those problems that are complete for NP, coNP, or a class even higher in the polynomial hierarchy) may be tamed in practice through suitable encodings.

Even though incomplete AFs were conceived as a very general representation of various different kinds of structural uncertainty in AFs, in particular they can serve as a target formalism for structural aggregation of different individual AFs.

[3] We mention in passing that similarly to the way [13] study possible and necessary variants of the *verification problem*, [77] have done so for the *existence problem*.

σ	σ-Ver	σ-PV	σ-NV
$\mathcal{CF}$	$\in \mathrm{P}$ ♠	$\in \mathrm{P}$ ♣	$\in \mathrm{P}$ ♣
$\mathcal{AD}$	$\in \mathrm{P}$ ♠	$\in \mathrm{P}$ •	$\in \mathrm{P}$ ♣
$\mathcal{ST}$	$\in \mathrm{P}$ ♠	$\in \mathrm{P}$ •	$\in \mathrm{P}$ ♣
$\mathcal{CO}$	$\in \mathrm{P}$ ♠	$\in \mathrm{P}$ •	$\in \mathrm{P}$ ♣
$\mathcal{GR}$	$\in \mathrm{P}$ ♠	$\in \mathrm{P}$ •	$\in \mathrm{P}$ ♣
$\mathcal{PR}$	coNP-c. ★	Σ_2^p-c. ♣	coNP-c. ♣
σ	σ-CA	σ-PCA	σ-NCA
$\mathcal{CF}$	$\in \mathrm{P}$ *	$\in \mathrm{P}$ ♦	$\in \mathrm{P}$ ♦
$\mathcal{AD}$	NP-c. ★	NP-c. ♦	Π_2^p-c. ♦
$\mathcal{ST}$	NP-c. ★	NP-c. ♦	Π_2^p-c. ♦
$\mathcal{CO}$	NP-c. ▲	NP-c. ♦	Π_2^p-c. ♦
$\mathcal{GR}$	$\in \mathrm{P}$ *	NP-c. ♦	coNP-c. ♦
$\mathcal{PR}$	NP-c. ★	NP-c. ♦	Π_2^p-c. ♦
σ	σ-SA	σ-PSA	σ-NSA
$\mathcal{CF}_{\neq\emptyset}$	$\in \mathrm{P}$ ▼	$\in \mathrm{P}$ ▼	$\in \mathrm{P}$ ▼
$\mathcal{AD}_{\neq\emptyset}$	coNP-c. †	Σ_2^p-c. ▼	coNP-c. †
$\mathcal{ST}$	coNP-c. ★	Σ_2^p-c. ♦	coNP-c. ♦
$\mathcal{CO}$	$\in \mathrm{P}$ ▲	NP-c. ♦	coNP-c. ♦
$\mathcal{GR}$	$\in \mathrm{P}$ *	NP-c. ♦	coNP-c. ♦
$\mathcal{PR}$	Π_2^p-c. ‡	Σ_3^p-c. ♦	Π_2^p-c. ♦

Table 2: Overview of the computational complexity of decision problems in standard and incomplete argumentation frameworks for the six original semantics. The results are grouped into verification variants (top), credulous acceptance variants (center), and skeptical acceptance variants (bottom). Each block has the semantics in the left-most column, the base problem for standard AFs in the second column, and the "possible" and "necessary" variants of the base problem in the third and fourth column, respectively. For a complexity class $\mathcal{C}$, we write $\mathcal{C}$-c. to denote completeness for $\mathcal{C}$. For the problem σ-SA and its generalizations, only nonempty conflict-free ($\mathcal{CF}_{\neq\emptyset}$) and nonempty admissible ($\mathcal{AD}_{\neq\emptyset}$) sets are considered, since these problems are trivial for general $\mathcal{CF}$ or $\mathcal{AD}$ sets.

In the following, we define and illustrate the aggregation operator from sets of AFs to a single IAF as introduced by [13]. This aggregation operator is very liberal and imposes only a single condition: When all individual AFs agree on the existence (respectively, nonexistence) of an element, then this element must exist (respectively, cannot exist) in the aggregate. Arguments or attacks for which there is disagreement are included as uncertain elements, thus allowing to include or exclude them via completions.

Definition 4.13 (IAF Aggregation Operator)**.** *Denote with $\mathbb{IAF}$ the universe of all incomplete argumentation frameworks. For every $m \in \mathbb{N}$, the* aggregation operator *$agg^{inc} : \mathbb{AF}^m \to \mathbb{IAF}$ maps any set $\{AF_1, \dots, AF_m\}$ with $AF_i = \langle Ar_i, att_i \rangle$ of m individual AFs to $IAF = \langle Ar, Ar^?, att, att^? \rangle$ with*

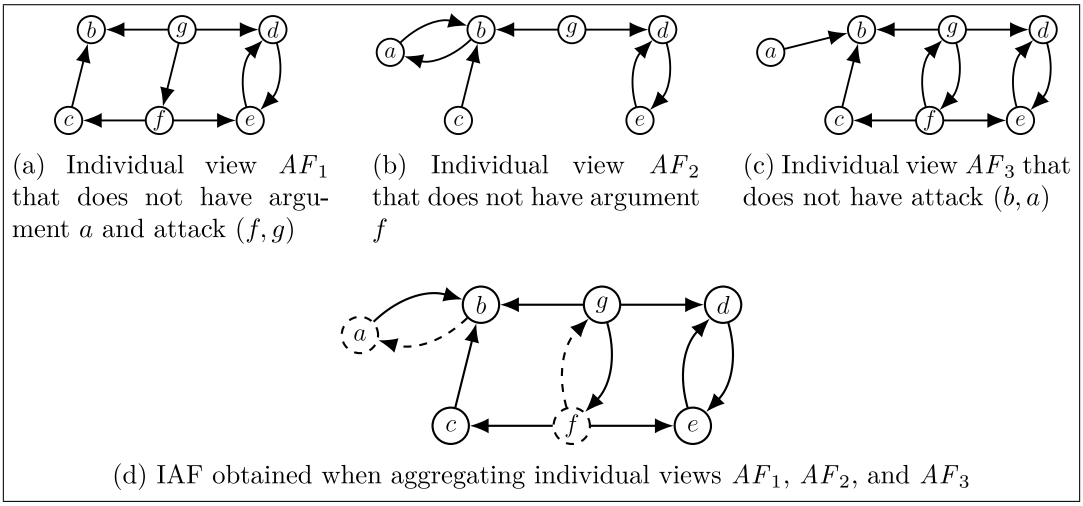
$$\begin{aligned} Ar &= \bigcap_{i=1}^{m} Ar_i, & att =&\{(a,b) \in (Ar \cup Ar^?) \times (Ar \cup Ar^?) \mid \\ & & &(\forall i \in \{1, ..., m\})\, [a, b \in Ar_i \Rightarrow (a,b) \in att_i]\}, \\ Ar^? &= \left(\bigcup_{i=1}^{m} Ar_i \right) \setminus Ar, & att^? =& \left(\bigcup_{i=1}^{m} att_i \right) \setminus att. \end{aligned}$$

That is, every argument that occurs in every individual AF is included as a definite argument, and all other arguments that occur in individual AFs are included as possible arguments. An attack is included as (conditionally) definite attack if it occurs in every individual AF that includes both incident arguments, and as a possible attack otherwise.

Example 4.14. *Again, we use our running example with the same three individual AFs $(AF_i = \langle Ar_i, att_i \rangle)_{i \in \{1,2,3\}}$ that each represent the subjective views of a participant in the example discussion, where*

$$\begin{aligned} Ar_1 &= \{b, c, d, e, f, g\}, & att_1 &= \{(c,b), (d,e), (e,d), (f,e), (f,c), \\ & & &\quad (g,d), (g,b), (g,f)\}, \\ Ar_2 &= \{a, b, c, d, e, g\}, & att_2 &= \{(a,b), (b,a), (c,b), (d,e), (e,d), \\ & & &\quad (g,d), (g,b)\}, \\ Ar_3 &= \{a, b, c, d, e, f, g\}, & att_3 &= \{(a,b), (c,b), (d,e), (e,d), (f,e), \\ & & &\quad (f,c), (g,d), (f,g), (g,b), (g,f)\}. \end{aligned}$$

For better readability, the individual AFs are illustrated again in Figures 7a, 7b, and 7c. The stable extensions are $\{c, e, g\}$ for AF_1, $\{a, c, e, g\}$ for AF_2, and $\{a, c, e, g\}$ and $\{a, d, f\}$ for AF_3.



(a) Individual view AF_1 that does not have argument a and attack (f, g)

(b) Individual view AF_2 that does not have argument f

(c) Individual view AF_3 that does not have attack (b, a)

(d) IAF obtained when aggregating individual views AF_1, AF_2, and AF_3

Figure 7: Graph representation of the individual AFs in Example 4.14

Aggregating these AFs with the method of Definition 4.13 yields the IAF IAF = $\langle Ar, Ar^?, att, att^? \rangle$ *displayed in Figure 7d, with:*

$$\begin{aligned}
Ar &= \{b, c, d, e, g\},\\
Ar^? &= \{a, f\},\\
att &= \{(a, b), (c, b), (d, e), (e, d), (f, e), (f, c), (g, d), (g, b), (g, f)\},\\
att^? &= \{(b, a), (f, g)\}.
\end{aligned}$$

Arguments a and f do not occur in every individual AF and are therefore uncertain arguments, all other arguments are definite arguments. For the attacks, we have three different cases: Everyone agrees on the existence of the attacks (c, b), (d, e), (e, d), (g, d), and (g, b), so these are definite attacks. For the attacks (a, b), (f, e), (f, c), and (g, f), not all individual AFs include both incident arguments, but those that do, agree on their existence, so these are conditionally definite attacks (which, like definite attacks, are included in the set att, but they might vanish alongside incident uncertain arguments in a completion). For the remaining attacks (b, a) and (f, g), there is disagreement about their existence among individual views who include both incident arguments, so these are uncertain attacks.

IAF has the following nine completions.

- *One includes neither of the arguments a or f and has $\{c, e, g\}$ as its only stable extension.*

- *Two include argument a, but not f, and both have $\{a, c, e, g\}$ as their only stable extension.*
- *Two include argument f, but not a, both have $\{c, e, g\}$ as a stable extension, and one of them has $\{b, d, f\}$ as a second stable extension.*
- *The remaining four completions include bot arguments a and f. All have $\{a, c, e, g\}$ as a stable extension, two of them have $\{a, d, f\}$ as a second stable extension, and one of these two has $\{b, d, f\}$ as a third stable extension.*

As in the example used in the previous section, we can again observe that IAF has completions where argument b is accepted (i.e., b is possibly credulously accepted in IAF), even though b is not accepted in any of the individual views.

Since the IAF aggregation operator produces a single IAF as its output, and every IAF is a compact representation of a set of AFs, the operator can be seen as an irresolute AF aggregation operator. However, the existing acceptability semantics for IAFs make it unnecessary to resolve the remaining ambiguity through some sort of fusion, since collective acceptability can already be determined at this stage. One could say that the IAF aggregation operator circumvents the issues inherent to aggregation by skipping the actual aggregation altogether. Instead of being a single aggregate, an IAF that represents a set of individual AFs can be seen as a kind of convex closure of those individual views. Every reasonable structural aggregate of the individual views will lie within that closure (i.e., will be a completion of the IAF). As such, the acceptability semantics for the IAF provide a way to define a very cautious collective acceptability for the individual views, which avoids the deliberate choice of a specific aggregate AF.

4.3 Value-Based Argumentation Frameworks

[18] proposed the notion of (audience-specific) value-based argumentation framework (see also Chapter 5 of this handbook [4]).

Definition 4.15 (Value-Based Argumentation Framework).
An audience-specific value-based argumentation framework (AVAF) *is a quintuple $VAF_\alpha = \langle Ar, att, V, val, \succ_\alpha \rangle$, where*

- *$\langle Ar, att \rangle$ is an AF (without self-attacks, i.e., att is supposed to be irreflexive),*
- *V is a nonempty set of (social or moral) values,*
- *$val : Ar \to V$ is a mapping assigning values to arguments, and*

- $\succ_\alpha$ *is a transitive and asymmetric (thus, in particular, irreflexive) relation reflecting the value preferences of audience* α *on* V*.*

A value-based argumentation framework (VAF) *is similarly defined as a quintuple* $\langle Ar, att, V, val, \succ \rangle$*, where* Ar*,* att*,* V*, and* val *have the same meaning as above but* $\succ$ *is the set of all possible audiences, i.e., of all possible preferences on* V*.*

If V is a singleton, or if no preferences among the values in V are expressed, an AVAF degenerates to an ordinary AF. In an AVAF, each audience α can be identified with its preference relation $\succ_\alpha$ over values in V, and while the function *val* mapping arguments to values is fixed for everyone, the preferences $\succ_\alpha$ over values are specific to this particular audience (or agent) α. The point is that an attack in an AVAF succeeds only if the attacked argument is not preferred to the attacking argument by the audience: From α's point of view, an argument of lower (social or moral) value cannot defeat an argument of higher value. Note that if both arguments are assigned the same value, or if there is no preference between two values, an attack between such arguments does succeed.

Definition 4.16 (Defeat Relation)**.** *Let* $VAF_\alpha = \langle Ar, att, V, val, \succ_\alpha \rangle$ *be an AVAF. An argument* $a \in Ar$ α-defeats *an argument* $b \in Ar$ *if and only if* a *attacks* b *in* $\langle Ar, att \rangle$ *and not* $val(b) \succ_\alpha val(a)$*.*

Example 4.17. *For illustration, we extend the AF* $\langle Ar, att \rangle$ *from our running example (recall Table 1 and Figure 2 in Section 2) as follows. Suppose that*

- *the arguments* a *and* g *are mapped to the value* PERSONAL SAFETY *(represented by white vertices in Figure 8);*

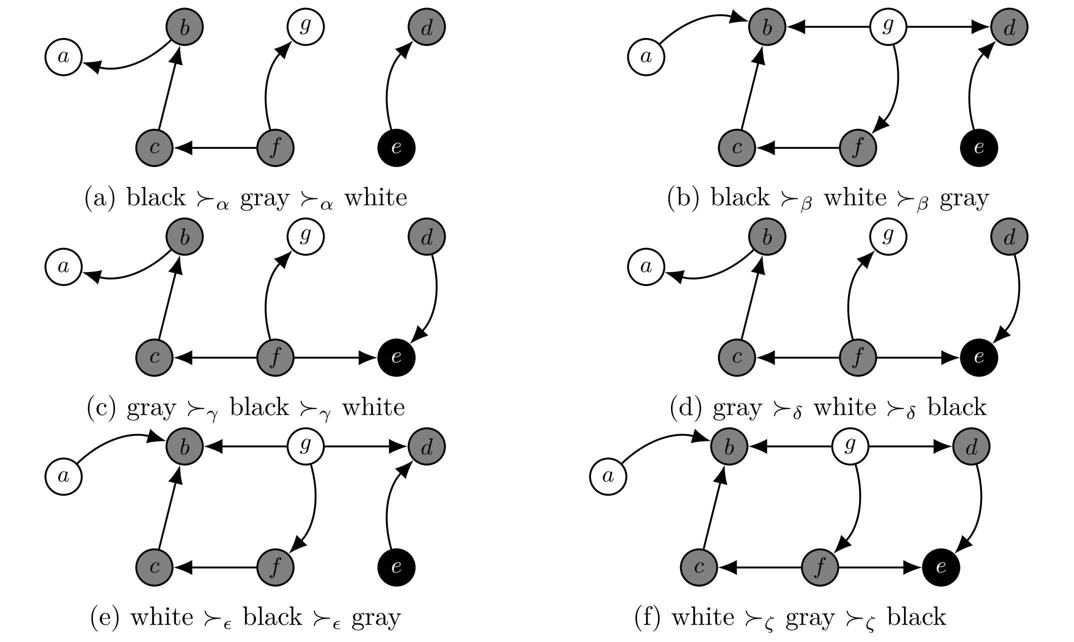
- *the arguments* b*,* c*,* d*, and* f *are mapped to the value* COLLECTIVE SAFETY *(represented by gray vertices in Figure 8); and*

- *the argument* e *is mapped to the value* RIGHT TO BE INFORMED *(represented by a black vertex in Figure 8).*

The six graphs displayed in Figure 8 show the six AVAFs resulting from each possible complete, strict preference ranking of these three values in

$$\succ = \{\succ_\alpha, \succ_\beta, \succ_\gamma, \succ_\delta, \succ_\eta, \succ_\zeta\},$$

where the arguments are vertices and a directed edge from x *to* y *in* VAF_ψ *with* $\succ_\psi \in \succ$ *means that* x ψ*-defeats* y*.*



(a) black $\succ_\alpha$ gray $\succ_\alpha$ white

(b) black $\succ_\beta$ white $\succ_\beta$ gray

(c) gray $\succ_\gamma$ black $\succ_\gamma$ white

(d) gray $\succ_\delta$ white $\succ_\delta$ black

(e) white $\succ_\epsilon$ black $\succ_\epsilon$ gray

(f) white $\succ_\zeta$ gray $\succ_\zeta$ black

Figure 8: Defeat relation in six AVAFs for Example 4.17.

For example, audience α ranks these values by black $\succ_\alpha$ gray $\succ_\alpha$ white, so the attacks (a, b), (d, e), (f, e), (g, b), (g, d), and (g, f) do not succeed in VAF_α because the attackers have lower value than the attackees in these cases and thus they do not defeat them.

The α-defeat relation from Definition 4.16 is an irreflexive binary relation on Ar, just as an attack relation on Ar. Therefore, an AVAF $VAF_\alpha = \langle Ar, att, V, val, \succ_\alpha \rangle$ induces another AF $\langle Ar, \alpha\text{-}defeat \rangle$ with *α-defeat* being a subrelation of *att*.

We thus can define the same semantics that were introduced for AFs in Section 2 for AVAFs as well. For example, a set S of arguments is said to be *conflict-free for audience α* if no argument in S α-defeats any other argument in S: For any two arguments $a, b \in S$, if a attacks b then $val(b) \succ_\alpha val(a)$.[4] Further, S is *admissible for audience α* if it is conflict-free for α and for each argument $a \in S$ and for each argument $b \in Ar$ such that b α-defeats a, there is an argument $c \in S$ that α-defeats b.

[4] This definition of conflict-free sets in preference-based AFs—i.e., AFs which feature a defeat relation derived from the attack relation and some preference over arguments—has been criticized by [2] and [67], who argue that sets of arguments that attack each other should not be considered conflict-free, even when none of the internal attacks succeed, i.e., are defeats. Following the work on rationalizability by [1], however, in this chapter we stick to the original definition.

If S is a maximal (with respect to set inclusion) admissible set for audience α, it is said to be *preferred by audience* α. And S is *stable for audience* α if it is conflict-free for α and every argument outside of S is α-defeated by some argument in S.

Note that for any AVAF VAF_α having no cycle in which all arguments are assigned the same value, the associated AF $\langle Ar, \alpha\text{-}defeat\rangle$ will have no cycle (as any cycle would be broken when an argument of lower value attacks an argument of higher value).[5] In such cases, there exists a unique, nonempty preferred extension for audience α.

Example 4.18. *Continuing Example 4.17, the six AVAFs from Figure 8 have the following preferred extensions for their audiences:*

- *{b, e, f} for Figure 8a with value preference black $\succ_\alpha$ gray $\succ_\alpha$ white;*
- *{a, c, e, g} for Figure 8b with value preference black $\succ_\beta$ white $\succ_\beta$ gray, Figure 8e with value preference white $\succ_\epsilon$ black $\succ_\epsilon$ gray, and Figure 8f with value preference white $\succ_\zeta$ gray $\succ_\zeta$ black; and*
- *{b, d, f} for Figure 8c with value preference gray $\succ_\gamma$ black $\succ_\gamma$ white and Figure 8d with value preference gray $\succ_\delta$ white $\succ_\delta$ black.*

Note that none of the arguments in the AVAFs of Example 4.18 is preferred by every audience, yet they all are preferred by some audience. [18] refers to these properties as objective and subjective acceptability:

Definition 4.19 (Objective and Subjective Acceptability)**.**
Let $\langle Ar, att, V, val, \succ\rangle$ be a VAF. An argument $a \in Ar$ is

- objectively acceptable *if a is contained in a preferred extension of every audience in $\succ$;*
- subjectively acceptable *if a is contained in a preferred extension of some audience in $\succ$; and*
- indefensible *if a is neither objectively nor subjectively acceptable (e.g., in case a is attacked by an objectively acceptable argument with the same value).*

[5] Indeed, [18] argues that single-valued cycles in a VAF indicate that the reasoning giving rise to them must be flawed. He further points out that, while in standard AFs even cycles are in fact required (in particular, two-cycles help to deal with uncertain and incomplete information) and odd cycles are at least plausible [42], cycles should be avoided in value-based argumentation: Odd cycles in VAFs are like *paradoxes* indicating that nothing can be believed, and even cycles are like *dilemmas* requiring a choice between alternatives to be made.

For AVAFs having no single-valued cycle, the preferred extension can be efficiently computed, as it is unique (which greatly simplifies the situation because, recalling Definition 4.11, there is no difference between skeptical and credulous acceptance). For VAFs, however, with no audience specified, all possible audiences need to be taken into account for the problem of determining whether an argument is objectively acceptable, subjectively acceptable, or indefensible. [18] discusses this problem, focusing on certain simple cases, such as when the number of values is two.

Let us have a look at Figure 8 again. We know that all six AVAFs displayed there result from one and the same AF $\langle Ar, att\rangle$, each according to another audience. We started from $\langle Ar, att\rangle$ and then created the six AVAFs by removing certain attacks (namely, those attacks where the attacked argument had a higher value than the attacking argument).

Now, let us ask the converse question: Suppose we are given (or observe) a number of AFs, not necessarily over the same set of arguments and each with its own attack relation. Suppose further that these attack relations are, in fact, defeat relations (i.e., the observed AFs, in fact, are each associated with an AVAF in the way described after Example 4.18). The question is whether these AVAFs can be *rationalized*, i.e., whether they can be explained in terms of a single common master AF, a common set of values, and a common value function, together with a profile of individual preference orders, one for each agent. This *rationalizability problem* has been introduced and studied by [1]. We now define their notion more formally; in fact, we define an entire class of rationalizability problems, parameterized by a set of constraints that the solutions are required to fulfill.

Definition 4.20 (Rationalizability).
A profile $\langle\langle Ar_1, 1\text{-}defeat\rangle, \langle Ar_2, 2\text{-}defeat\rangle, \ldots, \langle Ar_n, n\text{-}defeat\rangle\rangle$ *of AFs*[6] *is said to be* rationalizable by an AVAF *for a given set of constraints (some to be specified below) if there are*

- *an attack relation att on* $Ar = \bigcup_{i=1}^{n} Ar_i$*,*
- *a nonempty set* V *of (social or moral) values,*
- *a mapping* $val : Ar \to V$ *assigning values to arguments, and*
- *a profile* $\langle \succeq_1, \succeq_2, \ldots, \succeq_n\rangle$ *of preference orders on* V*,*[7] *each meeting the given constraints,*

[6] For notational convenience, we from now on use numbers $1, 2, \ldots, n$ instead of Greek letters to denote agents or audiences.

[7] Unlike [18], [1] use *preorders* (which are reflexive, transitive binary relations) or *weak orders* (which, in addition, are complete) for the agents' value preferences. The strict part of a preorder is denoted as $\succ$.

such that for all agents i, $1 \leq i \leq n$, and for any two arguments $a, b \in Ar_i$, a i-defeats b if and only if a attacks b in $\langle Ar, att \rangle$ and not $val(b) \succ_i val(a)$.

For a given rationalizable profile, $\langle Ar, att \rangle$ is referred to as its master AF *and att as its* master attack relation*.*

Examples of types of *constraints* considered by [1] are:

- the master attack relation *att* may be required to be fixed,
- the set V of values and the value function $val : Ar \to V$ may be required to be fixed,
- the number of values in V may be bounded by some constant, and
- the value preferences $\succeq_i$ may be required to be weak orders.

[1] study the question of whether it is possible, given some set of constraints, to characterize the class of profiles of AFs that can be rationalized by an AVAF for these constraints. They also investigate the computational complexity of the rationalizability problem that asks whether a given profile of AFs is rationalizable by an AVAF for a given set of constraints.

They start with the case of a single agent (where profiles contain just a single AF) and first observe that every single AF is rationalizable when there are no constraints, so rationalizability is trivial in this case. Next, they show that the single-agent rationalizability problem can be solved efficiently in many cases.

Theorem 4.21. [**[1]**] *For a single agent, the rationalizability problem can be solved in (deterministic) polynomial time whenever any of the following sets of constraints is given:*

- *the master attack relation is fixed;*
- *the master attack relation, the set of values, and the value function are fixed; and*
- *the master attack relation, an upper bound on the number of values is given, and the agent has a weak preference order.*

For multiple agents, however, the situation is more complicated. For example, the easy observation that every single AF is rationalizable in the absence of constraints does not carry over to profiles with more than one agent, as [1] demonstrate with the following example.

Example 4.22. *Suppose there are two agents,* 1 *and* 2*, discussing three arguments, a, b, and c. While agent* 1 *thinks they form a cycle, a* 1*-defeating b, b* 1*-defeating c, and c* 1*-defeating a, agent* 2 *believes that the three arguments are isolated, i.e., the* 2*-defeat relation is empty. Of course, for this profile to be rationalizable by any AVAF (without any constraints), a master attack relation would have to include at least the attacks from a to b, from b to c, and from c to a. However, for agent* 2 *to be able to remove these attacks in her* 2*-defeat relation,* 2*'s preference would have to include at least the comparisons* $val(a) \succ_2 val(c)$*,* $val(c) \succ_2 val(b)$*, and* $val(b) \succ_2 val(a)$*, which by transitivity of* $\succ_2$ *implies* $val(a) \succ_2 val(a)$*, a contradiction. Hence, even when there are no constraints whatsoever, this profile is not rationalizable by any AVAF.*

While the rationalizability problem is not trivial when there are no constraints in the multiagent case, [1] show that it can still be solved efficiently in this case and also when the only constraints are that the master attack relation and/or the value set and value function are fixed. In their proof, they show that any multiagent rationalizability problem can be decomposed into a set of n single-agent rationalizability problems that can be solved independently of each other by Theorem 4.21.

It follows that, among the constraints they consider, a true multiagent rationalizability problem can be obtained only when an upper bound on the number of values is given. For this problem in general, they have the following result that is obtained by a reduction from the 3-colorability problem, one of the standard NP-complete problems [52].

Theorem 4.23. [**[1]**] *For multiple agents, the rationalizability problem is* NP*-complete whenever a fixed master attack relation and an upper bound of at least three on the number of values are given.*

Whether Theorem 4.23 also holds for an upper bound of at least *two* values, or whether the rationalizability problem becomes tractable in this case, is an interesting open problem. However, when the agents share a *common* set of arguments, we obtain an efficiently solvable special case of the general problem.

Theorem 4.24. [**[1]**] *For multiple agents, the rationalizability problem can be solved in (deterministic) polynomial time whenever a fixed master attack relation is given and there are at most two values.*

As a recommendation for further reading, [63] study two approaches for agents with individual AVAFs to arrive at a common consensus: They may either seek to aggregate their preferences on values (making use of preference aggregation techniques such as voting [16]), or they may seek to aggregate their defeat graphs (making use of the graph aggregation techniques proposed by [47]). Exploring the strengths and limitations of both approaches separately, [63] also propose a third option that combines these two approaches and thus avoids some drawbacks that they may have on their own.

5 Conclusion and Outlook

In this chapter, we highlighted the problem of collective acceptability in formal argumentation and its relations to computational social choice theory. We formally defined structural aggregation for abstract argumentation frameworks, surveyed axiomatic evaluation criteria for aggregation operators, and showcased three specific structural aggregation methods from the literature in more detail.

Beyond the structural aggregation operators covered here, there are some that employ numerical weights to implement the aggregation. [37] define an aggregation operator using weighted AFs [45] with weights on attacks as a refinement of the PAF aggregation operator by [31]. [51] propose an aggregation operator for AFs that uses weights on arguments and attacks to fuel a system of equations that determines the collectively most acceptable arguments. [55] recently initiated the study of aggregation operators for probabilistic AFs [62].

[80] goes beyond the purely abstract perspective of AFs and proposes constraints on aggregation operators that can incorporate information which has been abstracted away in the AF representation.

It is an interesting open task to verify axiomatic properties for structural AF aggregation operators—e.g., the properties presented in Section 3—for the aggregation methods presented in Section 4.

Acknowledgments

This work was supported in part by DFG grants BA 6270/1-1, RO 1202/14-2, and RO 1202/21-1 and by the project “Online Participation” funded by the NRW Ministry for Innovation, Science, and Research.

References

[1] S. Airiau, E. Bonzon, U. Endriss, N. Maudet, and J. Rossit. Rationalisation of profiles of abstract argumentation frameworks: Characterisation and complexity. *Journal of Artificial Intelligence Research*, 60:149–177, 2017.

[2] L. Amgoud and S. Vesic. A new approach for preference-based argumentation frameworks. *Annals of Mathematics and Artificial Intelligence*, 63(2):149–183, 2011.

[3] K. Arrow. *Social Choice and Individual Values.* John Wiley and Sons, 1951 (revised edition 1963).

[4] Katie Atkinson and Trevor Bench-Capon. Value-based argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 5. College Publications, 2021.

[5] E. Awad, R. Booth, F. Tohmé, and I. Rahwan. Judgement aggregation in multi-agent argumentation. *Journal of Logic and Computation*, 27(1):227–259, 2017.

[6] R. Baumann and G. Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In *Proceedings of the 3rd International Conference on Computational Models of Argument*, volume 216 of *FAIA*, pages 75–86. IOS Press, September 2010.

[7] D. Baumeister, S. Bouveret, J. Lang, N. Nguyen, T. Nguyen, J. Rothe, and A. Saffidine. Positional scoring-based allocation of indivisible goods. *Journal of Autonomous Agents and Multi-Agent Systems*, 31(3):628–655, 2017.

[8] D. Baumeister, G. Erdélyi, O. Erdélyi, and J. Rothe. Complexity of manipulation and bribery in judgment aggregation for uniform premise-based quota rules. *Mathematical Social Sciences*, 76:19–30, 2015.

[9] D. Baumeister, G. Erdélyi, O. Erdélyi, J. Rothe, and A. Selker. Complexity of control in judgment aggregation for uniform premise-based quota rules. *Journal of Computer and System Sciences*, 112:13–33, 2020.

[10] D. Baumeister, P. Faliszewski, J. Lang, and J. Rothe. Campaigns for lazy voters: Truncated ballots. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, pages 577–584. IFAAMAS, June 2012.

[11] D. Baumeister, D. Neugebauer, and J. Rothe. Verification in attack-incomplete argumentation frameworks. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory*, pages 341–358. Springer-Verlag *Lecture Notes in Artificial Intelligence #9346*, September 2015.

[12] D. Baumeister, D. Neugebauer, and J. Rothe. Credulous and skeptical acceptance in incomplete argumentation frameworks. In *Proc. COMMA*, volume 305 of *FAIA*, pages 181–192. IOS Press, 2018.

[13] D. Baumeister, D. Neugebauer, J. Rothe, and H. Schadrack. Verification in incomplete argumentation frameworks. *Artif. Intell.*, 264:1–26, 2018.

[14] D. Baumeister, M. Roos, and J. Rothe. Computational complexity of two variants of the possible winner problem. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems*, pages 853–860. IFAAMAS, May 2011.

[15] D. Baumeister, M. Roos, J. Rothe, L. Schend, and L. Xia. The possible winner problem with uncertain weights. In *Proceedings of the 20th European Conference on Artificial Intelligence*, pages 133–138. IOS Press, August 2012.

[16] D. Baumeister and J. Rothe. Preference aggregation by voting. In J. Rothe, editor, *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, Springer Texts in Business and Economics,

chapter 4, pages 197–325. Springer-Verlag, 2015.

[17] D. Baumeister, J. Rothe, and H. Schadrack. Verification in argument-incomplete argumentation frameworks. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory*, pages 359–376. Springer-Verlag *Lecture Notes in Artificial Intelligence #9346*, September 2015.

[18] T. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[19] D. Black. *The Theory of Committees and Elections*. Cambridge University Press, 1958.

[20] G. Bodanza, F. Tohmé, and M. Auday. Collective argumentation: A survey of aggregation issues around argumentation frameworks. *Argument & Computation*, 8(1):1–34, 2017.

[21] S. Bouveret, U. Endriss, and J. Lang. Fair division under ordinal preferences: Computing envy-free allocations of indivisible goods. In *Proceedings of the 19th European Conference on Artificial Intelligence*, pages 387–392. IOS Press, August 2010.

[22] F. Brandt. Some remarks on Dodgson's voting rule. *Mathematical Logic Quarterly*, 55(4):460–463, 2009.

[23] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[24] M. Caminada and G. Pigozzi. On judgment aggregation in abstract argumentation. *Journal of Autonomous Agents and Multi-Agent Systems*, 22(1):64–102, 2011.

[25] I. Caragiannis, A. Procaccia, and N. Shah. When do noisy votes reveal the truth? *ACM Transactions on Economics and Computation*, 4(3):Article 15, 15:1–15:30, 2016.

[26] C. Cayrol, F. de Saint-Cyr, and M. Lagasquie-Schiex. Revision of an argumentation system. In *Proceedings of the 11th International Conference on Principles of Knowledge Representation and Reasoning*, pages 124–134. AAAI Press, September 2008.

[27] C. Cayrol, F. de Saint-Cyr, and M. Lagasquie-Schiex. Change in abstract argumentation frameworks: Adding an argument. *J. Artif. Intell. Res.*, 38:49–84, 2010.

[28] W. Chen and U. Endriss. Preservation of semantic properties in collective argumentation: The case of aggregating abstract argumentation frameworks. *Artificial Intelligence*, 269:27–48, 2019.

[29] Y. Chevaleyre, J. Lang, N. Maudet, J. Monnot, and L. Xia. New candidates welcome! Possible winners with respect to the addition of new candidates. *Mathematical Social Sciences*, 64(1):74–88, 2012.

[30] J.-A.-N. de Caritat, Marquis de Condorcet. Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix, 1785. Facsimile reprint of original published in Paris, 1972, by the Imprimerie Royale. English translation appears in I. McLean and A. Urken, *Classics of Social Choice*, University of Michigan Press, 1995, pages 91–112.

[31] S. Coste-Marquis, C. Devred, S. Konieczny, M. Lagasquie-Schiex, and P. Marquis. On the merging of Dung's argumentation systems. *Artif. Intell.*, 171(10):730–753, 2007.

[32] S. Coste-Marquis, C. Devred, and P. Marquis. Symmetric argumentation frameworks.

In *Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 317–328. Springer-Verlag *Lecture Notes in Artificial Intelligence #3571*, July 2005.

[33] S. Coste-Marquis, S. Konieczny, J. Mailly, and P. Marquis. Extension enforcement in abstract argumentation as an optimization problem. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 2876–2882. AAAI Press/IJCAI, July 2015.

[34] R. de Haan. Parameterized complexity results for the Kemeny rule in judgment aggregation. In *Proceedings of the 22nd European Conference on Artificial Intelligence*, pages 1502–1510. IOS Press, August/September 2016.

[35] R. de Haan. Complexity results for manipulation, bribery and control of the Kemeny judgment aggregation procedure. In *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems*, pages 1151–1159. IFAAMAS, May 2017.

[36] J. Delobelle, A. Haret, S. Konieczny, J. Mailly, J. Rossit, and S. Woltran. Merging of abstract argumentation frameworks. In *Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning*, pages 33–42. AAAI Press, April 2016.

[37] J. Delobelle, S. Konieczny, and S. Vesic. On the aggregation of argumentation frameworks: operators and postulates. *Journal of Logic and Computation*, 28(7):1671–1699, 2018.

[38] F. Dietrich and C. List. Judgment aggregation by quota rules: Majority voting generalized. *Journal of Theoretical Politics*, 19(4):391–424, 2007.

[39] F. Dietrich and P. Mongin. The premise-based approach to judgment aggregation. *Journal of Economic Theory*, 145:562–582, 2010.

[40] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoret. Comput. Sci.*, 170(1):209–244, 1996.

[41] C. Dodgson. A Method of Taking Votes on more than two Issues. Pamphlet printed by the Clarendon Press, Oxford, and headed "not yet published" (see the discussions in [65; 19], both of which reprint this paper), 1876.

[42] P. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[43] P. Dunne and T. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(1):187–203, 2002.

[44] P. Dunne, W. Dvořák, T. Linsbichler, and S. Woltran. Characteristics of multiple viewpoints in abstract argumentation. *Artificial Intelligence*, 228:153–178, 2015.

[45] P. Dunne, A. Hunter, P. McBurney, S. Parsons, and M. Wooldridge. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457–486, 2011.

[46] P. Dunne, P. Marquis, and M. Wooldridge. Argument aggregation: Basic axioms and

complexity results. *Proceedings of the 4th International Conference on Computational Models of Argument*, 245:129–140, 2012.

[47] U. Endriss and U. Grandi. Graph aggregation. *Artificial Intelligence*, 245:86–114, 2017.

[48] B. Fazzinga, S. Flesca, and F. Furfaro. Revisiting the notion of extension over incomplete abstract argumentation frameworks. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, pages 1712–1718. ijcai.org, July 2020.

[49] P. Fishburn. Condorcet Social Choice Functions. *SIAM Journal on Applied Mathematics*, 33(3):469–489, 1977.

[50] P. Fishburn. Monotonicity paradoxes in the theory of elections. *Discrete Applied Mathematics*, 4(2):119–134, 1982.

[51] D. Gabbay and O. Rodrigues. A numerical approach to the merging of argumentation networks. In *International Workshop on Computational Logic in Multi-Agent Systems*, pages 195–212. Springer, 2012.

[52] M. Garey and D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, 1979.

[53] A. Gibbard. Manipulation of Voting Schemes: A General Result. *Econometrica*, 41(4):587–601, 1973.

[54] E. Hemaspaandra, H. Spakowski, and J. Vogel. The complexity of Kemeny elections. *Theoretical Computer Science*, 349(3):382–391, 2005.

[55] A. Hunter and K. Noor. Aggregation of perspectives using the constellations approach to probabilistic argumentation. In *Proceedings of the 34nd AAAI Conference on Artificial Intelligence*, pages 2846–2853. AAAI Press, 2020.

[56] J. Kemeny. Mathematics Without Numbers. *Dædalus*, 88:577–591, 1959.

[57] A. Kerkmann, J. Lang, A. Rey, J. Rothe, H. Schadrack, and L. Schend. Hedonic games with ordinal preferences and thresholds. *Journal of Artificial Intelligence Research*, 67:705–756, 2020.

[58] K. Konczak and J. Lang. Voting procedures with incomplete preferences. In *Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling*, pages 124–129, July/August 2005.

[59] L. Kornhauser and L. Sager. Unpacking the court. *Yale Law Journal*, 96(1):82–117, 1986.

[60] L. Kornhauser and L. Sager. The one and the many: Adjudication in collegial courts. *California Law Review*, 349(3):382–391, 1993.

[61] B. Kuckuck and J. Rothe. Duplication monotonicity in the allocation of indivisible goods. *AI Communications*, 32(4):253–270, 2019.

[62] H. Li, N. Oren, and T. Norman. Probabilistic argumentation frameworks. In *Proceedings of the International Workshop on Theorie and Applications of Formal Argumentation*, volume 7132 of *LNAI*, pages 1–16. Springer, July 2011.

[63] G. Lisowski, S. Doutre, and U. Grandi. Aggregation in value-based argumentation frameworks. *arXiv preprint arXiv:1907.09113*, 2019.

[64] C. List and P. Pettit. Aggregating Sets of Judgments: An Impossibility Result. *Eco-*

nomics and Philosophy, 18(1):89–110, 2002.

[65] I. McLean and A. Urken. *Classics of Social Choice.* University of Michigan Press, 1995.

[66] V. Menon and K. Larson. Computational aspects of strategic behaviour in elections with top-truncated ballots. *Journal of Autonomous Agents and Multi-Agent Systems*, 31(6):1506–1547, 2017.

[67] S. Modgil and H. Prakken. A general account of argumentation with preferences. *Artificial Intelligence*, 195:361–397, 2013.

[68] P. Mongin. The doctrinal paradox, the discursive dilemma, and logical aggregation theory. *Theory and Decision*, 73(3):315–355, 2012.

[69] A. Niskanen, D. Neugebauer, M. Järvisalo, and J. Rothe. Deciding acceptance in incomplete argumentation frameworks. In *Proceedings of the 34nd AAAI Conference on Artificial Intelligence*, pages 2942–2949. AAAI Press, 2020.

[70] A. Niskanen, J. Wallner, and M. Järvisalo. Synthesizing argumentation frameworks from examples. *Journal of Artificial Intelligence Research*, 66:503–554, 2019.

[71] P. Pettit. Deliberative democracy and the discursive dilemma. *Philosophical Issues*, 11(1):268–299, 2001.

[72] A. Procaccia, J. Rosenschein, and G. Kaminka. On the robustness of preference aggregation in noisy environments. In *Proceedings of the 6th International Conference on Autonomous Agents and Multiagent Systems*, pages 416–422. IFAAMAS, May 2007.

[73] L. Rey. Complexity of abstract argumentation. Master's thesis, Heinrich-Heine-Universität Düsseldorf, Institut für Informatik, Düsseldorf, Germany, August 2014.

[74] J. Rothe, editor. *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division.* Springer Texts in Business and Economics. Springer-Verlag, 2015.

[75] J. Rothe, H. Schadrack, and L. Schend. Borda-induced hedonic games with friends, enemies, and neutral players. *Mathematical Social Sciences*, 96:21–36, 2018.

[76] M. Satterthwaite. Strategy-Proofness and Arrow's Conditions: Existence and Correspondence Theorems for Voting Procedures and Social Welfare Functions. *Journal of Economic Theory*, 10(2):187–217, 1975.

[77] K. Skiba, D. Neugebauer, and J. Rothe. Complexity of possible and necessary existence problems in abstract argumentation. In *Proceedings of the 24th European Conference on Artificial Intelligence*, pages 897–904. IOS Press, August/September 2020.

[78] Z. Terzopoulou and U. Endriss. Neutrality and relative acceptability in judgment aggregation. *Social Choice and Welfare*, 2020.

[79] F. Tohmé, G. Bodanza, and G. Simari. Aggregation of attack relations: A social-choice theoretical analysis of defeasibility criteria. In *Proceedings of the 5th International Symposium on Foundations of Information and Knowledge Systems*, pages 8–23. Springer-Verlag *Lecture Notes in Artificial Intelligence #4932*, February 2008.

[80] J. Wallner. Structural constraints for dynamic operators in abstract argumentation. *Argument & Computation*, 11:151–190, 2020.

[81] J. Wallner, A. Niskanen, and M. Järvisalo. Complexity results and algorithms for

extension enforcement in abstract argumentation. *Journal of Artificial Intelligence Research*, 60:1–40, 2017.

[82] L. Xia. Computing the margin of victory for various voting rules. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, pages 982–999. ACM Press, June 2012.

[83] L. Xia and V. Conitzer. Determining possible and necessary winners given partial orders. *Journal of Artificial Intelligence Research*, 41:25–67, 2011.

Received May 2021

Value-based Argumentation

Katie Atkinson, Trevor Bench-Capon
Department of Computer Science, University of Liverpool, UK
{katie,tbc}@liverpool.ac.uk

Abstract

Value-based argumentation is concerned with recognising, accounting for, and reasoning with, the social purposes promoted by agents' beliefs and actions. Value-based argumentation frameworks extend Dung's abstract argumentation frameworks by ascribing an additional property to arguments, representing the values they promote, and recognising audiences. Values are ordered according to the preferences of an audience (different audiences will have different preferences) and an attack is successful only if the value of the attacked argument is not preferred to its attacker by its audience. Arguments can be related to values through the use of an argumentation scheme, thus enabling us to structure value-based argumentation. We describe the motivation of value-based argumentation, its formal description and properties, the argumentation scheme and its associated critical questions and some of the applications to which value-based argumentation has been put.

1 Philosophical motivations for value-based argumentation

The formal models of value-based argumentation that are presented in this chapter are intended to capture various philosophical concepts that are reflected in everyday human reasoning. In this section we explain the key philosophical accounts that motivated the development of the computational models of value-based argument.

1.1 Values and audiences

The inspiration for value-based argumentation originally came from the book *New Rhetoric* of Perelman [70]. The key insight of the *New Rhetoric* was that the acceptability of an argument depended not only on the argument itself, or on available counterarguments, but on the audience to which it was addressed. For an argument to be accepted, *its audience has to accept it.* In subsequent work on this topic Perelman says:

> If men [sic] oppose each other concerning a decision to be taken, it is not because they commit some error of logic or calculation. They discuss apropos the applicable rule, the ends to be considered, the meaning to be given to values, the interpretation and characterisation of facts. [69], p150.

Perelman's academic roots were in jurisprudence and he drew on legal disputes to support his argument:

> Each [party] refers in its argumentation to different values [...] the judge will allow himself to be guided in his reasoning by the spirit of the system: i.e. by the values which the legislative authority seeks to protect and advance. [69], p152.

Consideration of this had also been noted in AI and Law. In their highly influential paper, [43] discussed what should happen in factor-based reasoning with cases [29] when there were no precedents to allow the case to be decided. They argued that in such cases the decision should be made according to which social purposes would be promoted by deciding for the plaintiff and which would be promoted by deciding for the defendant, and the decision made according to which would better serve the prevalent social values. Note that this means that different arguments can be accepted in different jurisdictions (attitudes to the death penalty in Georgia and Minnesota were very different in the 1970s), and at different times ("*stare decisis* would bow to changing values"[1]).

Thus there seems something missing from a purely logical view: sometimes the logic will fail to compel, and we will need to make a choice on other grounds. Since the situation occurs in important arenas like law, we do not want the choice to be arbitrary: we want to provide rational grounds for such choice. As Perelman puts it:

> Logic underwent a brilliant development during the last century when, abandoning the old formulas, it set out to analyze the methods of proof used effectively by mathematicians.. . . One result of this development is to limit its domain, since everything ignored by mathematicians is foreign to it. Logicians owe it to themselves to complete the theory of demonstration obtained in this way by a theory of argumentation. [70], p10.

[1] Justice Marshall in Furman v Georgia 408 U.S. 238 (1972).

The situation is reflected in Dung's abstract argumentation. Sometimes, the acceptability of an argument will not be unequivocally determined by the framework. Given a dilemma (cycles with even length in standard argumentation frameworks [28]) the restrictive grounded semantics will allow neither horn to be embraced, whereas the more permissive preferred semantics will allow either proposition to be believed, but offer no reason to opt for one rather than the other. Value-based argumentation attempts to offer reasons for this choice as part of a "theory of argumentation".

1.2 Direction of fit

The other key influence on value-based argumentation was the work of John Searle on practical reasoning and his notion of *direction of fit* [73]. Searle wrote

> Assume universally valid and accepted standards of rationality, assume perfectly rational agents operating with perfect information, and you will find that rational disagreement will still occur; because, for example, the rational agents are likely to have different and inconsistent values and interests, each of which may be rationally acceptable. [73], p. xv.

Searle's idea was that such rational disagreement was possible because of direction of fit. There is only a single actual world, and a single history of that world, and so our beliefs about the present and the past have to match that actual world. Because there is only one actual world, there is a right answer to questions of fact, and while there may be disagreement, this is something that should be capable of resolution, given complete information. Values, interests and aspirations can play no part in such *theoretical* reasoning: that would be to indulge in wishful thinking.

The future is, however, a different matter. There are many possible futures, and we can, through our actions, play a part in determining which will come to pass. In *practical* reasoning, reasoning about what we should do, we attempt to fit the world to our desires, so that our actions will bring about the future that we prefer. But here different values, interests, aspirations and even tastes, may be a legitimate source of rational disagreement. Some may find it strange if someone prefers vanilla ice cream to chocolate, but it is not irrational. Of course, these aspirations can affect deeper matters: in politics a desire for tax rises may exhibit a preference for equality over economic growth. Such a preference is not a matter of rationality, but of the values that one wishes to be expressed in a society.

Thus in practical reasoning, rational disagreement is to be expected [42]. The notion of direction of fit, however, applies not only to actions, but to the law. Disagreement is at the heart of law, and even at the highest level judges differ as to the proper outcome of a case. Five-to-four decisions occur in almost a fifth of cases heard by the US Supreme Court[2]. Not only is disagreement common, it is expected: that is why appeal courts typically comprise an odd number of judges, and why the more important the court the more the judges, so that the US Supreme Court has nine[3]. Nor can judicial disagreement be considered irrational: after all, the minority will produce an opinion stating their reasons for their views. To a certain extent the judges are trying to fit their view of the current cases to the existing law: the doctrine of *stare decisis* means that their decision should be consistent with past decisions. For a logical analysis of precedential constraint, see [56]. However, it is often the case that the precedents do not fully constrain the decision: it may be that all of them can be distinguished according to some features of the current case. For such cases the judges are free to decide for either party. Here they try to fit the law as it will be after their decision (for the current case will serve as a precedent for future cases) to the way they desire the law should be. That is, they consider which decision will promote the purposes of the law better, as described in [43]. Therefore, as in practical reasoning, the values and aspirations of the judges will determine their decision [10]. The justification is that the majority opinion of a properly appointed court should reflect the prevailing values of its society.

1.3 Value-based argumentation

To reflect the situation where the dispute is about how best to fit the world to our desires it is clear that a basic assumption of Dung's argumentation frameworks, that attacks always succeed in defeating the argument they attack, must be relaxed. As an example, while it is true that Sarah will not be able to go on holiday if she buys a new car, this attack can simply be ignored if Sarah prefers the holiday: she can continue to make do with her current car. For a different person, however, perhaps a petrol-head like Jeremy, the attack will be decisive and the holiday plans abandoned.

Thus to reflect debates where values, aspirations and tastes matter, not only in everyday practical reasoning, but in important areas such law, politics and ethics as well, a method of augmenting Dung's framework with a notion of values was needed. *Values* was the term used to cover these subjective preferences. It is a term widely understood in this sense in popular media, and the notion of a *value premise* is a key part of the Lincoln-Davis debate format used throughout the USA as the basis

[2] Between 2000 and 2018, according to the US Supreme Court as reported in *The Washington Post* https://www.washingtonpost.com/news/posteverything/wp/2018/06/28/those-5-4-decisions-on-the-supreme-court-9-0-is-far-more-common/. Last accessed February 2020.

[3] Nine is the traditional number. As we write, in the run up to the 2020 Presidential election, there is speculation that this may be increased.

for competitive debating in a number of leagues[4]. Thus the general notion of values is felt to be widely understood. For example, the French Republic was based on the three values of liberty, equality and fraternity. In value-based reasoning there have been many different sets of values used for different problems. Generally it is held that the identification of the relevant set of values is part of the formulation of the problem to be discussed [13]. Some attempts have been made to provide a basis for the identification of values: e.g. [75] used Schwartz Value Theory [72] and [30] used Maslow's hierarchy of needs [61]. Often, however, it seems that a very general account is not best suited to a particular problem, and the use of problem specific value sets remains common.

2 Values in abstract argumentation frameworks

Value-based argumentation first appeared in the context of an extension to Dung's abstract argumentation framework, first in [24] and later in the journal version [25]. The basic idea was to extend Dung's notion of an augmentation framework as pair of a set of arguments and a set of the attacks between them, $\langle Ar, att\rangle$ by adding a set of values, V, a function mapping the members of Ar onto V, val, and a set of audiences P, expressed as orderings on V. Note that P might contain all the factorially many possible orderings on V, or only a selection of them. This might be to represent a particular set of agents with specific preferences, or some constraint on the ordering itself. For example, in order to represent facts, theoretical arguments are typically related to the value *truth*. Then to avoid wishful thinking, truth must be the most preferred value for every audience.

2.1 Extending Dung's argumentation frameworks with values

Accordingly, a value-based argumentation framework (VAF) is defined as an extension of a Dung-style argumentation framework (AF).

Definition 2.1 (Value-Based Argumentation Framework (VAF)). *A value-based argumentation framework is a 5-tuple $VAF = \langle Ar, att, V, val, P\rangle$ where Ar is a finite set of arguments, att is an irreflexive binary relation on Ar, V is a nonempty set of values, val is a function which maps from elements of Ar to elements of V and*

[4] Including the National Speech and Debate Association, or NSDA (formerly known as the National Forensics League, or NFL) competitions, and related debate leagues such as the National Christian Forensics and Communication Association, the National Catholic Forensic League, the National Educational Debate Association, the Texas University Interscholastic League, Texas Forensic Association, Stoa USA and their affiliated regional organizations.

P is the set of possible audiences (represented as orderings on V). We say that an argument $a \in Ar$ relates to a value $v \in V$ if accepting a promotes or defends v: The value in question is given by $val(ar)$. For every $ar \in Ar$, $val(ar) \in V$.

Note that if there is a single value, (perhaps *truth*), a VAF is equivalent to a standard Dung AF. If every argument maps to its own distinct value, we have a similar situation to the Preference Based Frameworks of Amgoud and Cayrol [3] and [4], except that Preference Based Argumentation uses only a single ordering so that P has only one member, and there is only a single audience.

In order to evaluate the status of arguments with respect to an audience we produce an audience specific value-based argumentation framework.

Definition 2.2 (Audience-Specific VAF (AVAF))**.** *An audience-specific VAF is a 5-tuple $AVAF = \langle Ar, att, V, val, Valpref_a \rangle$, where Ar, att, V and val are as for a VAF, a is an audience, $a \in P$, and $Valpref_a$ is a preference relation (transitive, irreflexive and asymmetric), $Valpref_a \subseteq V \times V$, reflecting the value preferences of audience a. The AVAF relates to the VAF from which it is derived in that Ar, att, V and val are identical, and $Valpref_a$ is the set of preferences derivable from the ordering $a \in P$ in the VAF.*

Our purpose in introducing VAFs is to allow us to distinguish between one argument attacking another, and that attack succeeding, so that the attacked argument may or may not be defeated. Whether the attack succeeds depends on the value order of the audience considering the VAF. We therefore define the notion of defeat for an audience:

Definition 2.3 (Defeat for an Audience)**.** *An argument ar $defeats_a$ an argument br for audience a, ($defeats_a(ar, br)$), in an AVAF $\langle Ar, att, V, val, Valpref_a \rangle$ if and only if both $attacks(ar, br) \in att$ and not $valpref(br, ar) \in Valpref_a$.*

We can now define audience specific versions of the notions standardly associated with AFs:

Definition 2.4 (Acceptable to an Audience)**.** *An argument $ar \in Ar$ is acceptable to an audience a with respect to set of arguments S, ($acceptable_a(ar, S)$) if: $\forall(x)(x \in Ar \wedge defeats_a(x, ar) \rightarrow \exists(y)(y \in S \wedge defeats_a(y, x))$*

Definition 2.5 (Conflict Free for an Audience)**.** *A set S of arguments is conflict free for an audience a if:*

$\forall(x)\forall(y)(x \in S \wedge y \in S) \rightarrow (\neg(attacks(x, y) \in att) \vee (valpref(val(y), val(x)) \in Valpref_a))$

Definition 2.6 (Admissability for an Audience)**.** *A conflict free for an audience a set of arguments S is admissible for the audience a if:* $\forall(x)(x \in S \rightarrow acceptable_a(x, S)$

Definition 2.7 (Preferred Extension for an Audience)**.** *A set of arguments S in a value-based argumentation framework* $\langle Ar, att, V, val, Valpref_a\rangle$ *is a preferred extension for audience-a, (preferred*$_a$*), if it is a maximal (with respect to set inclusion) admissible for audience a subset of Ar.*

A practical way of evaluating the status of arguments in an AVAF is to remove from the VAF all the unsuccessful attacks, those for which $valpref(br, ar)$ $\in Valpref_a$, whereupon it can be treated as a standard AF. Thus for any AVAF, $vaf_a = \langle Ar, att_{avaf}, V, val, valpref_a\rangle$ there is a corresponding AF, $af_a = \langle A, att_{af}\rangle$ such that for $(x, y) \in att_{avaf}$, $(x, y), \in att_{af}$ if and only if $defeats_a(x, y)$. The preferred extension of af_a will contain the same arguments as the preferred extension for audience a of the VAF. Note that if the original VAF does not contain any cycles in which all arguments pertain to the same value, af_a will contain no cycles, since every cycle will be broken at the point at which the attack is from an inferior value to a superior one for audience a. Hence both af_a and vaf_a will have a unique, non-empty, preferred extension for such cases.

Theorem 2.8. ***Every AVAF with no single-valued cycles has a unique nonempty preferred extension.***

PROOF. Let avaf be an AVAF, and let af be the standard argumentation framework resulting from removing all failing attacks. If avaf is cycle-free, then af is cycle free and hence by Theorem 2.6 of [25] it has a unique, not-empty preferred extension. But suppose avaf has a cycle. We know that this contains at least two values. Let v be the least preferred value in the cycle, and arg be the final argument in a chain relating to this value. The attack from arg to the next argument in the cycle will fail. Therefore this attack will not appear in af and the cycle will be broken at this point. This applies to all cycles in avaf. Therefore af is cycle free, and so has a unique, non-empty, preferred extension. QED

Moreover, since the AF derived from an AVAF contains no cycles, the grounded extension coincides with the preferred extension for this audience, and so there is a straightforward polynomial-time algorithm to compute it, given in [25].

For the moment we will restrict consideration to VAFs which do not contain any cycles in a single value. For such VAFs, the notions of sceptical and credulous acceptance are of no relevance, since any given audience will accept only a single preferred extension. These preferred extensions may, and typically will, however, differ from audience to audience. We therefore introduce two useful notions: *objective acceptance*, arguments which are acceptable to all audiences irrespective of their

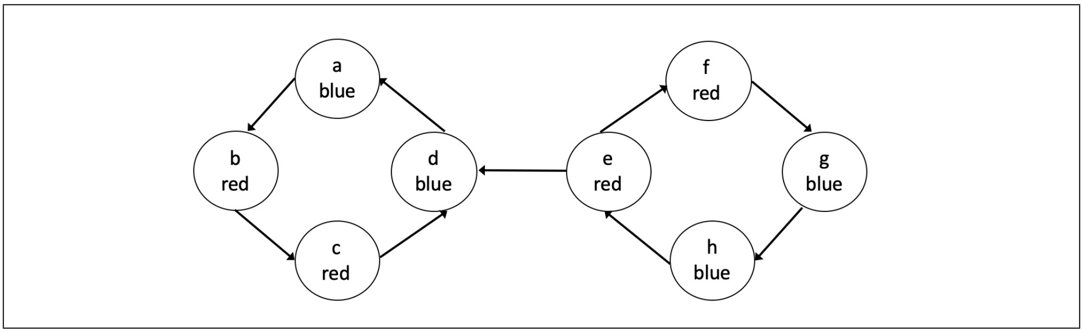


Figure 1: VAF with values red and blue

particular value order, and s*ubjective acceptance*, arguments which can be accepted by audiences with the appropriate value order.

Definition 2.9 (Objective Acceptance.)**.** *Given a VAF $\langle Ar, att, V, val, Valpref \rangle$, an argument $a \in A$ is objectively acceptable if and only if for all $valpref \in Valpref$, a is in every $valpref$.*

Definition 2.10 (Subjective Acceptance.)**.** *Given a VAF $\langle Ar, att, V, val, Valpref \rangle$, an argument $a \in A$ is subjectively acceptable if and only if for some $valpref \in Valpref$, a is in that $valpref$.*

An argument which is neither objectively nor subjectively acceptable (such as one attacked by an objectively acceptable argument with the same value) is said to be *indefensible.*

All arguments which are not attacked will, of course, be objectively acceptable. Otherwise, objective acceptance typically arises from cycles in two or more values. For example, consider a three-cycle in two values, say two arguments with V_1 and one with V_2. The argument with V_2 will either resist the attack on it when it is preferred to V_1, or, when V_1 is preferred, fail to defeat the argument it attacks which will, in consequence, be available to defeat its attacker. Thus the argument with V_2 will be objectively acceptable, and both the arguments with V_1 will be subjectively acceptable. A more elaborate example is shown in Figure 1.

There will be two preferred extensions, according to whether $red > blue$, or $blue > red$. If $red > blue$, the preferred extension will be $\{e, g, a, b\}$, and if $blue > red$, $\{e, g, d, b\}$. Now e, g and b are objectively acceptable, but d, which would have been objectively acceptable if e had not attacked d, is only subjectively acceptable (when $blue > red$), and a, which is indefensible if d is not attacked, is also subjectively acceptable (when $red > blue$). Arguments c, f and h are indefensible. Results characterising the structures which give rise to objective acceptance are given in [6].

The question now arises as to whether it is possible to determine to which audiences an argument is acceptable. This question is fully explored in [35].

2.2 Computational complexity results of value-based argumentation frameworks

Not long after VAFs were first proposed in the literature, a study was conducted on a number of decision problems in VAFs [50]. In that paper it was shown that, for a given audience, those decision questions which are typically computationally hard in the standard Dungian AF setting, actually admit efficient solution methods in the value-based setting. The paper also highlighted a number of questions that arise solely in value-based frameworks that lack efficient decision processes.

The two key questions addressed in the paper concern the decision problems in VAFs of subjective and objective acceptance, as set out in Definitions 2.9 and 2.10 above. Concerning the decision problem of subjective acceptance, it is shown in [50] that the complexity of this problem is NP-complete, and for objective acceptance, the decision problem is shown to be CoNP-complete. The paper also considers decision problems related to determining subjective acceptance by attempting to identify which pair-wise orderings are "critical" in that a given ordering will admit an audience for which an argument is subjectively accepted, whereas reversing this order will yield a situation in which the argument of interest is never accepted. Full results and their proofs are given in [50]. Extrapolating from the results, they demonstrate that the identification of an argument as subjectively or objectively acceptable is just as hard as the corresponding problems of determining credulous and sceptical acceptance in standard argumentation frameworks; see [49] for a full discussion of this point. Further complexity results, especially those concerning which audience can accept a given argument, can be found in [35].

Further studies on computational complexity problems were later reported in [48]. By considering properties of the directed graph structure defined by taking those values involved in conflicting arguments, Dunne identified an extensive class of argumentation systems for which the subjective and objective decision problems admit polynomial time solutions.

More recently, [66] examined specific questions in abstract argumentation frameworks under preferred semantics. They looked at the acceptance problem in standard argumentation frameworks, deciding whether a specific argument is in at least one preferred extension (i.e. it is credulously accepted) or in all such extensions (i.e. it is skeptically accepted). The paper presents an algorithm that enumerates all preferred extensions and builds algorithms that decide the acceptance problem without requiring explicit enumeration of all extensions. The improvements in efficiency brought about by the algorithms are achieved through a number of mechanisms: introduction of new labels for arguments' status, introduction of a new mechanism for pruning the search space so that transitions leading to dead ends are avoided at an early stage, and introduction of a cost-effective heuristic rule that yields earlier identification of arguments for transitions that might reach a goal state designating a preferred extension. The techniques developed for the acceptance problem in AFs are then used analogously to solve decision problems in VAFs, specifically deciding subjective and objective acceptance. Algorithms to solve these problems are defined and full proofs of the soundness and completeness of these algorithms is given in [66].

The studies referenced above set out properties of VAFs with a view to demonstrating their viability for use in domain applications. We now turn to considering how values are captured in accounts of structured argumentation.

3 Values in structured argumentation

In the previous section we showed how abstract value-based argumentation could be used to account for the subjective preferences which come into play when we are reasoning about how to make the world fit our desires. But the question arises: *how do values become attached to arguments?* The discussion in section 1 suggested that arguments for which value preferences are relevant are likely to arise in practical reasoning, reasoning about what to do. We will therefore begin our search for the link between arguments and values by looking at practical reasoning.

3.1 Practical syllogism

Practical reasoning was identified as different from theoretical reasoning by Aristotle in his *Nicomachean Ethics*, The discussion was revived by [6] and [57]. Kenny's example of a practical syllogism is

K1: I'm to be in London at 4.15.
If I catch the 2.30 train, I'll be in London at 4.15.
So, I'll catch the 2.30 train.

Although Aristotle attempted to present the practical syllogism as a deduction, this position proved difficult to maintain, and Kenny's abductive presentation is now more common. It still has, however, a number of peculiarities.

- The conclusion is not really a prediction. Whether or not I actually catch the train is contingent on a number of things beyond my control. Rather it is a resolution, a *decision* to try to catch the train. The result of practical reasoning should not be a belief, but an action or a plan of action which will realise the desires one has decided to pursue.
- The truth of the premises is not enough to determine the decision. There may be earlier trains, and I may decide to catch one of those to be on the safe side. There may be many other ways of achieving the goal. Like any abduction, its soundness depends on it being the best (for me, in my current circumstance) way to achieve the goal.
- If I do catch the train, there will be many things that I cannot do. If I in fact prefer to do one of these things to being in London, then I may choose one of these other activities.
- There may be a number of other consequences of catching the train which are not desirable. These may be sufficiently undesirable that I decide not to catch the train.

These aspects are somewhat reflected in Searle's formulation in [73]:

S1: I want, all things considered, to achieve E.
The best way, all things considered, to achieve E is to do M.
So, I will do M.

In order to act on the basis of an argument such as K1, therefore, we need to consider alternative actions, alternative goals and any additional consequences, and then choose the best of these alternative goals and actions. Note the element of choice here: we can choose which of our goals we will seek to realise, and which actions to undertake to realise these goals. In order to decide which is best, I need to go beyond the goals themselves, and consider why these states of affairs are wanted. This is where values come in. It is our values that make certain states of affairs goals, because these states of affairs promote our values. In [21] there was a detailed discussion of how values give rise to a number of types of goal such as maintenance goals, achievement goals, avoidance goals and removal goals.

It is the values associated with these goals that determines which of them should be considered best by a particular person. Which is best will be determined by the preference ordering on values, and so may vary from person to person. Whether I decide to catch the train in K1 depends on the value served by being in London, and the values served by possible alternatives.

In order to assist with the formulation of a computational version of practical reasoning, we decided to propose an argumentation scheme, in the manner of [76].

3.2 Argumentation schemes

Walton's notion of an argumentation scheme is that it is a means of presumptive reasoning: if the premises are true, then we may presumptively draw the conclusion, subject to satisfactorily dealing with critical questions characteristic of the scheme.

Walton [76] proposes two schemes relating to practical reasoning. The first is the *necessary condition scheme*

W1: G is a goal for agent *a*.
Doing action A is necessary for agent *a* to carry out goal G.
Therefore agent *a* ought to do action A.

The other was quite similar: the *sufficient condition scheme.*

W2: G is a goal for agent *a*.
Doing action A is sufficient for agent *a* to carry out goal G.
Therefore agent *a* ought to do action A.

Walton associates four critical questions with each of these schemes:

- WCQ1: Are there alternative ways of realising goal G?
- WCQ2: Is it possible to do action A?
- WCQ3: Does agent *a* have goals other than G which should be taken into account?
- WCQ4: Are there other consequences of doing action A which should be taken into account?

Although these arguments are fair reflections of the practical syllogisms K1 and S1, they have no link to values. As we saw above, values are essential for evaluation. Thus if critical question WCQ1 is posed, and it proves that there is an alternative action, say A2, without values we have no reason to say that this is a *better* alternative, and so choose to realise G with A2 rather than A.

For this reason we introduced an argumentation scheme which did have the required link to values. This scheme was first presented in [16] and was more fully reported in [17]. The scheme was stated in [7] as:

AS1: In the circumstances R
we should perform action A
to achieve new circumstances S
which will realise some goal G
which will promote some value V.

In [7] and [17] sixteen critical questions were identified:

- CQ1: Are the believed circumstances true?
- CQ2: Assuming the circumstances, does the action have the stated consequences?
- CQ3: Assuming the circumstances and that the action has the stated consequences, will the action bring about the desired goal?
- CQ4: Does the goal realise the value stated?
- CQ5: Are there alternative ways of realising the same consequences?
- CQ6: Are there alternative ways of realising the same goal?
- CQ7: Are there alternative ways of promoting the same value?
- CQ8: Does doing the action have a side effect which demotes the value?
- CQ9: Does doing the action have a side effect which demotes some other value?
- CQ10: Does doing the action promote some other value?
- CQ11: Does doing the action preclude some other action which would promote some other value?
- CQ12: Are the circumstances as described possible?
- CQ13: Is the action possible?
- CQ14: Are the consequences as described possible?
- CQ15: Can the desired goal be realised?
- CQ16: Is the value indeed a legitimate value?

In [13] a seventeenth CQ was added:

- CQ17: Can others act so as to take us to a state other than S?

This scheme allowed arguments for actions to be related to values: instantiating the scheme would give such an argument. Instantiating the critical questions would provide a means of attacking such arguments. This process of reasoning is illustrated in [32] and [13].

3.3 Semantics for structured value-based argumentation

In order to provide a semantic underpinning for this argument scheme and critical questions, use was made of the notion of Action Based Alternating Transition Systems (AATS) with values (AATS+V). These were introduced in [11] and more fully reported in [13].

An AATS is a type of state transition diagram, introduced in [79], formally based on Alternating-time Temporal Logic [2]. In an AATS the states and transitions can be used to represent the current and future situations and the actions which will lead between them. In fact these transitions represent *joint actions*[5], that is, the cumulative effect of every agent relevant to the situation performing one action each. This means that a given action of a particular agent will appear in several transitions, depending on what the other agents do, and an agent may consequently not be in full control of the state that will be reached by using that action.

The definition of an AATS is:

Definition 3.1 (AATS ([79])). .

An Action-based Alternating Transition System *(AATS) is an* $(n + 7)$*-tuple* S $= \langle$Q, q$_0$, Ag, Ac$_1$, ... , Ac$_n$, $\rho, \tau, \Phi, \pi\rangle$*, where:*

- Q *is a finite, non-empty set of* states*;*
- q$_0 \in$ Q *is the* initial state*;*
- Ag = *{1,...,*n*} is a finite, non-empty set of* agents*;*
- Ac$_i$ *is a finite, non-empty set of actions, for each* $ag_i \in$ Ag *where* Ac$_i \cap$ Ac$_j$ $= \emptyset$ *for all* $ag_i \neq ag_j \in$ Ag*;*
- ρ *:* Ac$_{ag} \rightarrow 2^{\mathrm{Q}}$ *is an* action pre-condition function*, which for each action* $\alpha \in$ Ac$_{ag}$ *defines the set of states* $\rho(\alpha)$ *from which* α *may be executed;*

[5] No suggestion of cooperation is intended here: the actions are joint solely in the sense that they are performed simultaneously.

- τ *:* Q $\times$ J_{Ag} $\rightarrow$ Q *is a partial* system transition function*, which defines the state* τ*(*q, j*) that would result by the performance of* j *from state* q*. This function is partial as not all joint actions are possible in all states;*

- Φ *is a finite, non-empty set of* atomic propositions*; and*

- π *:* Q $\rightarrow$ 2^{Φ} *is an interpretation function, which gives the set of primitive propositions satisfied in each state: if* p $\in$ π*(*q*), then this means that the propositional variable* p *is satisfied (equivalently, true) in state* q*.*

As presented in [79], AATS have no values. Therefore they were extended in [13] to include values, giving an AATS+V in which the transitions are additionally labelled with the values promoted or demoted by that transition. The additional definitions are:

Definition 3.2 (AATS+V ([13]))**.**
Given an AATS, an AATS+V is defined by adding two additional elements as follows:

- V *is a finite, non-empty set of values.*

- δ *:* Q $\times$ Q $\times$ V $\rightarrow$ *{+, –, =} is a* valuation function *which defines the status (promoted (+), demoted (–) or neutral (=)) of a value* v_u $\in$ V *ascribed to the transition between two states:* δ*(*q_x*,* q_y*,* v_u*) labels the transition between* q_x *and* q_y *with one of {+, –, =} with respect to the value* v_u $\in$ V*.*

With this definition it is possible to describe the practical reasoning argumentation scheme and critical questions in terms of the extended AATS+V. This gives us:

AS2 In the initial state $q_0 = q_x \in Q$,
Agent $i \in Ag$ should participate in joint action $j_n \in J_{Ag}$ where ${j_n}^i = \alpha_i$,
Such that $\tau(q_x, j_n)$ is q_y,
Such that $p_a \in \pi(q_y)$ and $p_a \notin \pi(q_x)$, or $p_a \notin \pi(q_y)$ and $p_a \in \pi(q_x)$,
Such that for some $v_u \in Av_i$, $\delta(q_x, q_y, v_u)$ is +.

We may now state the critical questions in these terms also.

- CQ1: $q_0 \neq q_x$ and $q_0 \notin \rho(\alpha_i)$.

- CQ2: $\tau(q_x, j_n)$ is not q_y.

- CQ3: $p_a \notin \pi(q_y)$.

- CQ4: $\delta(q_x,\ q_y,\ v_u)$ is not $+$.
- CQ5: Agent $i \in Ag$ can participate in joint action $j_m \in J_{Ag}$, where $j_n \neq j_m$, such that $\tau(q_x,\ j_m)$ is q_y.
- CQ6: Agent $i \in Ag$ can participate in joint action $j_m \in J_{Ag}$, where $j_n \neq j_m$, such that $\tau(q_x,\ j_m)$ is q_y, such that $p_a \in \pi(q_y)$ and $p_a \notin \pi(q_x)$ or $p_a \notin \pi(q_y)$ and $p_a \in \pi(q_x)$.
- CQ7: Agent $i \in Ag$ can participate in joint action $j_m \in J_{Ag}$, where $j_n \neq j_m$, such that $\tau(q_x,\ j_m)$ is q_z, such that $\delta(q_x,\ q_z,\ v_u)$ is $+$.
- CQ8: In the initial state $q_x \in Q$, if agent $i \in Ag$ participates in joint action $j_n \in J_{Ag}$, then $\tau(q_x,\ j_n)$ is q_y, such that $p_b \in \pi(q_y)$, where $p_a \neq p_b$, such that $\delta(q_x,\ q_y,\ v_u)$ is –.
- CQ9: In the initial state $q_x \in Q$, if agent $i \in Ag$ participates in joint action j_n $\in J_{Ag}$, then $\tau(q_x,\ j_n)$ is q_y, such that $\delta(q_x,\ q_y,\ v_w)$ is –, where $v_u \neq v_w$.
- CQ10: In the initial state $q_x \in Q$, if agent $i \in Ag$ participates in joint action $j_n \in J_{Ag}$, then $\tau(q_x,\ j_n)$ is q_y, such that $\delta(q_x,\ q_y,\ v_w)$ is $+$, where $v_u \neq v_w$.
- CQ11: In the initial state $q_x \in Q$, if agent $i \in Ag$ participates in joint action $j_n \in J_{Ag}$, then $\tau(q_x,\ j_n)$ is q_y and $\delta(q_x,\ q_y,\ v_u)$ is $+$. There is some other joint action $j_m \in J_{Ag}$, where $j_n \neq j_m$, such that $\tau(q_x,\ j_m)$ is q_z, such that $\delta(q_x,\ q_z,\ v_w)$ is $+$, where $v_u \neq v_w$.
- CQ12: $q_x \notin Q$.
- CQ13: $j_n \notin J_{Ag}$.
- CQ14: $\tau(q_x,\ j_n) \notin Q$.
- CQ15: $p_a \notin \pi(q)$ for any $q \in Q$.
- CQ16: $v_u \notin V$.
- CQ17: ${j_n}^i = {j_m}^i$, $j_n \neq j_m$ and $\tau(q_x,\ j_n) \neq \tau(q_x,\ j_m)$.

This formal account of the practical reasoning argumentation scheme and critical questions enable them to be used in agent systems designed to model practical reasoning scenarios; examples of these are provided in Section 4.

3.4 Dialogue interactions: values in persuasion and deliberation

In the previous sections we have considered reasoning with a specific audience which can determine the value order and evaluate the arguments accordingly. Often, however, values are crucial in dialogues where we have two or more audiences each with their own value order. Two distinct types of such dialogue are *persuasion* and *deliberation* [77].

In persuasion it is the person being persuaded who determines the value order [42], since people will accept an argument only if it is acceptable on their own value ordering. Thus the proponents may not be able to use the arguments which convinced them because these will be acceptable on their value order, but perhaps not on the value order of the person they wish to persuade. Thus in a persuasion dialogue it is often necessary to elicit the value order of the other person, so that arguments acceptable to them can be found. Sometimes, however, it will not be possible to find arguments acceptable to the other, in which case the persuader must first try to get them to accept a value ordering and then to accept the argument which is the topic of the dialogue. Such dialogues are modelled in [36]. A strategy for efficient persuasion in dialogues is given in [9].

Deliberation is different in that while the value orders may well differ, neither party can determine what it should be. Therefore one purpose of a deliberation dialogue is to find a value ordering which will be acceptable to all concerned, so that a solution corresponding to this order can be found, which should be acceptable to all the parties. A set of speech acts to support dialogues based on this view of deliberation is given in [20] and a tool showing how these speech acts can be used to generate persuasion and deliberation dialogues in agent systems is described in [58].

4 Key applications of value-based argumentation

In this section we will illustrate the use of value-based argumentation in a number of domains.

4.1 General practical reasoning

We will begin by looking at the use of value-based argumentation in general practical reasoning. Our example will be that used in [13], which adapts a well known brain teaser. AI students may be familiar with it as it is often used to illustrate search problems.

The situation is that a farmer is returning from market with a chicken (C), a bag of seeds (S) and his faithful dog (D). He needs to cross a river, and there is a boat (B) but it can only carry the farmer and one of his possessions. He cannot leave the chicken and seeds together because the chicken will eat the seeds. Similarly, he cannot leave the dog and the chicken unattended together because the dog will eat the chicken. His problem is how to organise his crossing.

We will represent the states by two lists, one for the items on the right bank, and one for items on the left. Thus [BCDS, _] will be selected from Q as the initial state, q_0. The complete set of states is shown in Figure 2

The transitions will be formed by various joint actions. We will assume that the animals will eat if they can, and so ignore the possibility of, for example, leaving the dog and chicken alone, and the dog doing nothing. This gives us the following six joint actions.

j_1: All do nothing

j_2: Farmer rows alone, chicken eats seeds if possible, dog eats chicken if possible

j_3: Farmer rows seeds, dog eats chicken if possible

j_4: Farmer rows dog, chicken eats seeds if possible

j_5: Farmer rows chicken, animals do nothing

j_6: All continue their journey home.

We can also identify a number of possible values[6]:

P: Progress - Promoted when farmer moves one of his possessions to the right side of the river. It is demoted when a state is revisited (through the always undesirable "goal" of repetition), and, to a lesser extent, when a possession is rowed from the right bank to the left (Pr). Rowing an item back is preferred to repetition, since repeating a state cannot be progress, whereas reaching a new state by returning an item to the left bank might be on a solution path, even though a *prima facie* backwards step.

S: Farmer has seeds - demoted when farmer loses seeds.

C: Farmer has chicken - demoted when farmer loses chicken.

[6] Some labels for values are the same as the propositions used in the state description. The context makes it clear which is intended.

F: Friendship - promoted when farmer travels with dog (it was for this companionship that he brought the dog with him).

We assume that the farmer values his possessions most, then wishes to make progress, and then have the joys of companionship. His value order is thus

$C, S > P > Pr > F$

We can now apply the joint actions to q_0 and label the transitions according to how they promote or demote the values. Initially five of the six actions are available, since the preconditions for j_6 are not satisfied. The result is shown as the first layer of Figure 2.

We can see that the only action which promotes a value without demoting a preferred value is j_5, and so the farmer will row the chicken, using the following argument:

- Farmer should row the chicken to promote Progress.

From q_2 three actions are possible. But two of them demote progress by reaching previous states, so the argument is

- Farmer should not row the seeds, or do nothing, as that would demote progress. So Farmer should row alone.

Having reached q_6, there are two actions which promote progress, rowing the seeds, and rowing the dog. But rowing the dog additionally promotes friendship, and so that will be chosen. From q_8 the only harmless action is to row the chicken to reach q_{10}. From q_{10} progress can be promoted by rowing the seeds, while all other actions demote a value. From here the only neutral action is to row alone to reach q_{13}. From here the farmer can promote progress by rowing the chicken. Now at last everything is on the right bank, and progress can be made by them all proceeding home.

This example shows how the puzzle can be solved by simply considering how to best promote values at every move. No look ahead is needed. In the standard puzzle, heuristic search gives two solutions, since rowing either the dog or the seeds in q_6 will achieve the goal. In the practical reasoning version this is resolved because in q_6 the farmer chooses to row the dog, to promote friendship as well as progress. For another example of practical reasoning, deciding whether to travel by aeroplane or train, see [31].

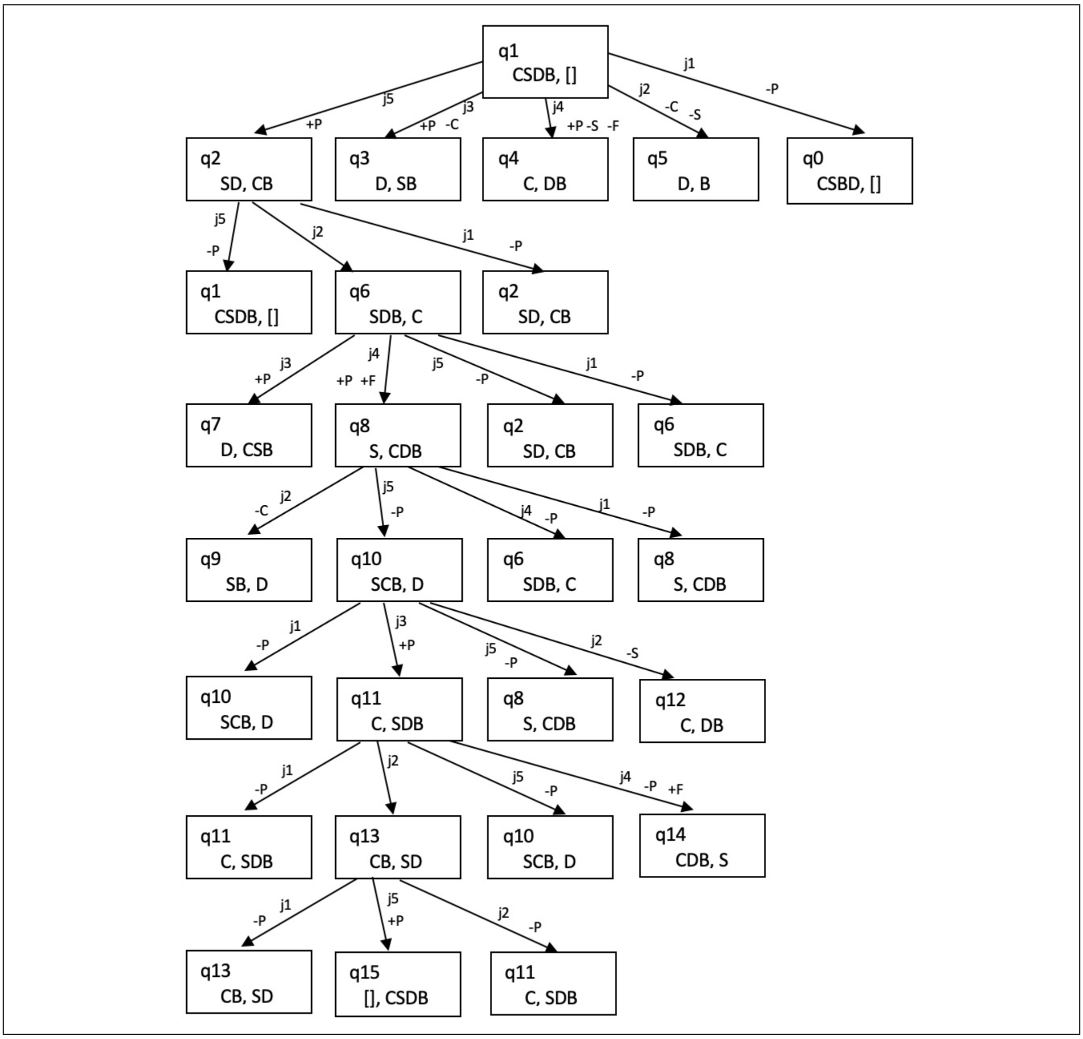


Figure 2: AATS+V for the farmer's river crossing problem. Note that when seeds and chicken are eaten, they no longer appear in the state descriptions.

4.2 Domain-specific application: law

A domain in which value-based argumentation has been widely used is law, and in that domain arguing with values precedes abstract value-based argumentation and the formal modelling of structured argument with values by over a decade.

The notion of values was in introduced to AI and Law by [43][7]. In that paper Berman and Hafner noted that when using factor-based reasoning [29], often there were factor-based arguments for both sides which needed to be chosen between. Factor-based reasoning as proposed in HYPO [71] and CATO [1], however, offered no rationale for choosing between them. The answer given in [43] was that the arguments which better served the purposes of the law should be accepted. This idea was elaborated into a more formal theory of reasoning with cases as theory construction, in which value preferences were derived from precedents which were then applied to new cases, in [40] and [41], which was was the basis of the CATE [46] and AGATHA [45] systems. In [54] it was proposed that the argumentation scheme for practical reasoning, described in [17] and discussed above, could be used to generate the case based arguments required by factor-based reasoning and link them to values. The wild animals cases of [43] had been modelled as a Dung style argumentation framework in [23]. These various strands were brought together in [32], which added values to the AF of [23], and evaluated the arguments according to the resulting VAF.

In [43] the example cases were some well known property law cases (often used as an introduction to property law in US law schools) concerning wild animals. That paper discussed three cases:

- *Keeble v Hickergill (1707).* This was an English case in which Keeble rented a duck pond, to which he lured ducks, which he shot and sold for consumption. Hickergill, out of malice, scared the ducks away by firing guns. The court found for Keeble. Two arguments for Keeble are possible: that he was engaged in an economically valuable activity, and that he was operating on his own land. The former reading is adopted in [43], but others, e.g. [39], prefer the latter.

- *Pierson v Post (1805).* In this New York case, Post was hunting a fox with hounds. Pierson intercepted the fox, killed it with a handy fence rail, and carried it off. The court found for Pierson. The argument was that Post had never had possession of the fox. The argument that hunting vermin is a useful activity which needs protection and encouragement formed the basis of the minority opinion. In this case, because of its legal setting, the original complainant, Post, whose role corresponds to the plaintiff in the other cases, is named second. We shall, however, refer to Post as the plaintiff and Pierson as the defendant to maintain consistency of role with the other cases.

- *Young v Hitchens (1844).* In this English case, Young was a commercial fish-

[7] Berman and Hafner used *purposes* rather than *values*, but they functioned in the same way. We will use *purpose* and *value promoted* as synonymous.

erman who spread a net of 140 fathoms in open water. When the net was almost closed, Hitchens went through the gap, spread his net and caught the trapped fish. The case was decided for Hitchens. The basis for this was that Young had never had possession of the fish, and that it was not part of the court's remit to rule as to what constituted unfair competition.

Later work [39] also included four other cases in the discussion:

- *Ghen v Rich (1881).* In this Massachusetts case, Ghen was a whale hunter who harpooned a whale which subsequently was not reeled in, but was washed ashore. It was found by a man called Ellis, who sold it to Rich. According to local custom, Ellis should have reported his find, whereupon Ghen would have identified his lance and paid Ellis a fee. The court found for Ghen.
- *Conti v ASPCA*[8] *(1974).* In this New York case, Chester, a parrot owned by the ASPCA, escaped and was recaptured by Conti. The ASPCA found this out and reclaimed Chester from Conti. The court found that they were within their rights to do so.
- *New Mexico vs Morton (1975)* and *Kleepe vs New Mexico (1976).* These two cases concerned the ownership of unbranded burros normally present on public lands, which had temporarily strayed off them. Both were won by the state.

These seven cases were formalised as a Dung style AF in [23] and this was also used in [32]. It is shown in Figure 3.

The twenty six arguments, the arguments they attack and the values associated with them in [32] are shown in Table 1.

The basic approach in [23] was to remove the arguments not applicable to a particular case and then consider preferred extensions. Then if argument A was sceptically acceptable, the plaintiff would win, but otherwise the defendant would win (the burden of proof is on the plaintiff). This, however, is not straightforward in the Dung style AF since there are even-length cycles in the AF, and so there will be multiple preferred extensions, some with A and some without.

The cycles in question are:

- the two-cycle M-O, which arises in *Pierson*
- the four-cycle B-T-S-E, which arises in *Young*
- the four-cycle B-T-S-F, which arises in *Young*

[8] The American Society for the Prevention of Cruelty to Animals.

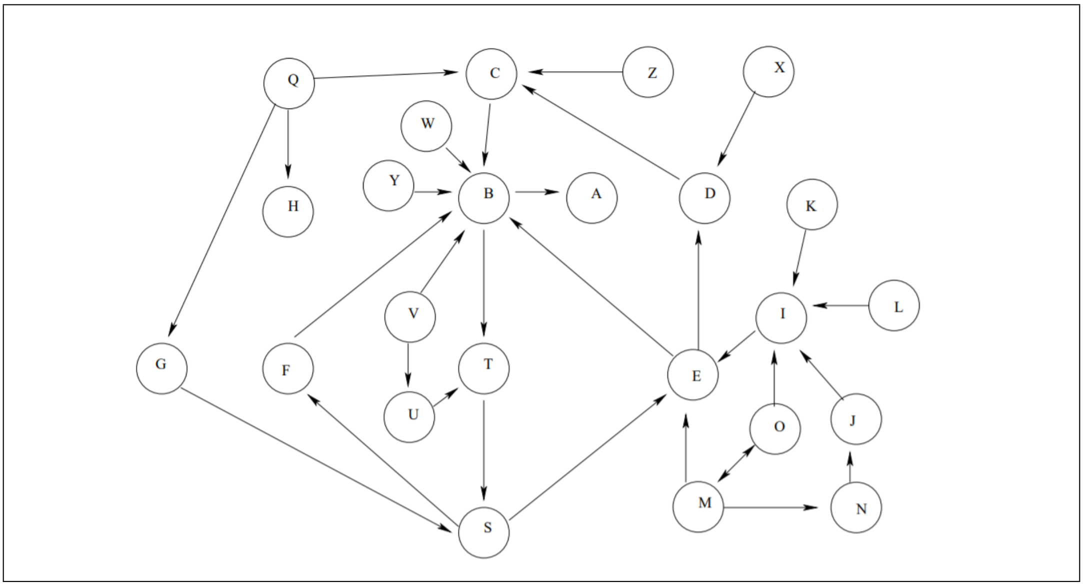


Figure 3: Dung style AF for wild animals cases, as given in [23]

This is precisely the situation for which Berman and Hafner commended the use of values: we need to choose between M, which promotes *clarity*, and O which promotes *useful activity*. In the actual case of *Pierson*, clarity was chosen, so that M was able to resist the attack of O, and so A was not in the preferred extension.

In the case of the two four-cycles that appeared in *Young*, the case was in fact resolved by the acceptance of U, which claimed that deciding what constituted unfair competition was outside the remit of the court. With T defeated, S defeats F, and so defends B. Similarly, S also defeats E and so B is acceptable. Now B defeats A and so the defendant won. Note that V was absent from *Young*. It was, however, present in *Ghen*, which concerned whaling, an industry long governed by clear conventions. Here the courts felt that just as it was not in their remit to determine what was unfair competition, neither could they overturn established conventions on the matter. Thus V was able to defeat U and B and so enable the plaintiff to win. This was forced in the standard AF, but in a VAF the attack from U to T can be resisted by preferring the value of *economic activity* to that promoted by restricting the court's remit, which would enable Young to win, even in the absence of an applicable convention. Such a shift in attitude may well occur (attitudes to regulation of competition swing back and forth), and so *Young* may at some future time be overturned. This illustrates a feature of value-based argumentation in law: because value preferences can change, the outcome of a case may be different at

CL = Clear law, UA = Useful activity, PR = Protect property rights, EA = Economic activity, CR = The court should not make law

ID	Argument	Attacks	Values
A	Pursuer had right to animal		claim
B	Pursuer not in possession	A, T	CL
C	Owns the land so possesses animals	C	PR
D	Animals not confined by owner	C	
E	Effort promising success made to secure animal made by pursuer	B, D	CL
F	Pursuer has right to pursue livelihood	B	EA
G	Interferer was trespassing	S	PR
H	Pursuer was trespassing	F	PR
I	Pursuit not enough (Justinian)	E	CL
J	Animal was taken (Justinian)	I	CL
K	Animal was mortally wounded (Puffendorf)	I	CL
L	Bodily seizure is not necessary (Barbeyrac), interpreted as animal was brought within certain control (Tompkins)	I	UA
M	Mere pursuit is not enough(Tompkins)	E, O	CL
N	Justinian is too old an authority (Livingston)	J	
O	Bodily seizure is not necessary (Barbeyrac), interpreted as reasonable prospect of capture is enough (Livingston)	I, M	UA
Q	The land was open	G, H, C	PR
S	Defendant in competition with the plaintiff	E, F	EA
T	Competition was unfair	S	EA
U	Not for courts to regulate competition	T	CR
V	The iron holds the whale is an established convention of whaling	B, U	CR
W	Owners of domesticated animals have a right to regain possession	B	PR
X	Unbranded animals living on land belong to owner of land	D	PR
Y	Branding establishes title	B	
Z	Physical presence (straying) insufficient to confer title on owner	C	CL

Table 1: **Arguments in the Wild Animal Cases.**

different times and in different jurisdictions. This captures the essence of the role of values in this kind of legal reasoning. A more elaborate discussion in [32] also investigates the role of intermediate concepts [60] in moving from facts to legal conclusions.

Further discussions of value-based reasoning in the wild animals cases can be found in [26] and [12]. In [80] an additional case, *Popov v Hayashi* [8] was included. This celebrated case[9], concerned a record breaking home run baseball hit by Barry Bonds of the San Francisco Giants. There was a struggle amongst crowd members over its possession. Popov first laid his glove on the ball, but Hayashi emerged from the ensuing melee with the ball. The wild animals cases were cited in the case, analogies being drawn between the hunted animals and the "fugitive baseball" [52]. This case and the wild animals cases were further discussed in [27].

4.3 Domain-specific application: e-participation

Another domain in which value-based argumentation has proved effective is e-participation. Political disputes often turn on disagreement as to values, and so this is a natural way to model such disputes. In PARMENIDES [18], a policy was presented for critique by members of the public through a software tool. The policy was presented as an instantiation of the practical reasoning scheme AS1 given above. Thus the policy was presented in terms of an understanding of the current situation and what it was meant to achieve in terms of facts, goals and values. The user was then given the opportunity to critique the policy in terms of relevant critical questions characteristic of the scheme[10]. In this way disagreement with the policy could be made precise, and different motives for disagreement identified. For example, different people might doubt whether the current situation was indeed as suggested, others might doubt that the policy would achieve its ends, and yet others might oppose these ends because rejecting the values they promote. PARMENIDES was further developed in [44] and later PARMENIDES formed the basis for the development of the Structured Consultation Tool (SCT) [34], produced as part of the IMPACT project[11]. The SCT enabled not only a policy proposal to be critiqued, but also for the users to make proposals of their own, which could be automatically critiqued by instantiating critical questions [78].

We will base our example in this chapter on that of [22], which was also used in [81]. The example is an issue in UK Road Traffic policy. The number of fatal road accidents is an obvious cause for concern, and in the UK there are speed restrictions

[9] It was the subject of the 2004 comedy documentary film *Up for Grabs* https://www.imdb.com/title/tt0420356/

[10] Not all critical questions were used: for example, those relating to the components of the model were not appropriate.

[11] Integrated Method for Policy Making Using Argument Modelling and Computer Assisted Text Analysis, in the European Framework 7 project (Grant Agreement No 247228) in the ICT for Governance and Policy Modeling theme (ICT-2009.7.3).

on various types of road, in the belief that excessive speed causes accidents. The policy issue which we will consider is how to reduce road deaths. One option is to introduce speed cameras to discourage speeding.

Following [13] the first step is to build a model. In [22] there was an extensive discussion of how to construct the model on the basis of responses received to a Green Paper[12]. Like [81] we will focus on the final refinement of the model presented in [22], which includes responses from road safety organisations, motoring lobby groups, representations about financial constraints and civil liberties groups.

We now present the AATS+V. States are composed from the propositions considered relevant. In the model of [22] the propositions that were considered are (given as pairs of positive and negative propositions):

R: The number of road deaths is acceptable; There are more road deaths than there should be.

S: Many motorists break the speed limits; Speed limits are generally obeyed.

P: Privacy is respected; There are additional intrusions on privacy.

These three propositions give rise to, potentially, eight states. We may, if we wish, exclude one or more of these as impossible. For example, if we believe that it is impossible that the number of road deaths is acceptable and yet many motorists break the speed limits, we may introduce constraints on states to filter it out. In [81], we specify all the possible states available for consideration. One state is designated as the current state:

- Many motorists break the speed limits $\wedge$ There are more road deaths than there should be $\wedge$ Privacy is respected.

We consider the impact of changes of state in terms of three values:

L: human life (Life);

B: the financial cost to the Government (Budget); and

F: the impact on civil liberties (Freedom). Here the principal concern is the impact on privacy.

[12] A Green Paper is a Government publication issued as part of a consultation process that details specific issues, and then points out possible courses of action in terms of policy and legislation in order to receive feedback from interested parties.

The main agents involved are the *Government* (G), and *Motorists* (M), each considered as a body. In some cases the consequences of action are indeterminate (or at least cannot be determined using the elements we are modelling). To account for this we introduce a third agent, termed *Nature* (N). The action ascribed to Nature determines the outcomes of the actions of the other agents, where these outcomes are uncertain or probabilistic. We take the Government to be the independent agent, the one attempting to fulfill its values, while the actions of the Motorists and Nature are relative to its choices.

The Government has three actions: introducing speed cameras (G_1), educating motorists (G_2), or doing nothing (G_3). Motorists may reduce their speed or do nothing. Nature has two actions according to which fatal accidents are or are not reduced as a result of the Government and motorist actions. Actions are assumed to be always carried out together with other agents, represented as joint actions. The joint actions available are:

j_0: Government does nothing, motorists do nothing and nature does nothing.

j_1: Government introduces cameras, motorists do nothing and nature does nothing.

j_2: Government introduces cameras, motorists reduce speed and nature reduces accidents.

j_3: Government introduces cameras, motorists reduce speed and nature does nothing.

j_4: Government educates motorists, motorists reduce speed and nature reduces accidents.

j_5: Government educates motorists, motorists do nothing and nature reduces accidents. (Being more skilled, the drivers can cope with their excessive speed).

Finally, we have transitions, which relate a source state, a destination state, a joint action, a list of values promoted, and a list of values demoted. The joint action can only be carried out where, in some sense, the conditions for doing the action are met (e.g. where motorists are not speeding, then they cannot reduce speed) and result in a state that also makes sense (e.g. where motorists reduce speed and nature reduces accidents, then motorists are not speeding and accidents are reduced). We can presume that accidents are always reduced when motorists are educated since either they do not speed or can control their vehicles better. The transitions from q_0 are shown in Table 2. We are not interested in what happens in subsequent states. The whole AATS+V is shown as Figure 4.

	j0	*j1*	*j2*
q0	⟨q0,_,_⟩	⟨q0,+B,-F⟩	⟨q5,+L+C,-F⟩
	j3	*j4*	*j5*
q0	⟨q6,+C,-F⟩	⟨q2,+L+C,-B⟩	⟨q3,+L,-B⟩

Table 2: Final Transition matrix.

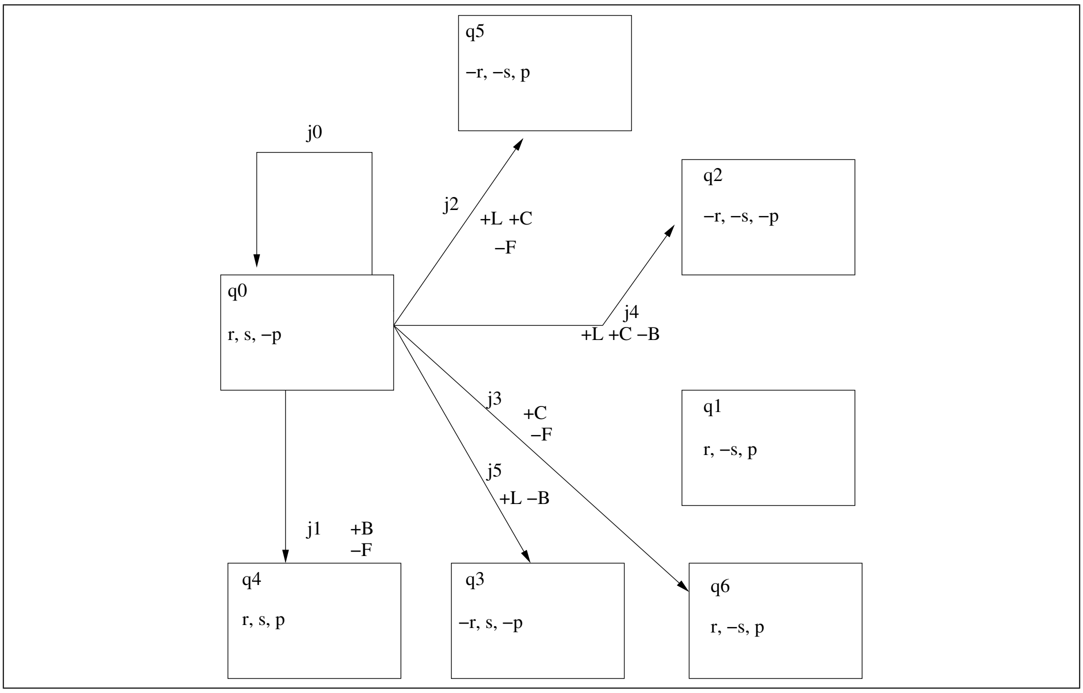


Figure 4: AATS+V for speed camera debate, as given in [22]

On the basis of this model, it seems that introducing speed cameras is a reasonable proposal. The hope is that this would induce motorists to cut their speed, and that the number of accidents would fall, so that j_2 is performed and q_5 is reached. This can be expressed in the form of AS1:

> The current state is: Many motorists break the speed limits ∧ There are more road deaths than there should be ∧ Privacy is respected.
>
> The action is: The government should introduce speed cameras.
>
> The destination state is: Speed limits are generally obeyed ∧ The number of road deaths is acceptable ∧ There are additional intrusions on privacy.

The values promoted are: Life, Compliance

Note that the Government is expressing a preference for Life and Compliance over Freedom, which is demoted by the action.

This proposal can now be the subject of criticisms. For example,

CQ1 There might be disagreement as to the current situation: it would be possible to deny that many motorists broke the speed limits, or to claim that the number of road deaths was, in fact, acceptable.

CQ2 It might be argued that the action would not have the stated effects. Introducing speed cameras could reach q_4 or q_6 which would fail to promote one or both of our values.

CQ9 The action may demote a value. For example, freedom is demoted by the proposal.

CQ11 Other values can be promoted. There is no ground for this criticism in our example.

CQ13 It might be argued that the model is flawed and the proposed action is not possible. For example, it might be argued that the installation of speed cameras on the scale proposed was simply infeasible.

CQ17 Perhaps one or other of the agents will not perform the hoped for outcome. For example, it might be argued that reducing speed will not in fact reduce accidents and so the joint action will be j_3 leading to q_5 and so failing to promote life.

Using these methods to generate arguments, we can perform a full analysis. There are five arguments to perform an action from instantiating AS1.

PR1 We should perform G_1 to reach q_5 to promote L

PR2 We should perform G_1 to reach q_5 or q_6 to promote C

PR3 We should perform G_1 to reach q_4 to promote B

PR4 We should perform G_2 to reach q_2 or q_3 to promote L

PR5 We should perform G_2 to reach q_2 to promote C

Two arguments to refrain from an action are generated by a contrapositive form of AS1:

NPR1 We should not perform G_1 to avoid q_5 and q_6 since this would demote F

NPR2 We should not perform G_2 to avoid q_2 and q_3 since that would demote B

We accept that q_0 is the current state, and that other features of the model are correct. But we still have CQ17, which gives rise to three objections:

Ob1 Motorists may choose M_0 not M_1: attacks PR1, PR2 and PR5.

Ob2 Reducing speed may not reduce accidents and deaths. Attacks PR1.

Ob3 Motorists may choose M_1 not M_0: attacks PR3.

We now reach the final stage, when we weigh the merits and demerits of competing options, which requires us to identify the attacks between arguments. One source of attack is that a value is demoted: thus NPR1 attacks PR1, PR2 and PR3, and NPR2 attacks PR4 and PR5. Another source of attack, giving rise to symmetric attacks, is an alternative way of promoting the same value: thus PR1 and PR4 mutually attack, and PR2 and PR5 mutually attack. Finally we have different actions promoting different values: PR1 and PR5 and PR2 and PR4 mutually attack in this way. Finally we can have attacks which question the motive put forward: if PR1 is advanced to justify speed cameras, some may argue that the real expectation is that q_4 will be reached and that the real motive is to save money, rather than lives. This, however, does not challenge the action, but the justification, and we will not include these attacks here. Finally we have arguments representing the actual responses of motorists and nature to the introduction of speed cameras. These will form two two-cycles. We can now evaluate the arguments using a VAF. The VAF is shown in Figure 5.

On the left of the diagram are the two epistemic questions that need to be resolved. In default of anything better let us assume that, on the best evidence available, it is reasonable to expect that motorists will in fact reduce their speed, and that reducing speed will indeed lessen accidents and deaths. Having resolved these two cycles, we have answered the attacks from Ob1 and Ob2, while Ob3 is no longer attacked and will defeat PR3. When arguments are defeated, we can remove them and their attacks (and attacks on them) from the VAF to obtain the simpler VAF, as shown in Figure 6. Note that if we had made different assumptions about the epistemic questions then a different VAF, and ultimately a different position, would result from this simplification. When an argument is not defeated, but its attack is resisted by a preferred argument, we mark it as *ineffective.* We cannot ignore it, since we have no argument to defeat it, but we will not act upon it. There

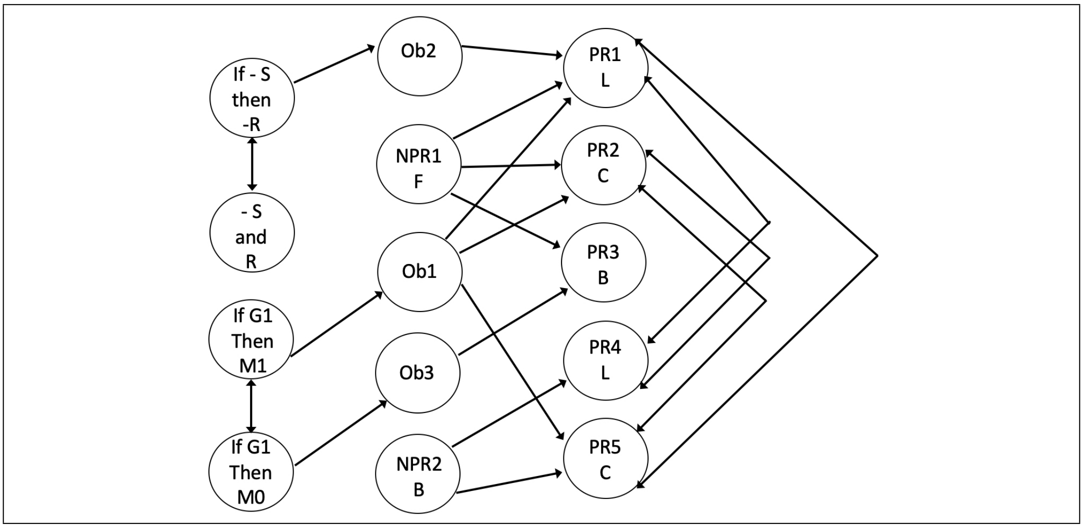


Figure 5: VAF for speed camera debate

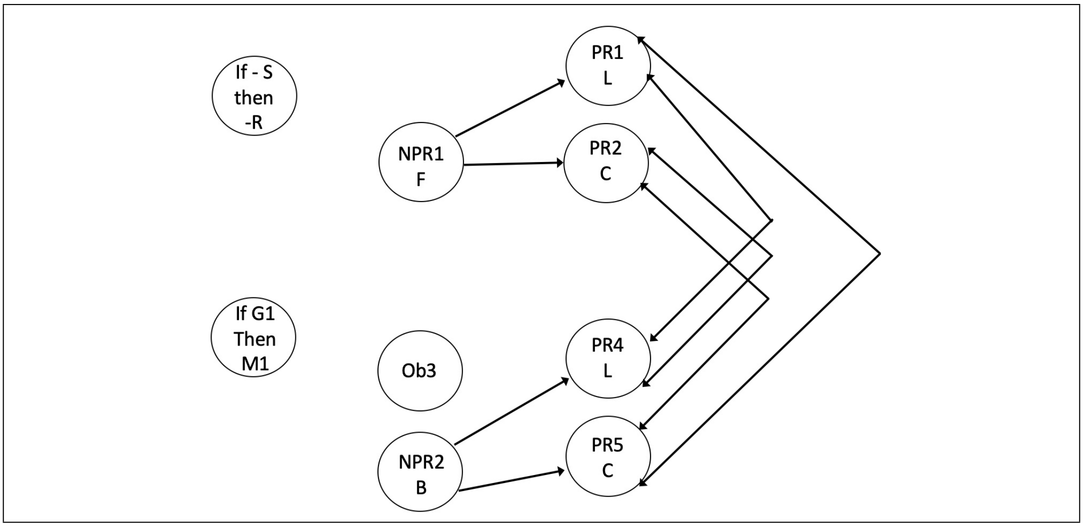


Figure 6: VAF for speed camera debate after epistemic choices

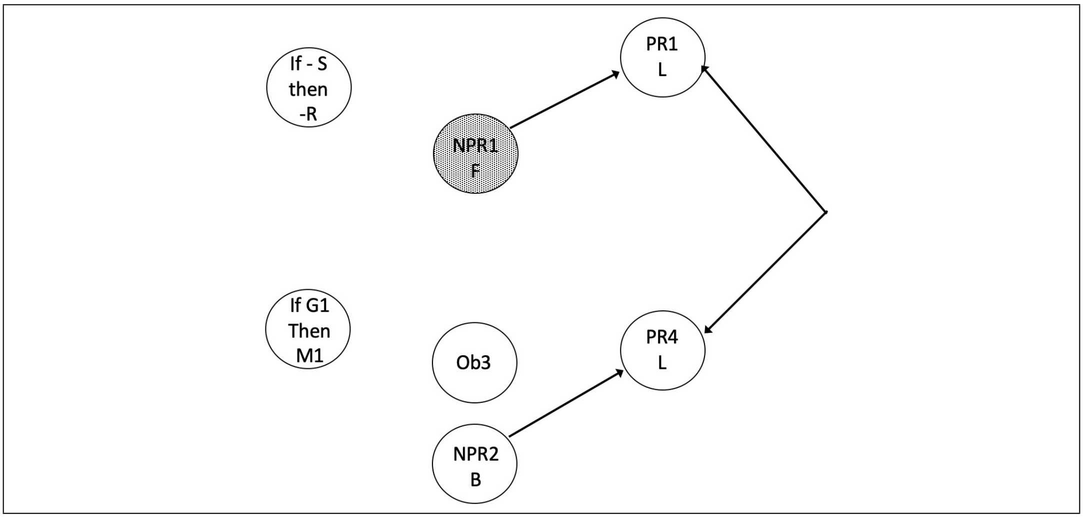


Figure 7: VAF with preferences $L > F > C$

are no such arguments as yet, since we have not yet exercised preferences, but only chosen between different factual assumptions.

We next consider the two negative arguments based on PRAS2; once we have reached Figure 5 by resolving the epistemic questions, these are unattacked. These arguments will therefore succeed in defeating the arguments they attack unless the value of the attacked argument is preferred to that of the attacker. For NPR1 we must therefore consider Privacy/Freedom against Life to resolve the attack on PR1, and against Compliance to resolve the attack on PR2. A reasonable order would seem to be $L > F > C$: saying that intrusion on privacy is a necessary evil to save lives, but would not be acceptable simply to ensure compliance with speed limits without other gains. NPR1 thus becomes ineffective, which we show in the diagram by shading the argument node. This yields the VAF in Figure 7.

The final question to resolve is whether PR4 can be accepted given NPR2: that is, can we prefer L to B? Unfortunately we are regarding budget as a hard constraint and so we must answer that $B > L$. This means that PR4 falls, leaving a preferred extension for an audience of $B > L > F > C$ comprising: the two factual assumptions, that motorists will reduce their speed, and that less speed means fewer accidents and deaths; the accepted course of action to install cameras to save lives; and two other considerations, that privacy must unfortunately be lessened (represented by the undefeated but ineffective argument), and that budgetary constraints preclude education as an alternative (represented by Obj3). Of course similar reasoning with different assumptions and different value orders would produce different results. If we assumed that motorists would continue to speed with the same value order, we would still install the cameras, but this time on the basis of PR3. If we made the original assumptions but used the value order $B > F > L > C$, we could do nothing, since we would have no way of saving lives without infringing privacy that we could afford, and if we had the value order $F > B > L > C$, we would educate motorists rather than install cameras.

Finally, if we prefer life to freedom, but money is available so that it was possible to prefer L to B, we would have two equally valued arguments, PR1 and PR4, neither attacked except by each other. In this case we should be inclined to choose PR4, since this would mean that the undefeated NPR1 would no longer have to coexist with an argument it attacks, so that it no longer need be regarded as ineffective[13]. In this way we are able to respect the value of privacy, even though F is not preferred to L.

Considerations of these varied alternatives allow us to see how the policy positions favoured depend very critically on how we rank values: the acceptability of a proposal will often depend on whether the public mood has been correctly judged in this respect.

4.4 Domain-specific application: behavioural economics

Value-based reasoning has also been used to explore two "games" used in behavioural economics, the *Dictator Game* [51] and the *Ultimatum Game* [68]. Classical economic theory assumes that people will behave in the manner of "economic man" described as follows by John Stuart Mill [62]:

> [Economics] is concerned with him solely as a being who desires to possess wealth, and who is capable of judging the comparative efficacy of means for obtaining that end.

However experiments performed in behavioural economics cast doubt on this key assumption. In the Dictator Game one player is given a sum of money and is then asked to give the second player as much or as little of it as he wishes. Classical economics would suggest that the player will give nothing, so maximising his own return. Experimentally, however, the results suggest otherwise: most players will give something to the other, sometimes as much as half. No studies report that the canonical model was observed. In one typical study [53], given \$10 to distribute, 79% of participants gave away a positive amount, with 20% giving away half. The mode

[13] One disadvantage of the approach of [5] in which arguments which resist their attacks also defeat them is that it fails to distinguish between defeated arguments and those which must be acknowledged even though not followed.

sum given away in that study was $3. The explanation is that other values come into play here: suggestions include concern for the other, simple generosity, concern for image (no one likes to be thought selfish). This game was thoroughly explored using value-based reasoning in [33]: here we will discuss the more interesting Ultimatum Game.

In the Ultimatum Game the first player is also given a sum of money and asked to decide how much he wishes to offer to the other player. But this time the second player can refuse, in which case both get nothing. Now classical economics suggests that the first player will offer the smallest amount possible and the second player will accept it because, for economic man, anything is better than nothing. As with the Dictator Game, these expectations are not borne out in practice. For example, Nowak and colleagues report that the majority of proposers offer 40âĂŞ50% and about half of responders reject offers below 30% [67]. These results are robust, and, with some variations, are replicated in all the many studies. Oosterbeek *et al* [68] report a meta-analysis of 37 papers with 75 results from Ultimatum Game experiments, which have an average of 40% offered to the responder. The experiments of [55], carried out over 15 small-scale societies in 12 countries over five continents, report mean offers between 26% and 58%, and note that in some societies there is considerable variation in which offers are rejected: however, again, none suggests that the canonical model is followed by those making and responding to offers. The Ultimatum Game was modelled in [33] and [15].

First we must model the game as an AATS+V. Obviously the states must include the money held by the two agents. We also wish to represent the reactions of the two players. When the offer is made, it is important whether the second player perceives it as fair, or as insulting. We therefore use a proposition which is true when the second player is annoyed by the offer made. At the end of the game we can consider the reaction of the first player. In particular, if the offer is rejected, a first player who made an ungenerous offer is likely to feel regret that he did not offer more. We therefore use a fourth proposition to record whether the first player feels regret.

Next we turn to actions. Obviously we need that the first player can offer n% of the available sum to the second player and that the second player can accept or reject it. The reception the offer receives will, however, depend critically on the size of n. We will therefore distinguish four cases: where $n > 50$, where $n = 50$, where $n > 0$ but < 50 and where $n = 0$. We should also recognise that the two actions are not chosen simultaneously, and that the choice to accept or reject will depend on how the second player reacts to the offer of the proposer. We therefore introduce a third action, in which the second player chooses a threshold, t, above which he will regard the offer as just, and below which he will feel insulted. We will assume that $t > 0$ and $t < 50$, discounting players who will not be satisfied with even an equal share. While the second player accepts and rejects, the first player can do nothing. This gives the set of joint actions shown in Table 3.

Joint Action	Player 1	Player 2
j1	A1:Offer > 50	B1:Set t < 50
j2	A2:Offer 50	B1:Set t < 50
j3	A3:Offer n < 50 and > 0	B1:Set t < n
j4	A3:Offer n < 50 and > 0	B1:Set t > n
j5	A5:Offer n = 0	B1:Set t > 0
j6	A4:Do nothing	B2:accept
j7	A4:Do nothing	B3:reject

Table 3: Joint Actions in the Ultimatum Game

Now we must identify some values and the transitions which promote and demote them. First there is economic value, the money, which we shall call M. This can be promoted in respect both of player 1 (M1) and in respect of player 2 (M2). These values are promoted to different degrees according to the size of the player's share. From the literature it appears that some people seem to value fairness, which we shall call E for equality. This is either promoted or not. Third we have the value of generosity (G), which again has been identified as a motivation by various experimenters. Whereas M will be promoted to varying degrees according to the amount of money, E is either promoted or not. What of G? Experimental evidence suggests that the impact of G does not increase as the amount given increases: we will therefore consider that G, like E, is either satisfied or not, and that any effect of the size of the gift is reflected in M2. Finally either player may be content with the outcome, and we represent this as C1 and C2. Again we will not model degrees of contentment. The AATS+V is shown in Figure 8.

Here will focus on the decision of the second player: the first player needs to think about this since the main aim of an offer is to have it accepted. The VAF for the second player is shown in Figure 9.

What the second player will do will depend on how it orders its values. Thus an offer above 50, or below 50 but above the second player's threshold of acceptability (states q_1 and q_3), will only be rejected if the player prefers equality to both its own and the other player's, money: $E > M1, M2$. Given the set of values we have used, we would expect any player to accept an offer of half the sum, since rejecting in

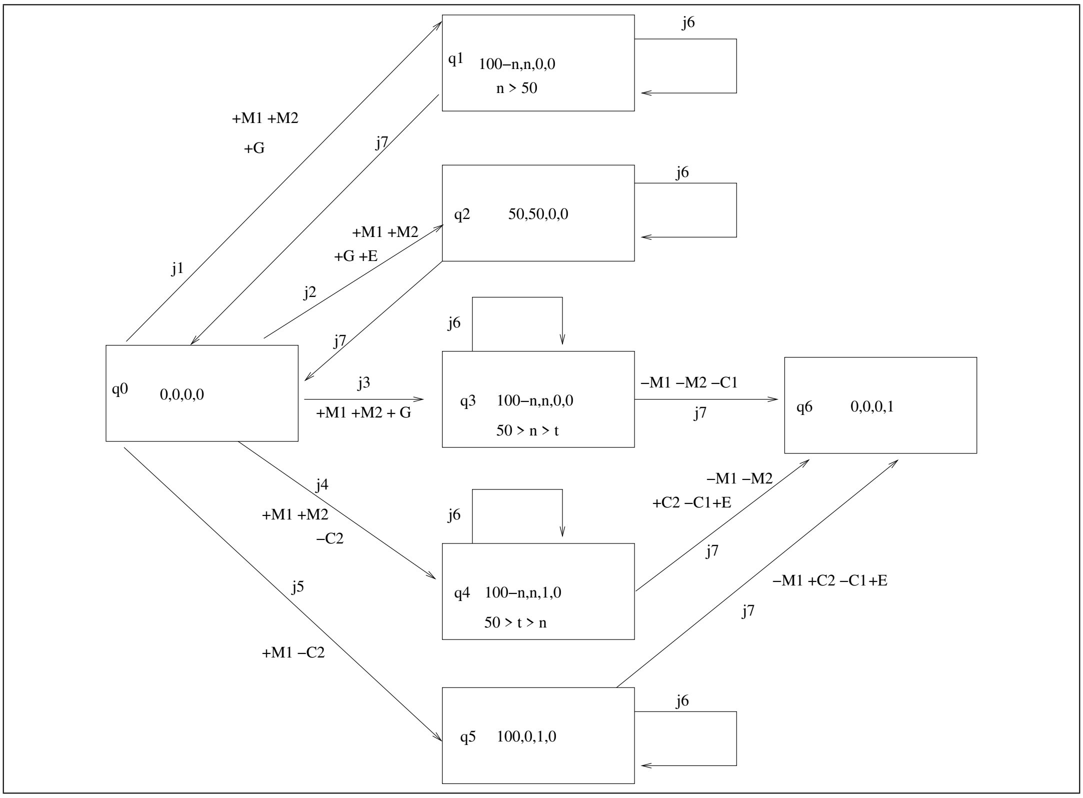


Figure 8: AATS+V for Ultimatum Game, as given in [33]

q_2 promotes nothing and demotes money for both players. If the second player is insulted by a non-zero offer and so is in q_4, however, he has a choice of whether to punish the first player and so restore its own pride, or to take the money. Normally we would expect that the player will prefer its own money and its own contentment to the money and contentment of the other agent, and so require $M2 > C2 > M1, C1$ for acceptance, or $C2 > M2 > M1, C1$ for rejection. If E is preferred to both $M2$ and $C2$ the second player will also reject the offer, but here motivated by a desire for equality, rather than the insult. Finally if a zero offer is made we would expect rejection, either because of the insult, or because equality is desired. Indeed a zero offer will only be accepted if the second player prefers the others player's money or contentment to its own contentment: $C1, M1 > C2$. This would be an extreme example of altruism, and we would expect it to be rare. This ordering would also lead to acceptance in q_4.

What the second player will do is crucial. In [15] the Ultimatum Game was used

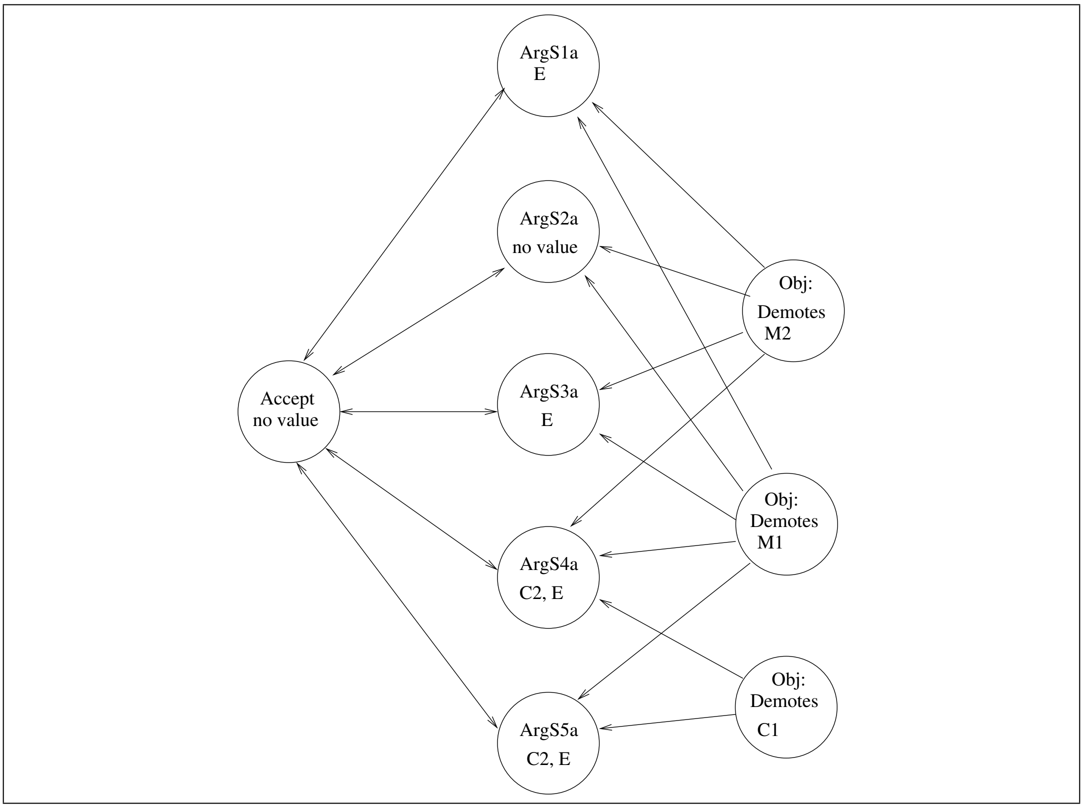


Figure 9: VAF for second player in Ultimatum Game, as given in [33]

to explore how an agent can take account of the expected actions of others. There the three actions of our above model were compressed into a set of joint actions as shown in Table 4.

There we say that player one can make a very high offer (vho) of more than half, an equal offer (eo) of half, a fair offer (fo) at the threshold for the second player, or a low offer (lo), below that threshold. All of these may be accepted or rejected by the second player, giving eight joint actions, promoting and demoting values as shown. Note that equality cannot be promoted, since the initial state is one of equality. From this table we can see why most players will make at least a fair offer: only if the first player is desperate to “get one over” the other will a low offer be made, since only a low offer promotes $C1$ but carries with it a high probability of demoting $M1$ and $C1$. How high the offer will go depends on how much the player values the wealth and happiness of the other, and whether it values a feeling of generosity.

In [55] 15 small scale societies from various parts of the world were studied, and

Joint Action	Proposal	Response	Promoted	Demoted
j1	vho	accept	M1,M2,G, C2	E
j2	vho	reject	G	M1,C1
j3	eo	accept	M1,M2,G,C2	
j4	eo	reject	G	M1,C1
j5	fo	accept	M1,M2	E
j6	fo	reject		M1
j7	lo	accept	M1,M2,C1	E,C2
j8	lo	reject		M1,C1

Table 4: Value promotion and demotion in the Ultimatum Game

it emerged that different groups behave differently. It was suggested that the different societies' actions in the Ultimatum Game could be accounted for in terms of the degree of cooperation and degree of commercial exchange found in daily life. We can relate these characteristics to a value profile. Suppose we associate the value of generosity with the cooperative groups such as the whale hunting Lamelara, and the recognition of C2 (the need to maintain good relations with the other) with commercial exchange. Those who do not engage in cooperative or exchange activities, we term solitary. In [15] it was found that using value profiles representing these three life styles predicted offers and rejections that are very close to the empirical results of [55].

4.5 Other applications

As well as these examples, value-based reasoning has been demonstrated using examples in medicine [19], health advice [75] and [47], ontology alignment [59] and [74], an account of the emergence of norms [37] and discussions of ethics [14]. Most recently in [30] value-based reasoning has been used as the basis of a novel computational account of virtue ethics in agent systems.

In general, value-based reasoning can be used to model argumentation and reasoning in any domain where the direction of fit is from an agent's desires or needs to the world; any situation in which reasoning about action is required. Such reasoning is pervasive, covering many of the most important aspects of life: from everyday choices such as where to eat or how to travel, to law and politics, and fundamental questions of how we should live.

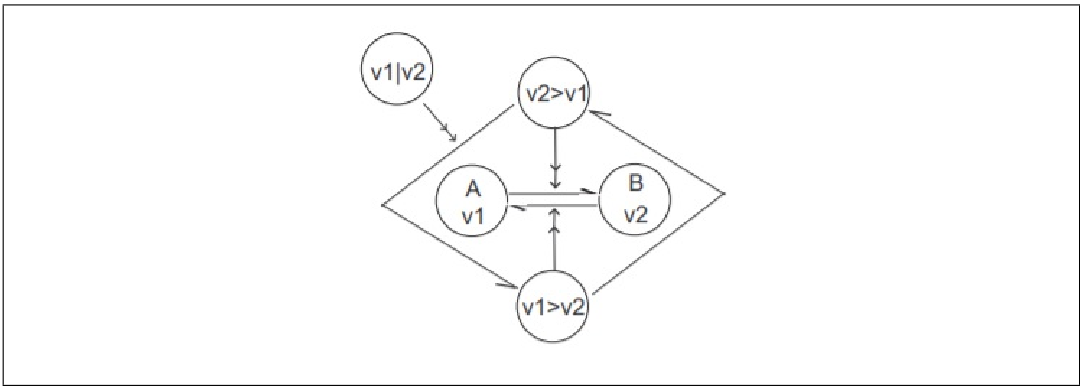


Figure 10: Value-based conflict in extended argumentation framework, as given in [64]

5 Value-based reasoning at the meta-level

Modgil [63] introduced an elegant and general way of handling preferences: instead of assigning different strengths to arguments, he permitted attacks to themselves be attacked. Such frameworks he termed *Extended Argumentation Frameworks* (EAF). This meant that an attack was unsuccessful not according to whether it was attacking a stronger argument, but according to whether it was itself defeated by some other argument.

The relation between VAFs and EAFs was explored in [64]. A conflict between two arguments is shown as an EAF in Figure 10. There the value preferences are represented as arguments, attacking attacks which require the other preference to succeed. These value preference arguments will, of course, mutually attack. The desired audience represented as an ordering on the values will attack one of these attacks, resolving the framework.

Frameworks of the sort shown in Figure 10 can now be rewritten as standard Dung-style argumentation frameworks using meta level arguments. If we replace arguments by the fact that they are acceptable, e.g. A by *A holds*, and introduce arguments that arguments do not hold ($\overline{A}$) and that one argument defeats another ($\overrightarrow{AB}$), we can rewrite Figure 10 as Figure 11.

Now an attack $\overrightarrow{AB}$ may fail in two ways: either A may be defeated so that $\overline{A}$ defeats it, or there may be a preference argument that defeats it. There are clear simplifications in this rewriting in that standard AFs can be used instead of the more complicated VAFs and EAFs. The use of EAFs in value-based reasoning was discussed in [31], and its application to the representation of norms in [38]. A full

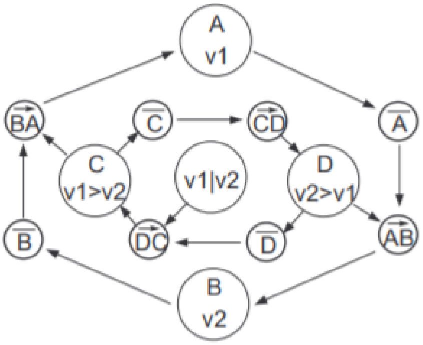


Figure 11: Meta level rewriting of Figure 10 to get a standard AF, as given in [64]

discussion of meta level argumentation can be found in [65].

6 Concluding remarks

In this chapter we have discussed value-based reasoning. Philosophically it models reasoning where the direction of fit is from an agent's desires to the world: that is where an agent is choosing how to act in order to promote its values, and this covers all domains involving an element of practical reasoning, reasoning about what should be done.

Value-based reasoning was originally presented as a form of abstract argumentation extending Dung's original framework by giving arguments the additional property of *promoting a value*, and evaluating the arguments according to an ordering on those values.

Although there are some theoretical results, the main motivation for value-based reasoning was always applications, especially law where [43] had drawn attention to the role of values in legal decisions, and [41] had incorporated values into theories of case law for particular areas of law. This emphasis on applications was facilitated by the development of a means of doing structured value-based argumentation, based on an argumentation scheme and critical questions semantically underpinned by a form of state transition diagram, AATS+V.

Because of the importance of applications, we have devoted much of this chapter to a detailed discussion of four application domains: general problem solving, law, e-participation and behavioural economics.

Extended argumentation frameworks [63] offer a means of generalising argumentation involving preferences. Value-based argumentation frameworks fit very well with this framework, since they can be systematically rewritten as standard AFs using meta level arguments describing the status of arguments in the VAF, the value preferences, and the audience concerned. Moving to meta level argumentation, however, does not remove the need for structured value-based argumentation, which is still needed to generate the arguments and attacks. This combination is used in [31].

The theory of value-based argumentation is fairly well understood, but its potential for modelling applications continues. As a means of representing problems in areas where values are crucial, such as ethics, law and politics, value-based reasoning offers a tried and tested solution.

Acknowledgments

In the many years that we have been working on computational models of value-based argument, we have collaborated with a wide range of people spanning many of the topics covered in this chapter. We are grateful to all colleagues with whom we have worked to progress research in this area: Latifa Al-Abdulkarim, Floris Bex, Elizabeth Black, Dan Cartwright, Alison Chorley, Sylvie Doutre, Paul Dunne, Floriana Grasso, Yanko Kirchev, Loredana Laera, Peter McBurney, Rolando Medellin-Gasque, Sanjay Modgil, Fahd Saud Nawwab, Samer Nofal, Henry Prakken, Giovanni Sartor, Valentina Tamma, Doug Walton, Maya Wardeh, Adam Wyner.

References

[1] Vincent Aleven and Kevin D Ashley. Doing things with factors. In *Proceedings of the 5th International Conference on Artificial Intelligence and Law*, pages 31–41, 1995.

[2] Rajeev Alur, Thomas Henzinger, and Orna Kupferman. Alternating-time temporal logic. *Journal of the ACM (JACM)*, 49(5):672–713, 2002.

[3] Leila Amgoud and Claudette Cayrol. On the acceptability of arguments in preference-based argumentation. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 1–7. Morgan Kaufmann Publishers Inc., 1998.

[4] Leila Amgoud and Claudette Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34(1-3):197–215, 2002.

[5] Leila Amgoud and Srdjan Vesic. A new approach for preference-based argumentation frameworks. *Annals of Mathematics and Artificial Intelligence*, 63(2):149–183, 2011.

[6] G.E.M. Anscombe. On practical reasoning. In Joseph Raz, editor, *Practical Reasoning*, pages 33–45. Oxford University Press, 1978.

[7] Katie Atkinson. *What Should We Do?* PhD thesis, University of Liverpool, 2005.

[8] Katie Atkinson. Introduction to special issue on modelling Popov v. Hayashi. *Artificial Intelligence and Law*, 20(1):1–14, 2012.

[9] Katie Atkinson, Priscilla Bench-Capon, and Trevor Bench-Capon. A strategy for efficient persuasion dialogues. In *International Conference on Agents and Artificial Intelligence*, pages 332–347. Springer, 2012.

[10] Katie Atkinson and Trevor Bench-Capon. Legal case-based reasoning as practical reasoning. *Artificial Intelligence and Law*, 13(1):93–131, 2005.

[11] Katie Atkinson and Trevor Bench-Capon. Action-based alternating transition systems for arguments about action. In *AAAI*, volume 7, pages 24–29, 2007.

[12] Katie Atkinson and Trevor Bench-Capon. Argumentation and standards of proof. In *Proceedings of the 11th international conference on Artificial intelligence and law*, pages 107–116, 2007.

[13] Katie Atkinson and Trevor Bench-Capon. Practical reasoning as presumptive argumentation using Action based Alternating Transition Systems. *Artificial Intelligence*, 171(10-15):855–874, 2007.

[14] Katie Atkinson and Trevor Bench-Capon. Addressing moral problems through practical reasoning. *Journal of Applied Logic*, 6(2):135–151, 2008.

[15] Katie Atkinson and Trevor Bench-Capon. Taking account of the actions of others in value-based reasoning. *Artificial Intelligence*, 254:1–20, 2018.

[16] Katie Atkinson, Trevor Bench-Capon, and Peter McBurney. Justifying practical reasoning. In *Proceedings of the Fourth International Workshop on Computational Models of Natural Argument (CMNA 2004)*, pages 87–90, 2004.

[17] Katie Atkinson, Trevor Bench-Capon, and Peter McBurney. Computational representation of practical argument. *Synthese*, 152(2):157–206, 2006.

[18] Katie Atkinson, Trevor Bench-Capon, and Peter McBurney. Parmenides: facilitating deliberation in democracies. *Artificial Intelligence and Law*, 14(4):261–275, 2006.

[19] Katie Atkinson, Trevor Bench-Capon, and Sanjay Modgil. Argumentation for decision support. In *International Conference on Database and Expert Systems Applications*, pages 822–831. Springer, 2006.

[20] Katie Atkinson, Trevor Bench-Capon, and Douglas Walton. Distinctive features of persuasion and deliberation dialogues. *Argument & Computation*, 4(2):105–127, 2013.

[21] Katie Atkinson and Trevor JM Bench-Capon. Taking the long view: Looking ahead in practical reasoning. In *COMMA*, pages 109–120, 2014.

[22] Katie M Atkinson, Trevor JM Bench-Capon, Dan Cartwright, and Adam Z Wyner. Semantic models for policy deliberation. In *Proceedings of the 13th International Conference on Artificial Intelligence and Law*, pages 81–90, 2011.

[23] Trevor Bench-Capon. Representation of case law as an argumentation framework. *Proceedings of JURIX 2002*, pages 103–112, 2002.

[24] Trevor Bench-Capon. Value based argumentation frameworks. *arXiv preprint cs/0207059. Originally presented at NMR 2002*, 2002.

[25] Trevor Bench-Capon. Persuasion in practical argument using value-based argumenta-

tion frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[26] Trevor Bench-Capon. Try to see it my way: Modelling persuasion in legal discourse. *Artificial Intelligence and Law*, 11(4):271–287, 2003.

[27] Trevor Bench-Capon. Representing Popov v Hayashi with dimensions and factors. *Artificial Intelligence and Law*, 20(1):15–35, 2012.

[28] Trevor Bench-Capon. Dilemmas and paradoxes: cycles in argumentation frameworks. *Journal of Logic and Computation*, 26(4):1055–1064, 2014.

[29] Trevor Bench-Capon. HYPO's legacy: Introduction to the virtual special issue. *Artificial Intelligence and Law*, 25(2):1–46, 2017.

[30] Trevor Bench-Capon. Ethical approaches and autonomous systems. *Artificial Intelligence*, page 103239, 2020.

[31] Trevor Bench-Capon and Katie Atkinson. Abstract argumentation and values. In *Argumentation in artificial intelligence*, pages 45–64. Springer, 2009.

[32] Trevor Bench-Capon, Katie Atkinson, and Alison Chorley. Persuasion and value in legal argument. *Journal of Logic and Computation*, 15(6):1075–1097, 2005.

[33] Trevor Bench-Capon, Katie Atkinson, and Peter McBurney. Using argumentation to model agent decision making in economic experiments. *Autonomous Agents and Multi-Agent Systems*, 25(1):183–208, 2012.

[34] Trevor Bench-Capon, Katie Atkinson, and Adam Wyner. Using argumentation to structure e-participation in policy making. In *Transactions on Large-Scale Data-and Knowledge-Centered Systems XVIII*, pages 1–29. Springer, 2015.

[35] Trevor Bench-Capon, Sylvie Doutre, and Paul Dunne. Audiences in argumentation frameworks. *Artificial Intelligence*, 171(1):42–71, 2007.

[36] Trevor Bench-Capon and Sanjay Modgil. Case law in Extended Argumentation Frameworks. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, pages 118–127. ACM, 2009.

[37] Trevor Bench-Capon and Sanjay Modgil. Norms and value based reasoning: justifying compliance and violation. *Artificial Intelligence and Law*, 25(1):29–64, 2017.

[38] Trevor Bench-Capon and Sanjay Modgil. Norms and extended argumentation frameworks. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, pages 174–178, 2019.

[39] Trevor Bench-Capon and Edwina Rissland. Back to the future: Dimensions revisited. In *Legal Knowledge and Information Systems - JURIX 2001*, pages 41–52, 2001.

[40] Trevor Bench-Capon and Giovanni Sartor. Theory based explanation of case law domains. In *Proceedings of the 8th international conference on Artificial intelligence and law*, pages 12–21, 2001.

[41] Trevor Bench-Capon and Giovanni Sartor. A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence*, 150(1-2):97–143, 2003.

[42] Trevot Bench-Capon. Agreeing to differ: modelling persuasive dialogue between parties with different values. *Informal Logic*, 22(3), 2002.

[43] Donald H Berman and Carole D Hafner. Representing teleological structure in case-

based legal reasoning: the missing link. In *Proceedings of the 4th International Conference on Artificial Intelligence and Law*, pages 50–59. ACM, 1993.

[44] Dan Cartwright and Katie Atkinson. Using computational argumentation to support e-participation. *IEEE Intelligent Systems*, 24(5):42–52, 2009.

[45] Alison Chorley and Trevor Bench-Capon. Agatha: Using heuristic search to automate the construction of case law theories. *Artificial Intelligence and Law*, 13(1):9–51, 2005.

[46] Alison Chorley and Trevor Bench-Capon. An empirical investigation of reasoning with legal cases through theory construction and application. *Artificial Intelligence and Law*, 13(3-4):323–371, 2005.

[47] Eugenio Di Tullio and Floriana Grasso. A model for a motivational system grounded on value based abstract argumentation frameworks. In *International Conference on Electronic Healthcare*, pages 43–50. Springer, 2011.

[48] Paul E. Dunne. Tractability in value-based argumentation. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 195–206. IOS Press, 2010.

[49] Paul E. Dunne and Trevor J. M. Bench-Capon. Coherence in finite argument systems. *Artif. Intell.*, 141(1/2):187–203, 2002.

[50] Paul E. Dunne and Trevor J. M. Bench-Capon. Complexity in value-based argument systems. In José Júlio Alferes and João Alexandre Leite, editors, *Logics in Artificial Intelligence, 9th European Conference, JELIA 2004, Lisbon, Portugal, September 27-30, 2004, Proceedings*, volume 3229 of *Lecture Notes in Computer Science*, pages 360–371. Springer, 2004.

[51] Christoph Engel. Dictator games: a meta study. *Experimental Economics*, 14(4):583–610, 2011.

[52] Paul Finkelman. Fugitive baseballs and abandoned property: Who owns the home run ball. *Cardozo L. Rev.*, 23:1609, 2001.

[53] Robert Forsythe, Joel L Horowitz, Nathan E Savin, and Martin Sefton. Fairness in simple bargaining experiments. *Games and Economic behavior*, 6(3):347–369, 1994.

[54] Katie Greenwood, Trevor Bench-Capon, and Peter McBurney. Towards a computational account of persuasion in law. In *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, pages 22–31. ACM, 2003.

[55] Joseph Henrich, Robert Boyd, Samuel Bowles, Colin Camerer, Ernst Fehr, Herbert Gintis, and Richard McElreath. In search of homo economicus: behavioral experiments in 15 small-scale societies. *The American Economic Review*, 91(2):73–78, 2001.

[56] John Horty and Trevor Bench-Capon. A factor-based definition of precedential constraint. *Artificial Intelligence and Law*, 20(2):181–214, 2012.

[57] A.J.P. Kenny. Practical reasoning and rational appetite. In Joseph Raz, editor, *Practical Reasoning*, pages 63–80. Oxford University Press, 1978.

[58] Yanko Kirchev, Katie Atkinson, and Trevor J. M. Bench-Capon. Demonstrating the

distinctions between persuasion and deliberation dialogues. In Max Bramer and Miltos Petridis, editors, *Artificial Intelligence XXXVI - 39th SGAI International Conference on Artificial Intelligence, AI 2019, Cambridge, UK, December 17-19, 2019, Proceedings*, volume 11927 of *Lecture Notes in Computer Science*, pages 93–106. Springer, 2019.

[59] Loredana Laera, Ian Blacoe, Valentina Tamma, Terry Payne, Jerôme Euzenat, and Trevor Bench-Capon. Argumentation over ontology correspondences in mas. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pages 1–8, 2007.

[60] Lars Lindahl and Jan Odelstad. Normative positions within an algebraic approach to normative systems. *Journal of Applied Logic*, 2(1):63–91, 2004.

[61] Abraham Harold Maslow. A theory of human motivation. *Psychological review*, 50(4):370, 1943.

[62] John Stuart Mill. On the definition of political economy, and on the method of investigation proper to it. *Essays on some unsettled questions of Political Economy*, page 326, 1844.

[63] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173(9-10):901–934, 2009.

[64] Sanjay Modgil and Trevor Bench-Capon. Integrating object and meta-level value based argumentation. *Computational Models of Argument: Proceedings of COMMA 2008*, 172:240, 2008.

[65] Sanjay Modgil and Trevor Bench-Capon. Metalevel argumentation. *Journal of Logic and Computation*, 21(6):959–1003, 2011.

[66] Samer Nofal, Katie Atkinson, and Paul E. Dunne. Algorithms for decision problems in argument systems under preferred semantics. *Artificial Intelligence*, 207:23–51, 2014.

[67] Martin A Nowak, Karen M Page, and Karl Sigmund. Fairness versus reason in the ultimatum game. *Science*, 289(5485):1773–1775, 2000.

[68] Hessel Oosterbeek, Randolph Sloof, and Gijs Van De Kuilen. Cultural differences in ultimatum game experiments: Evidence from a meta-analysis. *Experimental Economics*, 7(2):171–188, 2004.

[69] Chaim Perelman. *Justice, law, and argument: Essays on moral and legal reasoning.* Reidal, Dordrecht, 1980.

[70] Chaim Perelman and Lucie Olbrechts-Tyteca. The new rhetoric: a treatise on argumentation, trans. *John Wilkinson and Purcell Weaver (Notre Dame, IN: University of Notre Dame Press, 1969)*, 190:411–12, 1969.

[71] Edwina L Rissland and Kevin D Ashley. A case-based system for Trade Secrets law. In *Proceedings of the 1st International Conference on Artificial Intelligence and Law*, pages 60–66. ACM, 1987.

[72] Shalom H Schwartz. Universals in the content and structure of values: Theoretical advances and empirical tests in 20 countries. *Advances in experimental social psychology*, 25(1):1–65, 1992.

[73] John R Searle. *Rationality in action.* MIT press, 2003.

[74] Cássia Trojahn, Paulo Quaresma, and Renata Vieira. An extended value-based argumentation framework for ontology mapping with confidence degrees. In *International Workshop on Argumentation in Multi-Agent Systems*, pages 132–144. Springer, 2007.

[75] Thomas L van der Weide, Frank Dignum, J-J Ch Meyer, Henry Prakken, and GAW Vreeswijk. Practical reasoning using values. In *International Workshop on Argumentation in Multi-Agent Systems*, pages 79–93. Springer, 2009.

[76] Douglas Walton. *Argumentation schemes for presumptive reasoning.* Lawrence Erlbaum Associates, 1996.

[77] Douglas Walton and Erik Krabbe. *Commitment in dialogue: Basic concepts of interpersonal reasoning.* SUNY press, 1995.

[78] Maya Wardeh, Adam Wyner, Katie Atkinson, and Trevor Bench-Capon. Argumentation based tools for policy-making. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law*, pages 249–250, 2013.

[79] M. Wooldridge and W. van der Hoek. On obligations and normative ability: Towards a logical analysis of the social contract. *Journal of Applied Logic*, 3:396–420, 2005.

[80] Adam Wyner, Trevor Bench-Capon, and Katie Atkinson. Arguments, values and baseballs: Representation of Popov v. Hayashi. In *Proceedings of JURIX 2007*, pages 151–160, 2007.

[81] Adam Wyner, Maya Wardeh, Trevor Bench-Capon, and Katie Atkinson. A model-based critique tool for policy deliberation. In *JURIX*, pages 167–176, 2012.

Received May 2021

Weighted Argumentation

Stefano Bistarelli, Francesco Santini
Department of Mathematics and Computer Science, University of Perugia, Italy
{stefano.bistarelli,francesco.santini}@unipg.it

Abstract

When dealing with Abstract Argumentation, having preference values on arguments/attacks clearly brings more information to a framework, which can be considered as a directed graph. One of the advantages is the possibility to define a different notion of defence, checking also if the associated preference is stronger than the preference of the considered attack. In the real-world, such values can be represented by "likes" in social-networks, or generic votes in favour of attacks. We focus on qualitative/quantitative preference values on attacks, which indicate their (relative) strength and can measure an argument-pair inconsistency degree. Once assembled, also by moving values from arguments to attacks, it is then possible to redefine semantics, relax the notion of weighted acceptability, and check well-known properties as in Dung's frameworks, e.g., if a framework is well-founded.

1 Introduction and Motivations

In the approach presented in this paper, we recognise that not all arguments or attacks in an *Abstract Argumentation Framework* (AF) [28] are equal in strength. As we will see in the following, considering these strength degrees under the form of "weights" (or values) brings a new perspective when searching for collections of arguments to be considered as collectively acceptable. The reason is that a weighted framework overcomes some natural limitations of the classical frameworks elaborated by P.M. Dung [28], where several extensions for a given semantics may be provided, but with nothing to distinguish between them. If a attacks b and vice-versa, both $\{a\}$ and $\{b\}$ are admissible solutions, but if the attack from a to b is stronger, then $\{b\}$ may not be considered as acceptable anymore. Consider the example where:

a: The price of this car is too much expensive for us: we cannot afford it.
b: This is the car I always dreamed of: we should buy it.

In this case, according to Dung's approach, both arguments are credulously accepted, neither is sceptically accepted, and the grounded extension is empty.[1] Hence, the classical analysis is not very useful, and the degree of uncertainty is high: no argument can be always selected without any doubt. However, from what arguments imply in the example above, we can probably derive that attacks are not equal in strength. Reality rules over dreams, and a rational agent cannot go in the direction of buying the desired car because of its price: $\{a\}$ is preferable than $\{b\}$, at least without any additional argument suggesting for a loan. If the attack from a to b is stronger than the inverse attack, then uncertainty can be reduced and $\{a\}$ can be "more acceptable" than $\{b\}$.

In its original formulation [28], a framework AF is represented by a pair $\langle Ar, att\rangle$ consisting of respectively a set of arguments and a binary relationship of attack defined among them. Given a framework, it is possible to examine the question on which set(s) of arguments can be accepted, hence collectively surviving the conflict defined by *att*. Answering this question corresponds to defining an Argumentation semantics. The key idea behind *extension-based* semantics is to identify some sets of arguments (called *extensions*) that survive the conflict "together".

In order to better frame the scope of this paper, we first clarify the meaning of weights (on attacks) for us. As pointed out in [30], in this paper we consider three possible interpretations: weights can be seen as *i)* the number of votes in favour of an attack, *ii)* as a measure of the inconsistency of argument-pairs, or *iii)* as rankings of different types of attack. Note that the first interpretation lays a link between Argumentation and Social Choice theory [30]. On the other hand, we will not here survey frameworks where weights represent something different from the above approaches, as for example probabilities: this is for example the topic of [36]. In practice, we here consider weights as a basic strength-value, which may represent various issues like votes provided by users [31], importance degree of a value it promotes [9], or trustworthiness of its source [27]. Therefore, in all such divergent cases, the basic strength may be expressed by a numerical value, leading to *Weighted Argumentation Frameworks*, or simply WAFs in the following of the paper.

The approaches we survey implement both quantitative (e.g., [30], [38], [26], and [15]) and qualitative approaches (e.g., [41], [22], and [15]).[2] Quantitative approaches

[1] The idea behind the grounded extension is to accept only those arguments that one cannot avoid to accept, to reject only the arguments that one cannot avoid to reject, and abstaining as much as possible. Hence, it represents the most sceptical (or "least committed") semantics among those based on complete extensions [6, Chapter 4].

[2] Note that by using a parametric algebraic structure (i.e., c-semirings, see Section 4), the work in [15] is capable or representing both the preference systems.

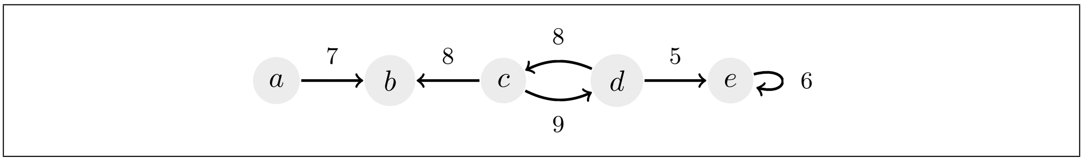


Figure 1: An example of WAF.

require to specify a numeric value for each attack, that is 0.7 or 9, for example. Qualitative frameworks express preferences via generic qualitative (usually partial) preference relations over attacks.

There is an established trend in the literature on the formalisation of Argumentation towards considering the strength of arguments/attacks: a summary of the bibliography is given also in Section 2. A shared motivation among some of these proposals is the observation that not all the arguments are equal, and that the relative strength of the arguments needs to be taken into account somehow. In this paper, we focus on weighted attacks: if there is an attack $att(a, b)$, then a relation $w(a, b) = s$ returns the weight (s) associated with that directed attack.[3] Hence the definition of an AF needs to include a further relation w. Other works (e.g., [39] and [5]) focus on preference values associated with arguments instead, and offer a different view on a strictly related problem.

The idea of explicitly adding weights to attacks, instead of arguments, was proposed by [7] for the first time. Considering a strength value on attacks allows us to consider a richer and more finer-grained model of frameworks than having weights on arguments. Richer because attacks are usually more than arguments in a framework, i.e., up to $|Ar|^2$ if a framework is a complete directed graph (or *digraph*) with self-attacks: consequently, weights are more than weights on arguments, and such a model can provide richer details. The weight-on-attacks model is also finer-grained, since it is possible to derive a finer definition of acceptability for an extension, which specifies a required level of defence of any argument in an extension: the strength of an argument impacts on all of its neighbours, while the strength on attacks only impacts on the adjacent corresponding argument [30].

Having motivated the use of Weighted Abstract Argumentation as an extension of the model designed by P.M. Dung, and the use of weights on attacks, Figure 1 reports an example of WAF as we intend it in this paper.

Outline of the Paper. This paper is structured as follows: Section 2 briefly surveys the related approaches. Most of these works mainly concern priority-based

[3] With the purpose to lighten the notation, we will use $w(a, b)$ in place of $w(att(a, b))$.

rules or values associated with arguments. Even if outside the immediate scope of this paper, these approaches need to be mentioned because they are strictly related to what presented in the following.

In Section 3 we present the main techniques in the literature to translate preference values on arguments into weights on attacks: since this paper is focused on the latter, in this way we suggest a possible bridge towards such two families of frameworks.

Section 4 presents the most important valued-structures where to draw weights from and perform operations. These systems are very specific to single proposals (e.g., weights are in $\mathbb{R}$), or they are more general and can be parametric, such as c-semirings [13].[4]

Section 5 introduces how novel definitions of acceptability can be derived when using weights. As advanced in Section 1, the presence of numerical values enriches the model conceived in [28] and allows for reducing uncertainty, simply because more information than just arguments and attacks is embedded in the model. An argument is now defended if the strength of the defence is stronger than the attack strength, by using different aggregation functions on the considered weights. This section also describes the relationships (e.g., implications) among different acceptability notions.

In Section 6 we show how weighted systems can be relaxed; the motivation behind it is dual. On one side, it is related to the notion of defence so that it becomes possible to weaken the condition that defence needs to be stronger than attack: accordingly, weighted defence can be reduced up to the notion of plain defence given by P.M. Dung, which in fact does not consider weights. On the other side, the conflict-freeness required by most semantics can be broken, and an amount of internal inconsistency can be tolerated.

Section 7 shows how classical extension-based semantics in Dung's Argumentation are revised according to *i)* the different notions of weighted acceptability presented in Section 5, *ii)* and also the relaxations in Section 6.

Section 8 summarises how the property of a framework to be well-founded [28] changes in presence of weighted systems.

Section 9 presents tools and real-world applications of weighted Argumentation. All these applications are related to information coming from online social and reviewing platforms, such as *Twitter.com* and *Amazon.com*.

Finally, Section 10 proposes possible future lines of research in the direction outlined in this paper, and it provides final thoughts and discussion.

[4] C-semirings have been used for the first time to associate a preference value with a *soft constraint*, and find the best solutions of *Soft Constraint Satisfaction Problems* [13].

2 Related Approaches

This section reports a summary of similar approaches in the literature that concern the use of preference values on arguments (instead of attacks) and the use of priority-based rules, by starting from the second approach.

There have been a number of proposals for extending Dung's framework in order to allow for more sophisticated modelling and analysis of conflicting information. A common theme among some of these proposals is the observation that not all arguments are equal, as we introduced in Section 1. Hence, the relative strength of arguments needs to be taken into account somehow. Such a preference/strength-/priority can be modelled in several ways, which we will inspect in the following.

A first well-studied use of preferences in the non-monotonic logic literature is based on the use of priority orderings over formulae in the language or defeasible inference rules. Such methods are usually proposed for structured approaches, instead of the abstract framework of Dung we investigate in this paper. The strength of arguments is inferred from the strength of the rules from which the arguments are constructed: in this case, priority orderings need to be "lifted" to preferences over arguments. There exist several proposals [45; 44; 29], and some well-known instantiations are represented for example by *ASPIC*$^+$ [43] and *Defeasible Logic Programming* [35], which comes with *strict* (high priority) and *defeasible* (low priority) rules.

In the literature it is possible to find many proposals where arguments (and not attacks, as in this paper) are associated with a value or preference. Two of the most well-known proposals are respectively given by [9] with *Value-based AF*s (*VAF*s), and [4] with *Preference-based AF*s (*PAF*s). A *VAF* is a five-tuple $\langle Ar, att, V, val, valpref \rangle$, where Ar is a finite set of arguments, att is an irreflexive binary relation on Ar (i.e., $\langle Ar, att \rangle$ is a standard AF), V is a non-empty set of values, val is a function which maps elements of Ar into elements of V, and $valpref$ is a preference relation (transitive, irreflexive and asymmetric) on $V \times V$. We say that an argument a relates to value v if accepting a promotes or defends v: the value in question is given by $val(a)$, for every $a \in Ar$, $val(a) \in V$. When a *VAF* is considered by a particular audience, the value ordering is fixed.

A *PAF* is a triplet $\langle Ar, att, Pref \rangle$ where $Pref$ is a partial pre-ordering (reflexive and transitive binary relation) on $Ar \times Ar$. The notion of defence changes accordingly: let a and b be two arguments, b attacks a if-and-only-if $att(b, a)$ and not $a > b$, i.e, a is not (qualitatively) preferred in the partial pre-ordering.

In [42] the author extends Dung's theory of Argumentation to integrate a meta-level Argumentation concerning preferences. Dung's level of abstraction is preserved, so that arguments expressing preferences are distinguished by being the source of a second attack relation. This abstractly characterises the application of preferences by attacking attacks between the arguments that are subject to preference claims. By proposing a meta-level, the work in [42] also concerns higher-order models [39].

A quantitative study is proposed in [40], where the authors define *Social Abstract Argumentation Frameworks*, which basically associate positive and negative votes to each argument. Afterwards, a semantics is essentially given by fix-points of a set of equations that assign, for each argument a, a value that is based on its social support and on how strong the attack a is being subjected to is. This framework has been extend in [31] by considering weights on attacks as well.

In [25] the authors survey the works in [4], [22], and [38], focusing on how to relate preference-values and weights, on either arguments or attacks (see Section 3).

One more recently-popular framework is represented by *ranking-based semantics*, whose aim is to elicit a preference score for each argument by considering the structure of a given AF. In case of non-weighted AFs, the scores are extracted by considering properties related to attacks with respect to each argument, as the number of attack/defence paths and their length [20].

Two other works where preference values are elicited from the framework are [10] and [24]. Furthermore, some works extract a preference from frameworks where arguments are already labelled with a strength score: in this case, such semantics consider both the original weights and the structure of an AF [3].

3 From Weights on Arguments to Weights on Attacks

As introduced in Section 1, this paper concerns weighted attacks. Others proposals in the literature deal with values assigned to arguments instead. For this reason, in this section we fill the distance with those works by showing how to pass from values on arguments to values on attacks.

A first possible view is to have numbers associated with arguments: we call a framework like this as WAF_{Ar}, in order to distinguish them from the WAFs in the rest of this paper, which have weights on attacks instead. A WAF_{Ar} can be described by a triple $\langle Ar, att, w\rangle$, where $w : Ar \to \mathbb{R}$: $w(a)$ is the value associated with an argument in a WAF_{Ar}. Such a framework can be straightforwardly encoded to a *PAF* following the rule: a is better/equal than b iff $w(a) \geq w(b)$ [25].

In [37] the authors define an *Argumentation Framework with Varied-strength defeat* (AFV) as a triplet $\langle Ar, att, Vdef\rangle$, where besides the set of arguments and the attack relationship, *Vdef* is a function from *att* to the interval $[0, 1]$. $Vdef(a, b)$ represents the certainty degree of the statement "a attacks b". The authors present how to translate WAF_{Ar}s to *PAF*s and AFVs. The intuition is that, the larger the preference of argument a over argument b, the stronger the attack from a to b. Below we report different alternative approaches (all quantitative) the authors propose for both schemes:

PAF

1. $\mathit{Vdef}(a, b) = 0$ if b is better than a.
2. $\mathit{Vdef}(a, b) = 0$ if b is better than a, 1 if a is better than b, 0 otherwise.

WAF$_{Ar}$

1. $\mathit{Vdef}(a, b) = 0$ if $w(b) > w(a)$.
2. $\mathit{Vdef}(a, b) = \max(w(a) - w(b), 0)$.
3. $\mathit{Vdef}(a, b) = 1 - \max(w(b) - w(a), 0)$.

Hence, most of the definitions for *Vdef* lead to the suppression of the attacks $att(a, b)$ for which b is strictly preferred to a.[5] With the third proposal above instead, an attack from a to b is removed when $w(b) - w(a) = 1$.

The work [25] elaborates on [37]. They consider quantitative as well as qualitative approaches for expressing these preferences. The goal is to translate PAFs and WAF_{Ar}s to *Argumentation Frameworks with attacks of Various Strength* (AF_{VS}), formally $\langle Ar, ATT, \overrightarrow{\succeq} \rangle$. Differently from [37], besides the sets of arguments Ar, ATT is a finite set of attack relations over Ar,[6] and $\overrightarrow{\succeq}$ is a binary relation over ATT: it expresses a relative strength between the different attack relations in ATT. From PAFs to AF_{VS}s, ATT and $\overrightarrow{\succeq}$ are assembled by respecting some principles considering a qualitative scheme:

P1 The initial set of attacks between arguments in PAF must not be modified (no attack appears/disappears).

P2 An attack between two equivalent arguments must be strictly stronger than an attack between two incomparable arguments.

P3 An attack from a to b with a strictly preferred to b must be strictly stronger than an attack between two equivalent or incomparable arguments.

P4 Consider an attack from a to c and an attack from b to c of the same class. If a is strictly stronger than b, the attack (a, c) must be strictly stronger than the attack (b, c).

P5 Consider an attack from a to c and an attack from a to b of the same class. If b is strictly stronger than c, the attack (a, c) must be strictly stronger than the attack (a, b).

[5] $\mathit{Vdef}(a, b) = 0$ is equivalent to $(a, b) \notin att$.

[6] We use ATT for a set of attack relations, and att for a single attack relation.

P2 and **P3** together induce the partitioning of *att* into four classes of attacks $\{\xrightarrow{1}, \xrightarrow{2}, \xrightarrow{3}, \xrightarrow{4}\}$, respectively depending on a better/equivalent/incomparable/-worse than b in PAF.[7]

The quantitative approach in [25] is instead focused on translating a generic WAF_{Ar} with values on arguments to an AF_{VS}. The authors assume a function $w : Ar \to [0,1]$, and a function $f : [0,1] \times [0,1] \to \mathbb{R}$: the strength of $att(a,b)$ is quantified by $f(w(a), w(b))$. As in the aforementioned qualitative scheme from PAFs, even in this case some principles impose conditions on the relationship between the weights on arguments and the strength value on derived attacks:

P4' If the weight of a is greater than the weight of b, then the higher the difference of the weights, the stronger the attack from a to b.

P5' If the weight of a is lower than the weight of b, then the higher the difference of the weights, the weaker the attack from a to b.

From **P1**, **P2**, **P4'**, and **P5'** it is possible to derive some conditions on a weighting translation function f:

Definition 3.1 (Weighting translation f). *A weighting translation function is a function $f : [0,1] \times [0,1] \to \mathbb{R}$ such that: $\forall x, y, z, t \in [0,1]$,*

- *if $x > y$ and $t > z$ then $f(x,y) > f(y,y) > f(z,t)$.*
- *$f(x,x) = f(y,y)$.*
- *if $x - y > z - t > 0$, then $f(x,y) > f(z,t)$.*
- *if $x - y < z - t < 0$, then $f(x,y) < f(z,t)$.*

An example of a such a function respecting the above constraints is $\forall x, y \in [0,1], f_{\alpha\beta}(x,y) = \alpha(x-y) + \beta$, with $\alpha > 0$ and $\beta > 0$; α amplifies the difference between x and y, while β represents a bias when the difference is 0. For example, if $att(a,b)$, $w(a) = 0.7$, $w(b) = 0.6$, and $f_{1\,1}(x,y) = (x-y)+1$, then the weight on this attack is 1.1.

To recap, Figure 2 represents how different mapping of weights can be translated one to another (edge direction means “from-to”), with respect to different proposals having a preference on either arguments (WAF_{Ar}, VAF, and PAF), or attacks (AFV, AF_{VS}). Note that a PAF can be translated to a class of VAFs [21], and for this reason such an encoding is not represented in Figure 2. Each VAF can be however translated into a unique PAF instead (grey edge in Figure 2).

[7] These are exactly the same four classes used in [41], as reported in Section 4.

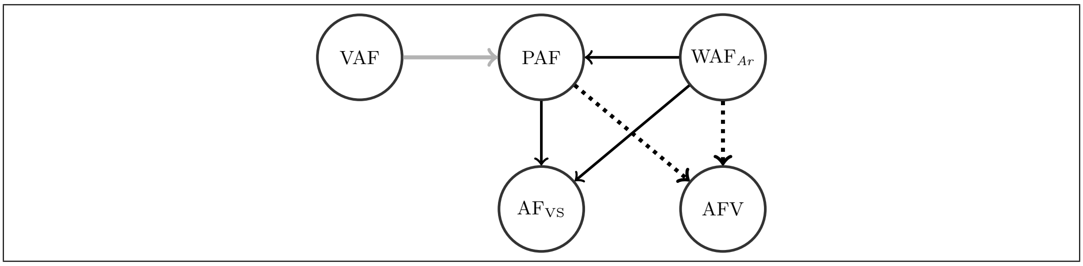


Figure 2: The edges in this graph describe how the different weighting systems proposed in this section (and *VAF* in Section 2) can be translated to others. Plain edges are explained in [25], dotted edges in [37]. The grey edge points to the fact that a unique *PAF* can associated with a *VAF* [21].

Note that in the following we will often refer to the work in [41], which is a variant of the AF_{VS} explained in this section. However, the translations to AF_{VS} described above and in [25] are more general, and can lead to more than just four classes of preference among attacks as proposed in [41] instead (see Section 5).[8]

4 Frameworks and Structures to Represent Weights

In the literature, a WAF is a classical AF equipped with a structure to represent weights and some operations to aggregate weights and compare them and prefer one or another. In the following of this section we show how different authors design such a framework of values.

In [30], a weighted argument system is a triple $\langle Ar, att, w\rangle$ where $\langle Ar, att\rangle$ is a Dung-style abstract argument system, and $w : att \longrightarrow \mathbb{R}_>$ is a function assigning real valued weights to attacks. $\mathbb{R}_>$ denotes the real numbers greater than 0, hence attacks are required to have a positive non-zero weight, since an attack could be discarded at no cost otherwise. The aggregation operator is the arithmetic sum, and values are composed in order to compute a relaxation threshold for disregarding attacks (see Section 6). The preference operator is simply $\leq$: for example, 4 is less strong than 5.

A WAF in [41] is a triplet $\langle Ar, ATT, R\rangle$ where Ar is a set of arguments, ATT is in general a set of n binary attack relations $\{\xrightarrow{1}, \xrightarrow{2}, \ldots, \xrightarrow{n}\}$ defined over Ar, and R is a binary relation defined over ATT. The relation $R \subseteq ATT \times ATT$

[8] For this reason, such translations are not supposed to maintain the defence notion and semantics from PAF or WAF to the work in [41].

denotes an order of strength between argument conflicts.[9] The paper proposes four classes of precedence, that is $\gg$ (and $\ll$), $\approx$, and ?: $att(a, b) \gg att(b, a)$ means that the former attack is stronger than the latter (vice-versa, a weaker attack). Equivalent and incomparable classes are considered as well, i.e, $att(a, b) \approx att(b, a)$ and $att(a, b)?att(b, a)$, respectively. This is accordingly reflected by the definition of defence, where considering $att(a, b)$ and $att(c, a)$ we can have that c is a *strong*, *weak*, *normal*, or an *unqualified* defender of b. Therefore, an argument b is defended by T if, and only if, for any argument a such that $att(a, b)$, there is an argument $c \in T$ such that $att(c, a)$, and according to the desired defence strength, $att(c, a) \gg att(a, b)$, $att(c, a) \ll att(a, b)$, $att(c, a) \approx att(a, b)$, and $att(c, a)?att(a, b)$. In such a framework there is an implicit aggregation operator: the attack strength corresponds to the strongest weight among all the counter-attacks. In order to show an example using the notation in [41], given $a \xrightarrow{1} b \xrightarrow{2} c$, a is a strong defender of c if we suppose $\xrightarrow{1} \gg \xrightarrow{2}$; the labels on attacks thus represent a class of attacks.

A WAF in [26] is a triple $\langle Ar, att, w\rangle$ where $\langle Ar, att\rangle$ is a Dung-style Abstract Argumentation Framework, and $w : Ar \times Ar \longrightarrow \mathbb{N}$ is a function assigning a natural number to each attack (i.e. $w(a, b) > 0$ iff $(a, b) \in att$, and a null value otherwise (i.e, $w(a, b) = 0$ iff $(a, b) \notin att$). With respect to [30], the authors of [26] state that in most situations natural numbers are enough, and this simplifies some of the definitions to come.

In [26] the authors define $\sigma^{\boxtimes}$-extensions, where σ is one of the given semantics (e.g., admissible), and $\boxtimes$ is an *aggregation function* of weights from $\mathbb{N}^n$ to $\mathbb{N}$, that is the set of natural numbers. To be valid, $\boxtimes$ needs to satisfy three properties: *i) non-decreasingness* (if $x_i \geq x'_i$, then $\boxtimes(x_1, \ldots, x_i, \ldots, x_n) \geq \boxtimes(x_1, \ldots, x'_i, \ldots, x_n)$), *ii) minimality* ($\boxtimes(x_1, \ldots, n_n) = 0$ if $\forall i,\ x_i = 0$), and *iii) identity* ($\boxtimes(x) = x$). In [26] the authors focus on $+$ and max to simplify the presentation, but several other aggregation functions can be considered as well (e.g., *leximin* or *leximax*). The preference operator, since $\mathbb{N}$ is always used, is simply given by $\leq$: 5 is stronger than 4.

Finally, some other works adopt an algebraic structure named *c-semiring* to represent weights, originally defined in [13]; they derive from algebraic semirings and are characterised by two operators as well, that is $\otimes$ and $\oplus$. In practice, c-semirings are *commutative* ($\otimes$ is commutative) and *idempotent* (i.e., $\oplus$ is idempotent) semirings, where $\oplus$ defines a complete lattice: every subset of elements have a *least upper bound*, or *lub*, and a *greatest lower bound*, or *glb*. In fact, c-semirings are semirings where $\oplus$ is used as a preference operator, while $\otimes$ is used to compose preference-values together.

[9] A WAF is in this case an $\mathrm{AF}_{\mathrm{VS}}$ as proposed in Section 3.

Definition 4.1 (C-semirings [13])**.** *A c-semiring is a tuple* $\mathbb{S} = \langle S, \oplus, \otimes, \bot, \top \rangle$ *such that* S *is a set,* $\top, \bot \in S$*, and* $\oplus, \otimes : S \times S \to S$ *are binary operators making the triples* $\langle S, \oplus, \bot \rangle$ *and* $\langle S, \otimes, \top \rangle$ *commutative monoids (semi-groups with identity), satisfying* i) $\forall s, t, u \in S, s \otimes (t \oplus u) = (s \otimes t) \oplus (s \otimes u)$ *(distributivity), and* ii) $\forall s \in S, s \otimes \bot = \bot$ *(annihilator). If* $\forall s, t \in S, s \oplus (s \otimes t) = s$*, the c-semiring is said to be absorptive. In short, c-semirings are commutative and absorptive semirings.*

The idempotency of $\oplus$ leads to the definition of a partial ordering $\leq_{\mathbb{S}}$ over the set S (S is a poset). Such partial order is defined as $s \leq_{\mathbb{S}} t$ if and only if $s \oplus t = t$, and $\oplus$ returns the *lub* of s and t (defined also as $\sqcup$, while the *glb* is defined by $\sqcap$). This means that t is "better" than s.

Some more properties can be derived on c-semirings [13]: *i)* both $\oplus$ and $\otimes$ are monotone over $\leq_{\mathbb{S}}$, *ii)* $\otimes$ is intensive (i.e., $s \otimes t \leq_{\mathbb{S}} s$), and *iii)* $\langle S, \leq_{\mathbb{S}} \rangle$ is a complete lattice. $\bot$ and $\top$ respectively are the bottom and top elements of such a lattice. When also $\otimes$ is idempotent, *i)* $\oplus$ distributes over $\otimes$, *ii)* $\otimes$ returns the *glb* of two values in S, and *iii)* $\langle S, \leq_{\mathbb{S}} \rangle$ is a distributive lattice.

Well-known instances of c-semirings are: $\mathbb{S}_{boolean} = \langle \{false, true\}, \vee, \wedge, false, true \rangle$, $\mathbb{S}_{fuzzy} = \langle [0,1], \max, \min, 0,\ 1 \rangle$, $\mathbb{S}_{bottleneck} = \langle \mathbb{R}^+ \cup \{+\infty\}, \max, \min,\ 0, \infty \rangle$, $\mathbb{S}_{probabilistic} = \langle [0,1],\ \max, \times, 0, 1 \rangle$ (also called the *Viterbi* semiring), $\mathbb{S}_{weighted} = \langle \mathbb{R}^+ \cup \{+\infty\}, min, +, +\infty, 0 \rangle$.[10] Note that c-semiring can also deal with non-numeric preference values: for instance, using a c-semiring set $S = \{bad, fair, good\}$, a total ordering as $bad \leq_{\mathbb{S}} fair \leq_{\mathbb{S}} good$, and an aggregation operator for which $bad \otimes fair = bad$.

A c-semiring based WAF, i.e., a $WAF_{\mathbb{S}}$, is a quadruple $\langle Ar, att, w, \mathbb{S} \rangle$, where $\mathbb{S}$ is a c-semiring $\langle S, \oplus, \otimes, \bot, \top \rangle$, Ar is a set of arguments, att the attack binary-relation on Ar, and $w : Ar \times Ar \longrightarrow S$ is a binary function: given $a, b \in Ar$ and $att(a, b)$, then $w(a, b) = s$ means that a attacks b with a weight $s \in S$. Moreover, it is required that $att(a, b)$ iff $w(a, b) <_{\mathbb{S}} \top$. Note that the Boolean c-semiring can be used to model classical Dung's Argumentation [15].

Differently from all the other proposals in this section, a c-semiring is parametric with respect to both the aggregation operator (i.e., $\otimes$) and the preference operator (i.e., $\oplus$), and it is not bound to a single set of values as $\mathbb{R}$ or $\mathbb{N}$. One more advantage is that a Cartesian product of c-semirings (which is still a c-semiring) can model multi-criteria weights, and in general, c-semirings can model partially-ordered preference values besides to totally-ordered ones.

[10] Note that when considering the Weighted c-semiring, it happens that $7 \leq_{\mathbb{S}} 3$ even if $7 \geq 3$, i.e., lesser means better.

5 Weighted Acceptability

A *weighted acceptability* notion extends the original one given in [28] by considering the two strength levels of attack and defence. For example, only if the defence weight is stronger, then an argument is successfully defended. For this reason, the notion of defence becomes more constrained than the unweighted one, which only considers edges in a digraph. We start by recalling the notion *acceptability* of an argument b in [28], with respect to a set of arguments T.

Definition 5.1 ($\mathbb{D}_{Dung}$). *An argument b is acceptable w.r.t. $T \subseteq Ar$ (or T defends b) iff for any argument $a \in Ar$ s.t. $att(a,b)$, then $\exists c \in T$ s.t. $att(c,a)$.*

Three different definitions of weighted defence are presented in [41], [26], and [15]. In the following, we condense their main features and we show how they differ.

We start by presenting the notion of acceptability given in [41]. When requiring a preference level $[\gg, \approx]$ (see Section 4), for each attacker a of b there must be either a strong or a normal defender $c \in T$. In Definition 5.2 we report the defence in [41] by using $[\gg, \approx]$.

Definition 5.2 ($\mathbb{D}_{Martínez\ et\ al.}$). *Given a WAF $\langle Ar, ATT, R\rangle$ as formalised in [41], with $R = [\gg, \approx]$, $a, b, c \in Ar$, $T \subseteq Ar$, then b is acceptable w.r.t. T iff $\forall att(a,b), \exists c \in T$ s.t. $att(c,a) \gg att(a,b)$ or $att(c,a) \approx att(a,b)$.*

In practice, only one argument $c \in T$ counter-attacking a with an equal or stronger level is needed to defend an argument b. This defence is typical of *Argumentation Frameworks with attacks of Various Strength*, i.e, the AF_{VS} (see Section 3). A different approach along the same line is [22]. Hence the notions of defence of these two works are strictly connected.

Note also that the intuition for extending the notion of defence is the same in both AF_{VS}s and AFVs (Section 3 and [38]). The only difference is due to the preference relation over attacks: an AF_{VS} uses a pre-ordering (thus allowing for incomparable attacks), whereas in an AFV the preference relation is based on a function with values on a linearly ordered scale, e.g., the interval $[0, ..1]$, providing a total ordering over attacks.

On the contrary, the idea in [26] is to aggregate all the weights of counter-attacks, and to check if they are stronger than the considered attack:

Definition 5.3 ($\mathbb{D}_{Coste-Marquis\ et\ al.}$). *Given a WAF $\langle Ar, att, w\rangle$ as defined in [26], an argument b is acceptable w.r.t. a subset of arguments T iff $\forall a \in Ar$ s.t. $att(a,b)$, we have that $w(T,a) \geq w(a,b)$, where $w(T,a)$ is a shortcut for $\boxtimes_{c \in T} w(c,a)$.*

Thus, an argument b is acceptable if for each attack from $a \in Ar$ against b, the aggregated weight of the collective defence of b is greater than $w(a, b)$.[11]

Finally, we report the definition of acceptability in [15], parametrically given for a c-semiring $\mathbb{S} = \langle S, \oplus, \otimes, \bot, \top \rangle$:

Definition 5.4 ($\mathbb{D}_{\textit{Bistarelli et al.}}$). *Given a $WAF_{\mathbb{S}}$ $\langle Ar, att, w, \mathbb{S} \rangle$ as defined in the work [15], $T \subseteq Ar$ defends $b \in Ar$ (or b is w-acceptable) iff $\forall a \in Ar$ such that $att(a, b)$, we have that $w(a, T \cup \{b\}) \geq_{\mathbb{S}} w(T \cup \{b\}, a)$, where $w(a, T \cup \{b\})$ is a shortcut for $\otimes_{c \in (T \cup \{b\})}(a, c)$ and $w(T \cup \{b\}, a)$ is a shortcut for $\otimes_{d \in T \cup \{b\}}(d, a)$.*

Besides aggregating the weights of all the counter-attacks as in [26], Definition 5.4 also aggregates the weights of all the attacks from a to $d \cup \{b\}$: in this case, all the attacks from any argument in Ar towards a set T plus the argument to be accepted need to be considered.

In [15] the authors use c-semirings to represent the other two proposals reported in this section, that is [41] and [26], as $WAF_{\mathbb{S}}$. This allows for discovering the relationships among such defences, as described in Section 5, and to easily show an example on how they differ, as highlighted in Figure 3. All three of them aggregate weights (in different ways) towards the same argument d to check if $\{e, f\}$ defends c from it: that is, c is defended if $w(\{e, f\}, d) \leq_{\mathbb{S}} w(d, \{c\})$. Only $\mathbb{D}_{\textit{Bistarelli et al.}}$ also aggregates all the weights on the attacks from the same attacker to the set of arguments to be defended: to check if d defends $\{a, b\}$ from c, we need that $w(\{d\}, c) \leq_{\mathbb{S}} w(c, \{a, b\})$.

Properties of Defence. By comparing all the notions of acceptability in the same c-semiring based framework [15], it is possible to catch their relationships as, for example, the implications among them:

- $\mathbb{D}_{\textit{Martínez et al.}}, \mathbb{D}_{\textit{Coste-Marquis et al.}}, \mathbb{D}_{\textit{Bistarelli et al.}} \Rightarrow \mathbb{D}_{\textit{Dung}}$
- $\mathbb{D}_{\textit{Bistarelli et al.}} \Rightarrow \mathbb{D}_{\textit{Coste-Marquis et al.}}$
- $\mathbb{D}_{\textit{Martínez et al.}} \Rightarrow \mathbb{D}_{\textit{Coste-Marquis et al.}}$

Moreover, if we replace the original structure to represent weights in [26] and [41] with different c-semirings, we obtain further interesting relationships. We recall from Section 4 that $\langle [0, 1], \max, \min, 0, 1 \rangle$ is the *Fuzzy* c-semiring and $\langle \{\textit{true}, \textit{false}\}, \vee, \wedge, false, true \rangle$ is the *Boolean* c-semiring:

- If $\mathbb{S} = \langle [0, 1], \max, \min, 0, 1 \rangle$, then $\mathbb{D}_{\textit{Martínez et al.}} \Leftrightarrow \mathbb{D}_{\textit{Coste-Marquis et al.}}$

[11] Such a phrasing of defence is also equivalent to the work in [16].

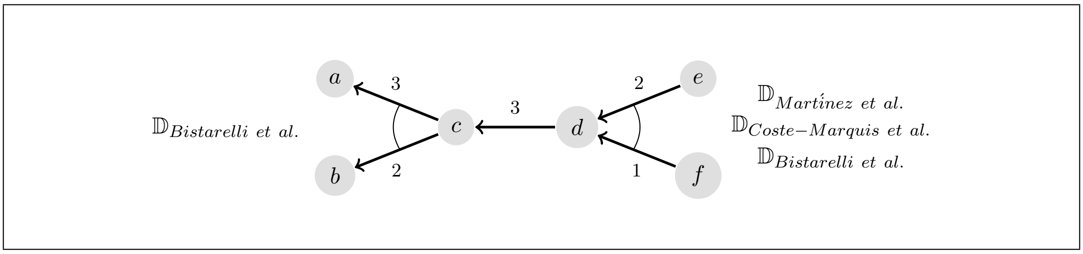

$\mathbb{D}_{Bistarelli\ et\ al.}$
a
b
c
d
e
f
3
2
3
2
1
$\mathbb{D}_{Mart\acute{i}nez\ et\ al.}$
$\mathbb{D}_{Coste-Marquis\ et\ al.}$
$\mathbb{D}_{Bistarelli\ et\ al.}$

Figure 3: The three notions of defence on the right of this figure aggregate attack weights to check if c is defended from d, while only $\mathbb{D}_{Bistarelli\ et\ al.}$ also aggregates all the weights on the attacks from the same attacker to a set of arguments to be defended (i.e., in this case $\{a, c\}$). Using the Weighted c-semiring, $\{e, f\}$ defends c from d according to $\mathbb{D}_{Coste-Marquis\ et\ al.}$ and $\mathbb{D}_{Bistarelli\ et\ al.}$, since $(2+1) \leq_{Weighted} 3$, but not according to $\mathbb{D}_{Mart\acute{i}nez\ et\ al.}$, since $2 \not\leq_{Weighted} 3$ and $1 \not\leq_{Weighted} 3$. Moreover, d defends $\{a, b\}$ from c according to $\mathbb{D}_{Mart\acute{i}nez\ et\ al.}$ and $\mathbb{D}_{Coste-Marquis\ et\ al.}$, but not according to $\mathbb{D}_{Bistarelli\ et\ al.}$ since $3 \not\leq_{Weighted} (3+2)$.

- If $\mathbb{S} = \langle [0,1], \max, \min, 0, 1 \rangle$, then $\mathbb{D}_{Bistarelli\ et\ al.} \Rightarrow \mathbb{D}_{Mart\acute{i}nez\ et\ al.}$
- If $\mathbb{S} = \langle \{true, false\}, \vee, \wedge, false, true \rangle$, then $\mathbb{D}_{Dung} \Leftrightarrow \mathbb{D}_{Bistarelli\ et\ al.} \Leftrightarrow \mathbb{D}_{Mart\acute{i}nez\ et\ al.} \Leftrightarrow \mathbb{D}_{Coste-Marquis\ et\ al.}$.

The last item states that, when dropping weights, the way these defences combine attacks and counter-attacks is irrelevant: they all flatten to the classical acceptability in [28].

6 Relaxations

The two main works that deal with internal relaxations are [30] and [15]. In both of them, the goal is to relax constraints represented by attacks: by tolerating some of them, it is possible to allow some inconsistency level with respect to the original considered framework, where all the attacks are always considered instead.

The main motivation is to obtain progressively more solutions, as one increases the inconsistency level to be tolerated. In this way, it is possible to return non-trivial solutions (e.g., stable extensions) in case conventional (unweighted) AFs have none. Clearly, the cost of such a relaxation represents a preference ordering over the obtained sets of arguments: indeed it is better to prefer extensions that are found with a milder relaxation than others, i.e., extensions on AFs that are closer to the original one.

The key idea in [30] is to consider an *inconsistency budget*, $\beta \in \mathbb{R}_{\geq}$, which is used to characterise how much inconsistency one is prepared to tolerate. The intended interpretation is that, given an inconsistency budget β, it is possible to disregard attacks up to a total weight of β. By doing so, one obtains several AFs on which to compute the desired semantics (or other typical problems in Argumentation). Classical AFs implicitly assume an inconsistency budget of 0, since it is not possible to disregard any attack. By allowing larger inconsistency budgets and consequently dropping more attacks, one can obtain progressively more frameworks and then solutions. The proposed disregard approach applies to all the classical semantics, hence obtaining e.g., β-admissible or β-complete extensions. For instance, the set of β-complete extensions is given by the union of all β-complete extensions on all the AFs obtained by disregarding attacks up to a total of β.

However, the main goal in [30] is to find alternatives when the single most sceptical (or least committed) semantics among all, i.e., the grounded one, returns an empty-set solution: different approaches to relax scepticism correspond to the design of different (unweighted) semantics, as the ideal and eager ones [6, Ch. 4].

Note that the work in [26] is inspired by the same relaxation as in [30]. Extensions are named as $\sigma^{\beta}_{\boxtimes}$-extensions: β is exactly the same inconsistency budget, but instead of only arithmetic sum, the aggregation of weights is via a parametric operator $\boxtimes$, as explained in Section 5.

In the second relaxation approach that we consider, i.e., [15], given a $WAF_{\mathbb{S}} = \langle Ar, att, w, \mathbb{S}\rangle$, a subset of arguments $T \subseteq Ar$ is α-conflict-free iff $w(T, T) \geq_{\mathbb{S}} \alpha$, where

$$w(T, T) = \bigotimes_{a \in T, b \in T} w(a, b)$$

means that we aggregate (using $\otimes$) all the weights associated with attacks in T. This approach is different from [30]: only the original framework is considered, hence no further AF or WAF is derived by disregarding attacks. The relaxation is obtained by tolerating an amount of attacks by breaking the conflict-free condition: an extension may contain attacks up to a threshold of α on the aggregation (i.e., $\otimes$) of their weights.

Figure 4 reports an example of WAF taken from [30]. Table 1 shows β-preferred and α-preferred extensions (using the Weighted c-semiring) on that WAF. On this example, both approaches return the same extensions when using the same α/β.

Table 2 shows the difference between β-grounded and α-grounded extensions. The approach in [30] aims to find several results, while the approach in [15] is more adherent to the original proposal in [28], and always returns a single grounded extension (which in this specific example is always the empty-set).

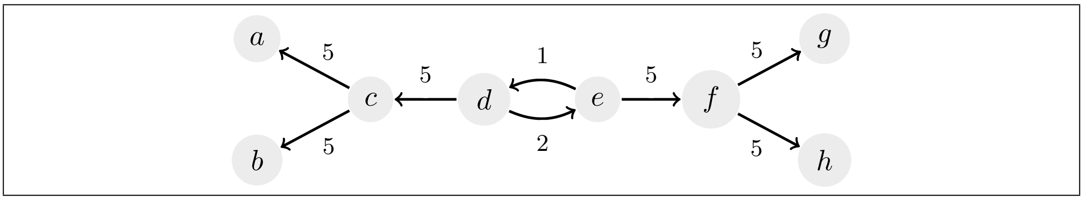


Figure 4: An example of weighted framework taken from [30].

α/β	$\mathcal{PR}$ in [30]	$\mathcal{PR}$ in [15]
0	$\{\{a,b,d,f\},\{c,e,g,h\}\}$	$\{\{a,b,d,f\},\{c,e,g,h\}\}$
1	$\{\{a,b,d,f\},\{c,e,g,h\}\}$	$\{\{a,b,d,f\},\{c,e,g,h\}\}$
2	$\{\{a,b,d,f\},\{c,e,g,h\}\}$	$\{\{a,b,d,f\},\{c,e,g,h\}\}$
3	$\{\{a,b,d,f\},\{c,e,g,h\},\{a,b,d,e,g,h\}\}$	$\{\{a,b,d,f\},\{c,e,g,h\},\{a,b,d,e,g,h\}\}$

Table 1: Considering the WAF in Figure 4, the sets of β-preferred and α-preferred extensions correspond while increasing α/β.

Relaxing Defence. The two works in [30] and [15] can be used to drop attacks in a given framework. All the weighted acceptability notions presented in Section 5 aggregate weights to understand if a defence is stronger than an attack: in general, if the defence strength is higher, then a set of arguments is effectively defended, otherwise the attack is predominant. In this section, we survey two methods that relax the concept of defence by tolerating arguments that are defended at a "milder" level.

A first approach can be found in [41]. From this work, the attack scenario $[T \subseteq \mathit{arguments}, \mathcal{P} = \{\gg\}]$ includes only strongly-defended arguments, since the defence condition is $\gg$. A scenario with defence condition $\mathcal{P}$ can be expanded into another scenario using a different defence condition $\mathcal{Q}$, through the concept of *defence upgrade* and an expansion operator $\uplus$.

Definition 6.1 (Defence upgrade)**.** *Let* $p = [T, \mathcal{P}]$ *be an attack scenario, and let*

α/β	$\mathcal{GR}$ in [30]	$\mathcal{GR}$ in [15]
0	$\{\emptyset\}$	$\emptyset$
1	$\{\emptyset,\{c,e,g,h\}\}$	$\emptyset$
2	$\{\emptyset,\{c,e,g,h\},\{a,b,d,f\}\}$	$\emptyset$
3	$\{\emptyset,\{c,e,g,h\},\{a,b,d,f\},\{a,b,d,e,g,h\}$	$\emptyset$

Table 2: Considering the WAF in Figure 4, β-grounded and α-grounded extensions differ on the same α/β.

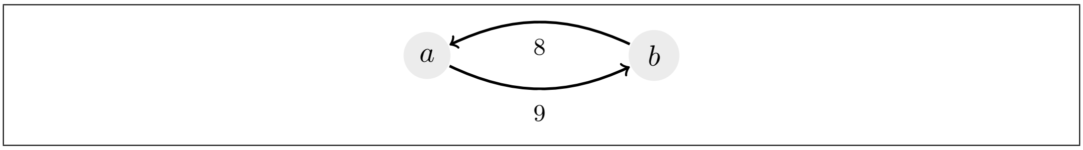


Figure 5: a defends itself according to $\mathbb{D}_{Bistarelli\ et\ al.}$ because the attack weight (9) is stronger than the defence weight (8), but for the same reason b only 1-defends itself: the defence is one unit less than what is really needed for a proper weighted defence (i.e., $9 - 8$), that is considering w-defence in Section 5.

$\mathcal{Q}$ be a set of defence conditions. The expansion of p according to $\mathcal{Q}$ is defined as $p \uplus \mathcal{Q} = [T' \cup T, \mathcal{P} \cup \mathcal{Q}]$, where $T' = \{a \in Ar \mid a$ is acceptable with respect to $[T, Q]\}$.

Hence, if $\mathcal{Q}$ is $[\gg, \approx]$, new arguments can be accepted in T', thus extending the set of accepted arguments $T' \cup T$.

The second approach is in [15]. There, the notion of weighted defence can be relaxed in order to be equal to[26] (see Section 5), and also to the classical defence given by [28]: this leads to completely ignoring weights. The γ-defence (or also $\mathbb{D}_\gamma$) in [15] is parametrised on a given c-semiring (see Section 4) and on a threshold-value γ: such a γ is used to consider arguments that are not "fully" defended according to $\mathbb{D}_{Bistarelli\ et\ al.}$, i.e., for which $w(a, T \cup \{b\}) \not\geq_{\mathbb{S}} w(T, a)$:

Definition 6.2 (γ-defence)**.** *Given $\langle Ar, att, w, \mathbb{S} = \langle S, \oplus, \otimes, \bot, \top \rangle\rangle$ and $\gamma \in S$, $T \subseteq Ar$ γ-defends $b \in Ar$ iff $\forall a \in Ar$ such that $att(a, b)$ we have that $w(T, a) \neq \top$ and*

$$\Big(w(a, T \cup \{b\}) \oslash w(T, a)\Big) \geq_{\mathbb{S}} \gamma$$

Definition 6.2 states that T defends b from a if the difference between the aggregation of the attack weights from a to T (union b) and the one from T to a is better than γ. The $\oslash$ operator is the inverse of $\otimes$ (e.g., the arithmetic $-$ in case $\otimes$ is the arithmetic plus). A simple example is given in Figure 5: a defends itself, while b only 1-defends itself, since $9 - 8 \geq_{\mathbb{S}} 1$, if we consider $\mathbb{S}$ as the Weighted c-semiring.

Clearly, by progressively increasing γ we consequently relax the constraint that defence needs to be equal or stronger than attack, and we obtain more and more extensions. This motivation is a leitmotif of the whole section.

Note that, if γ is chosen as the $\bot$ of the considered c-semiring, γ-defence is equivalent to the original definition of defence given by P.M. Dung [15]: $\mathbb{D}_\gamma \Leftrightarrow \mathbb{D}_{Dung}$. On the other hand, if $\gamma = \top$, then we have $\mathbb{D}_\gamma \Leftrightarrow \mathbb{D}_{Bistarelli\ et\ al.}$.

When all $a \in Ar$ attack one argument only, γ-defence is equivalent to the defence defined in [26], that is $\mathbb{D}_{Coste-Marquis\ et\ al.} \Leftrightarrow \mathbb{D}_\gamma$. Finally, by properly choosing γ, it is always possible to obtain a defence that is implied by the one in [26]: $\mathbb{D}_{Coste-Marquis\ et\ al.} \Rightarrow \mathbb{D}_{\gamma}$.[12]

7 Semantics in WAFs

Clearly, the concepts of weighted defence and relaxation advanced in Section 5 and Section 6 are not stand-alone, but they are the basis to elaborate on classical extension-based semantics [28].

As introduced in Section 5, in [30] the authors mainly focus on the β-grounded semantics, with the purpose to find alternatives when the unweighted one ([28]) is equal to empty-set, which is not very informative. Nevertheless, also the other β-semantics are briefly presented: they correspond to the union of the extension sets found on all the AFs obtained by disregarding an attack amount up to β.[13]

In [41], only the preferred and grounded semantics (or better, *scenarios*) are explicitly defined; they can be both captured by the use of the expansion operator $\uplus$. For example, Let $p = [T, \mathcal{P}]$ be an admissible scenario. If $p \uplus \mathcal{Q} = [T, \mathcal{P} \cup \mathcal{Q}]$ for any condition $\mathcal{Q}$, then T is a (classic) preferred extension: $T \subseteq Ar$ cannot be expanded. The grounded scenario is instead defined as the least fix-point of $\uplus$ using a defence condition $\mathcal{P}$.

The semantics defined in [26] follow two distinct paths, whose definitions we report below from the original paper. Definition 7.1 is used to consider an inconsistency budget β exactly as in [30]; in this case, however, $\boxtimes$ is a more general operator to aggregate weights to find β than the arithmetic sum adopted in [30].

Definition 7.1 ($\sigma^{\beta}_{\boxtimes}$-extensions). *Given a $WAF = \langle Ar, att, w\rangle$, a semantics σ, an aggregation function $\boxtimes$, and a budget β, the set of $\sigma^{\beta}_{\boxtimes}$-extensions, denoted as $\mathcal{E}^{\boxtimes,\beta}_{\sigma}(\langle Ar, att, w\rangle)$, is defined as $\mathcal{E}^{\boxtimes,\beta}_{\sigma}(\langle Ar, att, w\rangle) = \{E \in \mathcal{E}_{\sigma}(Ar, att \setminus att') \mid att' \in Sub(att, w, \beta)\}$, where the function $Sub(att, w, \beta)$ returns the set of subsets of att whose total aggregated weight does not exceed β.*

The $\sigma^{\beta}_{\boxtimes}$-grounded semantics may return several extensions, as in [30]. Afterwards, the same authors propose how to refine them by removing the empty-set from such a set of extensions, with the purpose to not trivialise the sceptical acceptance of arguments: the presence of an empty-set among the results impedes an argument to be sceptically accepted.

[12] We remind that $\mathbb{D}_{\gamma}$ is a relaxation of $\mathbb{D}_{Bistarelli\ et\ al.}$, which on the contrary implies the notion of defence in [26] (see Section 5).

[13] An example of β-preferred extensions is reported in Table 1.

The second path in [26] completely drops the β-budget and only focuses on defining semantics that use a weighted notion of acceptability based on $\boxtimes$, that is $\boxtimes$-acceptability. Semantics are simply named as $\boxtimes$-*semantics*, which are straightforwardly derived from their counterpart in [28]: for example, T is a $\boxtimes$-complete extension iff T is conflict-free and every argument $a \in Ar$, which is $\boxtimes$-acceptable with respect to T, belongs to T. In this case, the $\boxtimes$-grounded semantics returns a single extension.

The approach in [15] is the only one that allows for contemporarily disregarding an "amount" of attacks (also accomplished in [30] and [26]) *and* requiring defence to be stronger than an attack (also proposed in [26] and [41]). Moreover, it also permits to relax this latter constraint on defence, up to not considering weights at all.

Therefore, semantics are equipped with two thresholds: α^{γ}-semantics come with an internal inconsistency budget α (see Section 5), and a threshold γ on the defence relaxation (see Section 6).

Their definition is summarised in the following, given a $WAF_{\mathbb{S}} = \langle Ar, att, w, \mathbb{S} = \langle S, \oplus, \otimes, \bot, \top \rangle \rangle$ and $\alpha, \gamma \in S$.

- A subset of arguments $T \subseteq Ar$ is α-conflict-free iff $w(T, T) \geq_{\mathbb{S}} \alpha$. Note that γ is not considered in this semantics, since only the conflict internal to an extension is measured.

- An α-conflict-free set $T \subseteq Ar$ is α^{γ}-admissible iff the arguments in T are γ-defended by T from the arguments in $Ar \backslash T$.

- An α^{γ}-admissible $T \subseteq Ar$ is α^{γ}-complete iff each argument $b \in Ar$ that is γ-defended by T and s.t. $w(T \cup \{b\}, T \cup \{b\}) \geq_{\mathbb{S}} \alpha$ is in T (i.e., $b \in T$).

- An α^{γ}-preferred extension is a maximal (with respect to set inclusion) α^{γ}-admissible subset of Ar.

- An α^{γ}-admissible set T is also an α^{γ}-stable extension iff $\forall a \notin T, \exists b \in T$ then $w(b, a) \neq \top$, and $T \cup \{a\}$ is not α^{γ}-admissible.

Both approaches in [15] and [26], that is α^{γ}-semantics and $\boxtimes$-*semantics* respectively, preserve some of the formal properties of the unweighted semantics in [28]: for example, the implications *stable* ⇒ *preferred* ⇒ *complete* ⇒ *admissible* ⇒ *conflict-free*.

8 Well-founded WAFs

The goal of this section is to show how the well-foundedness property of frameworks [28] extend to WAFs. We commence by revising the notion of well-foundedness, then we show how it can be adapted to WAFs and how also well-founded WAFs have a single complete/preferred/stable/grounded extension. In this case, the considered framework always provides the same single solution in any these scenarios, consequently eliminating all the possible uncertainty. Since in [15] the authors showed how acceptability in [41] and [26] can be represented as a $WAF_{\mathbb{S}}$, we will consider such frameworks in the following of this section in order to propose a single and unitary point of view.

P.M. Dung defines the *sufficient* conditions behind well-foundedness in AFs in his pioneering work [28]. A well-founded AF is an AF without an infinite defeating sequence of arguments.

Definition 8.1 (Well-founded$_{Dung}$ AFs)**.** *An AF is well-founded iff there exists no infinite sequence $a_1, a_2, \ldots, a_n, \ldots$ (with $a_i \in Ar$) such that for each i, $att(a_{i+1}, a_i)$.*

In case of a finite number of arguments, a framework is well-founded if it is acyclic. However, the notions of weighted defence presented in Section 5 consider sets of arguments to check the aggregation strength of attacks and counter-attacks: Definition 8.1 is not enough anymore to capture the aggregation of weights from/to sets, since it is based on plain sequences of arguments.

Because of these reasons, in Definition 8.2 we redefine the notion of sequence of arguments into a sequence of sets, or better, *set-maximal attack* (SMA) sets. All the remainder of this section is inspired by the work in [19], definitions and theorems included.

Definition 8.2 (Set-maximal attack (SMA) sets)**.** *Given $WAF_{\mathbb{S}} = \langle Ar,\ att, w, \mathbb{S}\rangle$ and $T, U \subseteq Ar$, then T is a set-maximal attack on U, iff*

i) T is conflict-free;

ii) $\forall b \in T, \exists c \in U$ s.t. $att(b, c)$;

iii) there exists no T' s.t. conditions i) *and* ii) *hold and $T \subsetneq T'$.*

Example 8.3. *Figure 6 (right) shows a fragment of an infinite sequence of SMA sets, obtained on the WAF represented in Figure 6 (left); it starts from set $\{f\}$. The sequence is: T_1-T_2-T_3-T_4-... (it continues as T_2-T_3-T_4 for an infinite number of times).*

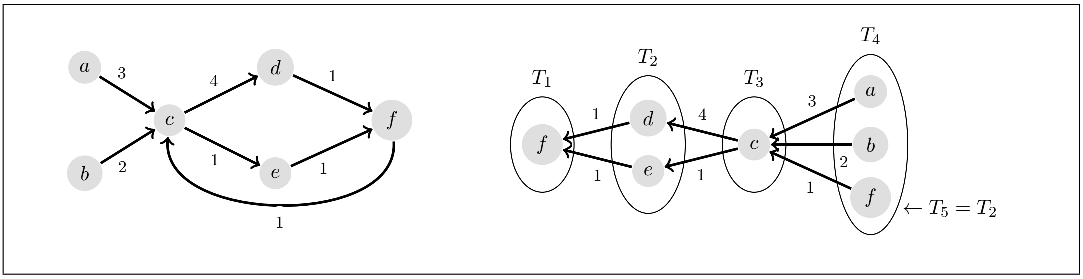


Figure 6: Left: an example of WAF; right: a fragment of an infinite sequence of SMA sets on the framework on the left. T_5 is identical to T_2 and the sequence infinitely continues from it.

Definition 8.4 generalises the well-foundedness property on different notions of (weighted) defence: for example, the ones in Section 5. Therefore, well-foundedness becomes parametric with respect to the chosen c-semiring *and* the selected defence. Other weighted defences may directly inherit from the definition to check the conditions under which they allow for a well-founded framework.

Definition 8.4 (Generalisation in $WAF_{\mathbb{S}}$). *Given a $WAF_{\mathbb{S}} = \langle Ar,\ att, w, \mathbb{S}\rangle$, if there does not exist an infinite sequence ω of SMA sets $T_1, T_2, \ldots$, such that for every T_{i+1}, T_i, and T_{i-1} we have that T_{i+1} defends T_{i-1} from each $a \in T_i$ according to a generic weighted defence $\mathbb{D}$, then $WAF_{\mathbb{S}}$ is well-founded w.r.t. $\mathbb{D}$.*

Hence, besides looking at infinite cycles as in [28] (in this case of sets, and not just of arguments), in WAFs there is one more constraint: to be infinite, this chain of SMA sets needs to respect the constraint imposed by weighted defences: T_{i+1} has to defend T_{i-1} from each $a \in T_i$. Therefore, a framework has less chances to be well-founded when considering weighted defences, since the absence of infinite sequences of SMA sets is not sufficient.

Example 8.5. *Still supposing the WAF in Figure 6 (left), we see that by considering it as a classical framework (dropping weights), the framework is not well-founded$_{Dung}$ because there exists an infinite sequence of arguments given by the cycle $f \leftarrow e \leftarrow c \leftarrow f \leftarrow \ldots$.*

Moreover, considering it as a $WAF_{\mathbb{S}}$, the related infinite sequence in Figure 6 does not respect the defence conditions of $\mathbb{D}_{Mart\acute{i}nez\ et\ al.}$: T_4 cannot defend T_2 from T_3 because there is no attack from T_4 to c (whose values are 3, 2, 1), which is at least as strong as the attack from c to d in T_2 (i.e., 4 is stronger). Consequently, the framework in Figure 6 (left) is not well-founded when using $\mathbb{D}_{Mart\acute{i}nez\ et\ al.}$, in accordance with Definition 8.4.

As accomplished for acceptability notions, we can relate the property of a framework to be well-founded by considering the three proposals in Section 5. Since Definition 8.4 is parametrically based on a generic definition of weighted defence, we can plug the three notions directly in: e.g. $wfd_{Martínez\ et\ al.}$ is defined on $\mathbb{D}_{Martínez\ et\ al.}$. We also consider the classical well-founded property in [28], that is wfd_{Dung}, by simply removing weights from a $WAF_{\mathbb{S}}$, while keeping the same Ar and att.

Theorem 8.6 (Implications and well-foundedness)**.** *Given a $WAF_{\mathbb{S}}$, the following implications hold (where wfd_* is the well-founded property as derived from the defence notion described in the paper indicated by $*$):*

1. $wfd_{Dung} \Leftarrow wfd_{Bistarelli\ et\ al.}$, $wfd_{Dung} \Leftarrow wfd_{Martínez\ et\ al.}$, $wfd_{Dung} \Leftarrow wfd_{Coste-Marquis\ et\ al.}$.
2. $wfd_{Coste-Marquis\ et\ al.} \Leftarrow wfd_{Bistarelli\ et\ al.}$.
3. $wfd_{Coste-Marquis\ et\ al.} \Leftarrow wfd_{Martínez\ et\ al.}$.
4. *With the Fuzzy c-semiring,* $wfd_{Martínez\ et\ al.} \Leftrightarrow wfd_{Coste-Marquis\ et\ al.}$, $wfd_{Martínez\ et\ al.} \Rightarrow wfd_w$.
5. *With the Boolean c-semiring,* $wfd_{Bistarelli\ et\ al.} \Leftrightarrow wfd_{Martínez\ et\ al.} \Leftrightarrow wfd_{Coste-Marquis\ et\ al.}$.

The well-foundedness property is interesting because it points to a framework where there exists only one set of arguments that is worth to be considered under any semantics. According to [28], every well-founded AF has exactly one complete extension, which is also grounded, preferred, and stable. The same result is preserved also in each of the weighted approaches in Section 5. Theorem 8.7 formalises this result.

Theorem 8.7 (Uniqueness of extensions)**.** *Given a notion of weighted acceptability, any well-founded WAF where the grounded extension is also complete, has exactly one complete extension, which is also grounded, preferred, and stable.*

Note there is an additional condition with respect to [28]: it is related to the (weighted) grounded extension, which needs to be also complete according to Theorem 8.7. This condition is required by the fact that WAFs may have several grounded extensions, as in fact it may happen in [30], [41], [26], and [15]. This is in general not desirable, since the grounded extension should represent the most sceptical unique point of view. The multiplication of the grounded extensions is due to having several derived frameworks by disaggregating attacks ([30] and [41]), or to imposing additional constraints on weighted acceptability ([41] and [15]).

Uniqueness of the Grounded Extension. It could be interesting to relax the grounded extension in case it corresponds to the empty-set because it does not bring so much information. Nevertheless, it is also to maintain its unicity. In fact, the idea behind the grounded extension is to represent a core set of "least questionable" arguments, being composed of only non-attacked arguments and arguments defended by them (directly and indirectly). For this reason, having several of these least questionable positions makes them not so least questionable anymore.

For example, let us consider a $WAF_{\mathbb{S}}$ with arguments $Ar = \{a, b, c, d\}$, and $att(a, b)$, $att(b, c)$, $att(b, d)$, all with a weight of 1 (using the *Weighted* c-semiring): $w(a, b) = 1, w(b, c) = 1, w(b, d) = 1$. The set of weighted complete-extensions (*wcom* for short) in [15] is $\{\{a, c\}, \{a, d\}\}$, and then there is no single least element with respect to set inclusion: in a classical formulation of this semantics, we would have two *wgrd* extensions.[14]

Concerning the approach in [15], the authors of [18] re-obtain a single *wgrd* extension, which however is not always also *wcom*.[15] The *wgrd* extension is there defined as the maximal, w.r.t. set inclusion, *wadm* extension included in the intersection of *wcom* extensions. They follow the same approach used in [8] to alternatively define the ideal and eager semantics.

Definition 8.8 (Weighted grounded)**.** *Given a* $WAF_{\mathbb{S}}$*,* $F = \langle Ar, att, w, \mathbb{S}\rangle$*, an extension* $T = wgrd(F)$*, iff* $T \in wadm(F)$*, and* $T \subseteq \bigcap wcom(F)$*, and* $\nexists T' \in wadm(F)$ *satisfying* $T' \subseteq \bigcap wcom(F)$ *s.t.* $T \subsetneq T'$*.*

Some of the derived properties in [18] are also reported in the three propositions below.

Proposition 8.9 (Existence and unicity)**.** $wgrd(F)$ *always exists and is unique.*

Proposition 8.10. *If* $\mathbb{S}$ *is the* Boolean *c-semiring then* $wgrd(F)$ *is equivalent to the classical grounded extension [28].*

Proposition 8.11. $wgrd(F)$ *corresponds to the set of sceptically accepted arguments in* $wcom(F)$*.*

Considering the initial example in this paragraph, $wgrd(F) = \{a\}$ is not *wcom* but only *wadm*.

[14] According to [28] instead, the grounded extension is $\{a, c, d\}$ in this case. In order to differentiate them from their original formulation, in this paragraph we will use *wcom*, *wadm*, and *wgrd* to point to weighted complete, admissible, and grounded respectively.

[15] This possibility motivates the additional condition in Theorem 8.7.

9 Tools and Applications

We split this section in two parts. First we describe software tools that can be used as solvers of the weighted problems described in this paper, and then we show two applications on online platforms, Twitter.com and Amazon.com.

9.1 Tools

In the following we focus on ConArg and ConArgLib, respectively a solver and library based on the former stand-alone solver. Such tools can be used to compute both classical and weighted extensions, and write C++ programs that can implement decision-making procedure using them, for instance.

ConArg[16] [17; 14] is an Argumentation reasoner based on *Gecode*[17], which is an open, free, and efficient C++ library where to develop constraint-based applications. ConArg is able to find all the classical extensions on a given classical AF [28] and using one of the following semantics: conflict-free, admissible, complete, stable, grounded, preferred, semi-stable, eager, stage, and ideal. In addition, it can check the credulous or sceptical acceptance of a given argument with respect to semantics. Besides classical (unweighted) problems, ConArg also deals with WAFs, being able to solve α^{γ}-conflict-free, α^{γ}-admissible, α^{γ}-complete, α^{γ}-preferred, α^{γ}-stable, and α^{γ}-grounded extensions (as presented in Section 7). The solver is offered to users as a command-line executable, or through a Web-interface. ConArg has been extended to deal with probabilities [12] and ranking-based semantics [11].

ConArgLib is a C++ library implemented to help developers solve some problems related to extension-based Abstract Argumentation. ConArgLib represents one of the first attempts to provide a fast implementation of a library to support the solution of problems in Abstract Argumentation. A developer can use it as the basic brick to directly develop her own applications, instead of interfacing to an external solver: as an example, solving the existence of a non-empty extension, and the credulous/sceptical acceptance of arguments can be used to set-up a decision-making procedure by ranking arguments, and then selecting the decision supported by the strongest ones. ConArgLib solves all the problems solved by ConArg. Moreover, it extends it in two different ways: *i)* a developer may now choose different branching strategies (branching defines the shape of the search tree), and *ii)* can start a parallel search, using more than one thread at the same time.

[16] `http://www.dmi.unipg.it/conarg/`.

[17] `http://www.gecode.org`.

9.2 Applications

The most popular applications of WAFs are addressed to online social platforms, as social networks and e-commerce/reviewing portals. In this case weights can be extracted from different sources: e.g., the number of repetitions the same (or similar) argument has been proposed by different users, the number of likes, the number of shares, etc. In the following sections we show two applications, both using ConArg as the underlying engine.

A further application, using VAFs to help the analysis of Twitter.com discussions, is presented in [1].

9.2.1 Microdebates

Microdebates [34] are inspired by Twitter.comâĂŹs microblogging. A microdebate is a stream of tweets where users annotate their messages by using some special tags. Posts contain terms called hashtags, i.e. a # symbol followed by a text string, representing the stream of news the tweet belongs to. There may be more than one hashtag per post (in case the same post is related to multiple streams).

In such an application, a hashtag identifies the discussion (e.g., *#debateName*): as customary, this ensures that the tweet will appear in the right stream (a microdebate). Moreover, a user may take advantage of two additional special tags: $ and !$. *$opinionName* specifies the opinion a tweet supports, while *!$opinionName* specifies the opinion a tweet counters.

The *Microdebates App* is distributed via Google Play [46]. The application has a client-server architecture: the server runs a background process that manages the interaction with Twitte.comr. It maintains a MySQL database containing all the information that is shown on the client side: tweets, topics, word clouds, and other data extracted from the tweets, such as attacks and weights, needed to compute the extensions. In the Microdebates application, weights are determined by counting the tweets that express a given attack. The server retrieves from Twitter.com all new tweets about the topics listed in the database using a Java library for the Twitter.com API. To compute the extensions, the server process invokes ConArg methods for α-preferred extensions.

On the client side, the purpose is to provide the user with an interface to enter new tweets and, mainly, to navigate through microdebates. Microdebates uses computational (weighted) Argumentation to rank opinions and drive the visualisation. The result is a visual summary of the debate that takes into account semantic information such as explicit attack relations that link arguments together. If an argument a belongs to an α-preferred extension T (see Section 7), we know two

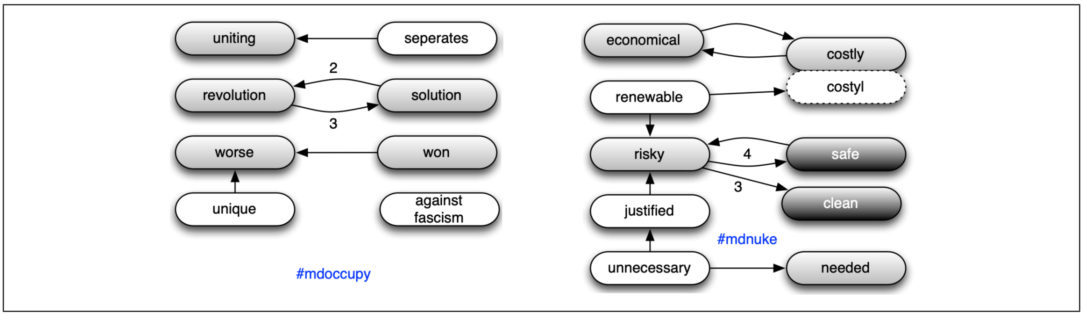


Figure 7: Two *WAF*s obtained with the Microdebates App, assembled through an experiment with real users (figure taken from [46]).

things: *i)* a can peacefully coexist with the other arguments in T, the inconsistency within T being at most α, and *ii)* for every tweet attacking it, there exists at least another tweet that counters the attack. So, it is reasonable to give arguments in T the status of "popular" argument.

The authors of [46] conducted microdebates experiments with ten participants in the 25-34 age group. Each group was given a topic, and a 40-minute time frame to discuss the topic by using Microdebates App. At the end of 40 minutes a two-hour break was given, and then a different topic was proposed, for a total of four different topics The first two topics were *"Are occupy protest movements justified?"* (*#mdoccupy*) and *"Is nuclear energy justified and should it be expanded?"* (*#mdnuke*), whose related *WAF*s are represented in Figure 7.

The first microdebate has three 1-sceptically preferred arguments (contained in all the extensions) displayed in white, and five 1-credulously preferred arguments (contained in at least one extension) displayed in grey. The second microdebate has two 3-sceptically preferred arguments (white), four 3-credulously preferred arguments (grey), and two losing arguments (black).

9.2.2 Amazon.com Reviews

In [32] and [33] the authors consider 253 reviews of a selected product (a ballet tutu for kids), extracted from the "Clothing, Shoes and Jeweller" section of Amazon.com. They manually extract abstract arguments from such reviews, and they study how their characteristics, e.g., the distribution of positive (in favour of purchase) and negative ones (against purchase), change through a period of four years. Among other results, they discover that negative arguments tend to permeate also positive reviews. As a second step, by using such observations and distributions, we successfully replicate the reviewers' behaviour by simulating the review-posting process from their basic components, i.e., the arguments themselves.

Reviews are in the period between January 2009 and July 2014. A total of 24 positive and 20 negative claims are manually identified. Arguments with the same claim are grouped together, and the number in each group represents the weight associated with them: for example, “The kid loved it” is a positive claim with a strength of 78, while “The tutu has a bad quality” is a negative claim with a weight of 18. Also attacks were manually extracted and the resulting AF is represented in Figure 8.

The set of classical stable (and semi-stable) extensions counts 256 different instances, which exactly correspond also to the set of preferred extensions. Complete extensions are $6,651$. Since all such information is complicated to be somehow analysed and interpreted (attacks are almost all symmetric), the authors switch to using the weighted Argumentation approach and the relaxation described in Section 6. In this case, it is necessary to relax defence to $\gamma = 22$ in order to obtain more than zero 0^{γ}-stable extensions, i.e., 16 in this case. Hence, we obtain a small subset of stable extensions with no internal conflict, but able to counter-attack better than the attack they receive. This set represents a refined subset of the original 256 stable extensions obtained when considering the graph as unweighted.

10 Summary and Future Research

In this paper we have surveyed Weighted Argumentation Frameworks, or WAFs for short. More in particular, we summarised different key-points of their formalisation, relaxation, semantics, and applications. Such frameworks can be useful when a more fine-grained level of detail is necessary to measure the acceptability of arguments by looking at the strength of attacks.

Preference systems, both qualitative and quantitative, are used in this paper to refine the notion of defence: different preference systems are used in the literature, from simple values in $\mathbb{R}$ to more general and parametric structures as c-semirings, which also deal with partially-ordered attacks. Having weights as labels allows for tolerating a certain amount of attack among arguments: some works name this inconsistency budget α or β, which is the “sum” of the disregarded weights. Besides such an internal (with respect to extensions) relaxation, also defence can be mitigated by thus allowing more arguments to be defended by the same set: a complete relaxation brings to totally ignore the difference between attack and defence weights, consequently leading to classical defence [28]. In addition, extension-based semantics can be revised according to previous concepts, and it is then possible to define weighted complete, preferred-extensions, and so on. Having redefined AFs

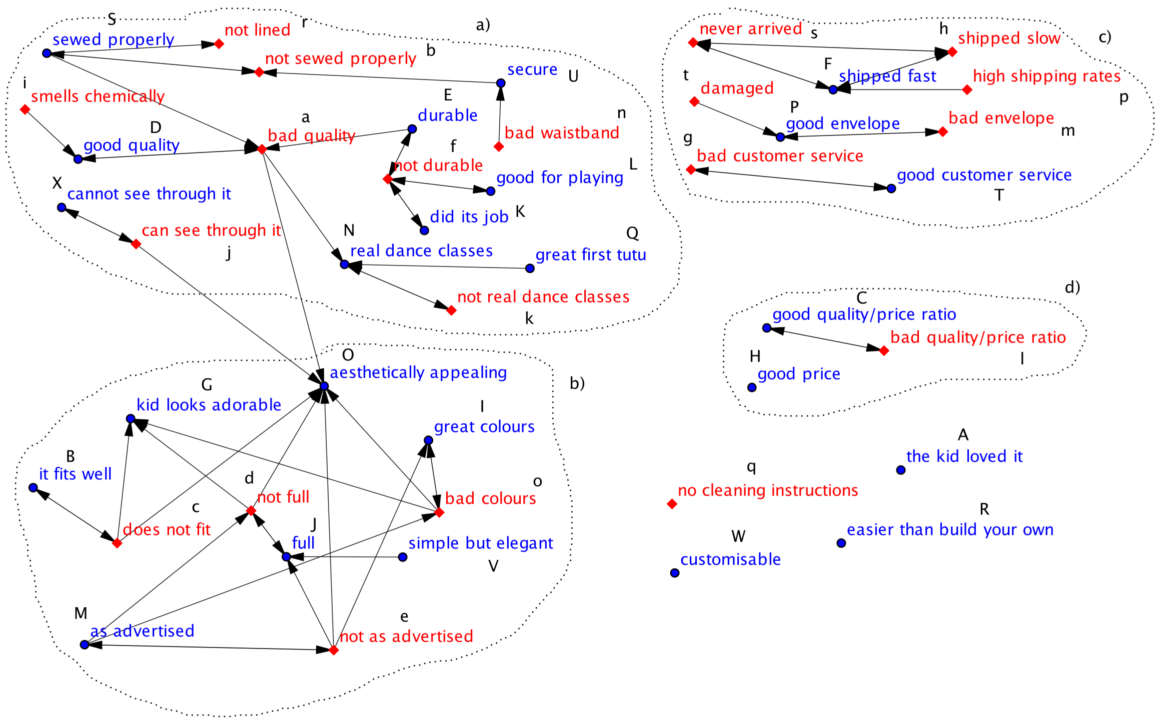


Figure 8: The *AF* manually extracted from 253 reviews about a ballet tutu for kids sold in Amazon.com. Abstract arguments represented by (blue) circles claim in favour of purchasing the tutu, while the ones represented by (red) rhombuses are against purchasing. Claims are grouped in four clusters concerning the product quality (a), aspect (b), shipping (c), and price (d). Figure taken from [32].

to WAFs, also classical properties of frameworks need to be revised, as for example whether a framework is well-founded or not: in well-founded frameworks, the grounded extension as well as the complete/preferred/stable extensions coincide. Finally, we have presented how WAFs naturally find an application to the world of social-media, where information is rated by users, and such scores can be easily converted to preference values for what posted.

Future research. Despite the quite large number of proposals in the literature, there is still a considerable number of open research lines concerning WAFs. For example, a few works go beyond the definition of weighted semantics and investigate if well-established properties that hold for classical frameworks in [28] also hold in WAFs. Section 8 explores well-founded frameworks, but other properties, such as for example, the existence of (weighted) stable extensions, or the uniqueness of a (weighted) preferred extension, need to be generalised as well. First steps towards formalising these results initiated in [19].

In addition, considering Section 3, the different translation methods of weights from arguments to attacks are not proven to maintain the semantics between the original framework and the destination one. The relations among these approaches need to be examined further.

Moreover, complexity results are only clearly stated in [30]: the β versions of the decision problems are in fact no harder (although of course no easier) than the corresponding unweighted decision problems, except for the grounded extension: in [30] there are several grounded extensions, and consequently the computation of e.g. the sceptical acceptance of an argument is coNP-complete, instead of being trivial as in [28]. More general results along this direction are needed, considering also the other presented approaches.

One additional path to investigate concerns *Weighted Bipolar Argumentation Frameworks* [23], where the set of edges in a graph is bipartite into support and attack relationships. For example, the proposal in [2] adopts weights on arguments: it will be interesting to check how compensating defence and attack techniques can be adapted to support and attack edges, and how the relaxation presented in Section 6 can be used in bipolar settings. Along the same line, we can study *weighted Higher-order Argumentation Frameworks*, where attacks may target attacks, and they are in turn associated with a weight.

Finally, existing applications prove that weighted frameworks are really useful when paired to social platforms as Twitter.com or Amazon.com, because such systems natively encourage users to rate posts or reviews. We reckon that the development of *Argument Mining* techniques on social platforms and the analysis of threads via WAFs can foster the use of such platforms (or dedicated “rooms” in them) as debating systems, undermining the effect of fake news and summarising how the discussion is structured.

References

[1] Teresa Alsinet, Josep Argelich, Ramón Béjar, Cèsar Fernández, Carles Mateu, and Jordi Planes. Weighted argumentation for analysis of discussions in twitter. *Int. J. Approx. Reason.*, 85:21–35, 2017.

[2] Leila Amgoud and Jonathan Ben-Naim. Weighted bipolar argumentation graphs: Axioms and semantics. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI*, pages 5194–5198. ijcai.org, 2018.

[3] Leila Amgoud, Jonathan Ben-Naim, Dragan Doder, and Srdjan Vesic. Acceptability semantics for weighted argumentation frameworks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI*, pages 56–62. ijcai.org, 2017.

[4] Leila Amgoud and Claudette Cayrol. On the acceptability of arguments in preference-based argumentation. In *UAI '98: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 1–7. Morgan Kaufmann, 1998.

[5] Katie Atkinson and Trevor Bench-Capon. Value-based argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 5. College Publications, 2021.

[6] Pietro Baroni, Dov M Gabbay, Massimiliano Giacomin, and Leendert van der Torre. *Handbook of formal argumentation.* College Publications, 2018.

[7] Howard Barringer, Dov M. Gabbay, and John Woods. Temporal dynamics of support and attack networks: From argumentation to zoology. In *Mechanizing Mathematical Reasoning, Essays in Honor of Jörg H. Siekmann on the Occasion of His 60th Birthday*, volume 2605 of *Lecture Notes in Computer Science*, pages 59–98. Springer, 2005.

[8] Ringo Baumann and Christof Spanring. Infinite argumentation frameworks - on the existence and uniqueness of extensions. In *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation*, volume 9060 of *LNCS*, pages 281–295. Springer, 2015.

[9] Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.

[10] Philippe Besnard and Anthony Hunter. A logic-based theory of deductive arguments. *Artif. Intell.*, 128(1-2):203–235, 2001.

[11] Stefano Bistarelli, Francesco Faloci, and Carlo Taticchi. Implementing ranking-based semantics in conarg. In *31st IEEE International Conference on Tools with Artificial Intelligence, ICTAI*, pages 1180–1187. IEEE, 2019.

[12] Stefano Bistarelli, Theofrastos Mantadelis, Francesco Santini, and Carlo Taticchi. Probabilistic argumentation frameworks with metaproblog and conarg. In *IEEE 30th Inter-*

national Conference on Tools with Artificial Intelligence, ICTAI, pages 675–679. IEEE, 2018.

[13] Stefano Bistarelli, Ugo Montanari, and Francesca Rossi. Semiring-based constraint satisfaction and optimization. *J. ACM*, 44(2):201–236, 1997.

[14] Stefano Bistarelli, Fabio Rossi, and Francesco Santini. Conarg: A tool for classical and weighted argumentation. In *Computational Models of Argument - Proceedings of COMMA*, volume 287 of *FAIA*, pages 463–464. IOS Press, 2016.

[15] Stefano Bistarelli, Fabio Rossi, and Francesco Santini. A novel weighted defence and its relaxation in abstract argumentation. *Int. J. Approx. Reason.*, 92:66–86, 2018.

[16] Stefano Bistarelli and Francesco Santini. A common computational framework for semiring-based argumentation systems. In *ECAI 2010 - 19th European Conference on Artificial Intelligence*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 131–136. IOS Press, 2010.

[17] Stefano Bistarelli and Francesco Santini. Conarg: A constraint-based computational framework for argumentation systems. In *IEEE 23rd International Conference on Tools with Artificial Intelligence, ICTAI*, pages 605–612. IEEE Computer Society, 2011.

[18] Stefano Bistarelli and Francesco Santini. A hasse diagram for weighted sceptical semantics with a unique-status grounded semantics. In *Logic Programming and Nonmonotonic Reasoning - 14th International Conference, LPNMR 2017*, volume 10377 of *Lecture Notes in Computer Science*, pages 49–56. Springer, 2017.

[19] Stefano Bistarelli and Francesco Santini. Well-foundedness in weighted argumentation frameworks. In *Logics in Artificial Intelligence - 16th European Conference, JELIA*, volume 11468 of *Lecture Notes in Computer Science*, pages 69–84. Springer, 2019.

[20] Elise Bonzon, Jérôme Delobelle, Sébastien Konieczny, and Nicolas Maudet. A comparative study of ranking-based semantics for abstract argumentation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 914–920. AAAI Press, 2016.

[21] Jean-Rémi Bourguet, Leila Amgoud, and Rallou Thomopoulos. Towards a unified model of preference-based argumentation. In *Foundations of Information and Knowledge Systems, 6th International Symposium, FoIKS*, volume 5956 of *Lecture Notes in Computer Science*, pages 326–344. Springer, 2010.

[22] Claudette Cayrol, Caroline Devred, and Marie-Christine Lagasquie-Schiex. Acceptability semantics accounting for strength of attacks in argumentation. In *ECAI 2010 - 19th European Conference on Artificial Intelligence*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 995–996. IOS Press, 2010.

[23] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Gradual valuation for bipolar argumentation frameworks. In Lluís Godo, editor, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 8th European Conference, ECSQARU*, volume 3571 of *Lecture Notes in Computer Science*, pages 366–377. Springer, 2005.

[24] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Graduality in argumentation. *J. Artif. Intell. Res.*, 23:245–297, 2005.

[25] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. From preferences over arguments to preferences over attacks in abstract argumentation: A comparative study. In *25th IEEE International Conference on Tools with Artificial Intelligence, ICTAI*, pages 588–595. IEEE Computer Society, 2013.

[26] Sylvie Coste-Marquis, Sébastien Konieczny, Pierre Marquis, and Mohand Akli Ouali. Weighted attacks in argumentation frameworks. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR*, pages 593–597. AAAI Press, 2012.

[27] Célia da Costa Pereira, Andrea Tettamanzi, and Serena Villata. Changing one's mind: Erase or rewind? In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 164–171. IJCAI/AAAI, 2011.

[28] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[29] Phan Minh Dung. An axiomatic analysis of structured argumentation with priorities. *Artif. Intell.*, 231:107–150, 2016.

[30] Paul E. Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457–486, 2011.

[31] Sinan Egilmez, João Martins, and João Leite. Extending social abstract argumentation with votes on attacks. In *Theory and Applications of Formal Argumentation - Second International Workshop, TAFA*, volume 8306, pages 16–31. Springer, 2013.

[32] Simone Gabbriellini and Francesco Santini. A micro study on the evolution of arguments in amazon.com's reviews. In *PRIMA 2015: Principles and Practice of Multi-Agent Systems - 18th International Conference*, volume 9387 of *LNCS*, pages 284–300. Springer, 2015.

[33] Simone Gabbriellini and Francesco Santini. From reviews to arguments and from arguments back to reviewers' behaviour. In *Agents and Artificial Intelligence - 8th International Conference, ICAART*, volume 10162 of *LNCS*, pages 56–72. Springer, 2016.

[34] Simone Gabbriellini and Paolo Torroni. Microdebates: Structuring debates without a structuring tool. *AI Commun.*, 29(1):31–51, 2016.

[35] Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming: An argumentative approach. *TPLP*, 4(1-2):95–138, 2004.

[36] Anthony Hunter, Sylwia Polberg, Nico Potyka, Tjitze Rienstra, and Matthias Thimm. Probabilistic argumentation: A survey. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 7. College Publications, 2021.

[37] Souhila Kaci and Christophe Labreuche. Valued preference-based instantiation of argumentation frameworks with varied strength defeats. *Int. J. Approx. Reason.*, 55(9):2004–2027, 2014.

[38] Souhila Kaci and Cristophe Labreuche. Arguing with valued preference relations. In

Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 11th European Conference, ECSQARU, volume 6717 of *Lecture Notes in Computer Science*, pages 62–73. Springer, 2011.

[39] Souhila Kaci, Leendert van der Torre, Srdjan Vesic, and Serena Villata. Preference in abstract argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 3. College Publications, 2021.

[40] João Leite and João Martins. Social abstract argumentation. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 2287–2292. IJCAI/AAAI, 2011.

[41] Diego C. Martínez, Alejandro Javier García, and Guillermo Ricardo Simari. An abstract argumentation framework with varied-strength attacks. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference*, pages 135–144. AAAI Press, 2008.

[42] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artif. Intell.*, 173(9-10):901–934, 2009.

[43] Sanjay Modgil and Henry Prakken. The *ASPIC*$^+$ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.

[44] Sanjay Modgil and Henry Prakken. Abstract rule-based argumentation. *FLAP*, 4(8), 2017.

[45] Henry Prakken and Giovanni Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7(1):25–75, 1997.

[46] Nefise Yaglikci and Paolo Torroni. Microdebates app for android: A tool for participating in argumentative online debates using a handheld device. In *26th IEEE International Conference on Tools with Artificial Intelligence, ICTAI*, pages 792–799. IEEE Computer Society, 2014.

Received May 2021

Enforcement in Formal Argumentation

Ringo Baumann
Department of Computer Science, Leipzig University, Germany
baumann@informatik.uni-leipzig.de

Sylvie Doutre
Institut de Recherche en Informatique de Toulouse, Université Toulouse Capitole, France
doutre@irit.fr

Jean-Guy Mailly
Laboratoire d'Informatique Paris Descartes, Université de Paris, France
jean-guy.mailly@u-paris.fr

Johannes P. Wallner
Institute of Software Technology, Graz University of Technology, Austria
wallner@ist.tugraz.at

Abstract

Within argumentation dynamics, a major strand of research is concerned with how changing an argumentation framework affects the acceptability of arguments, and how to modify an argumentation framework in order to guarantee that some arguments have a given acceptance status. In this chapter, we overview the main approaches for enforcement in formal argumentation. We mainly focus on extension enforcement, i.e., on how to modify an argumentation framework to ensure that a given set of arguments becomes (part of) an extension. We present different forms of extension enforcement defined in the literature, as well as several possibility and impossibility results. The question of minimal change is also considered, i.e., what is the minimal number of modifications that must be made to the argumentation framework for enforcing an extension. Computational complexity and algorithms based on a declarative approach are discussed. Finally, we briefly describe several notions that do not directly fit our definition of extension enforcement, but are closely related.

1 Introduction

At the beginning of the 2010s several problems regarding *dynamic aspects* of abstract argumentation have been addressed in the literature [30; 38; 28; 65]. One much cited problem among these is the so-called *enforcing problem* dealing with changing the acceptability of certain arguments [17]. Over the years, the problem gained more and more attention which finally leads to the writing of this chapter. In its very first version the problem can be briefly summarized as the question whether it is possible, given a specific type of syntactic changes, to modify a given AF such that a desired set of arguments becomes (a subset of) an extension. Consider the following snapshot of a dialogue among agents A and B depicted in Figure 1. Assume it is A's turn and her desired set of arguments is $E = \{a_1, a_2, a_3\}$. Furthermore, A and B are discussing under preferred semantics, which selects maximal conflict-free and self-defending sets of arguments.

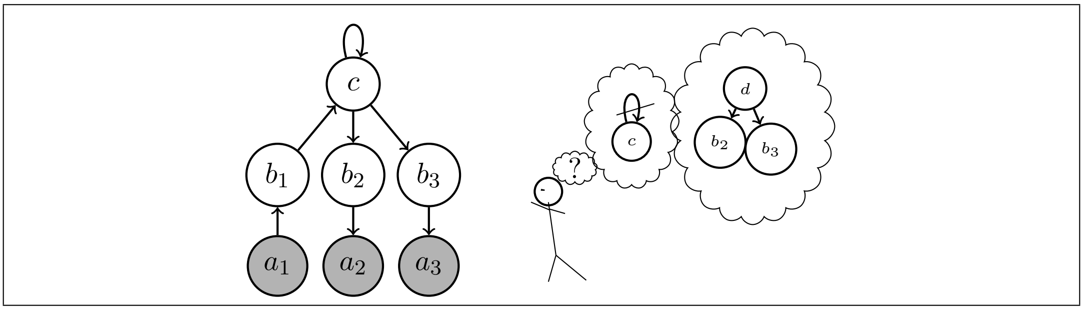


Figure 1: Snapshot of a dialogue

In order to enforce E, agent A may come up with new arguments which interact with the old ones (for example through introducing an argument d which attacks b_2 and b_3) and/or question old arguments or attacks between them, respectively (for example through questioning the self-attack of c). Please note that first, in this scenario, enforcing is possible and second, that there are at least two different possibilities to achieve that. This insight leads to a further well-studied issue, namely the so-called *minimal change problem* firstly introduced in [13]. This problem is defined as a generalization of the classical enforcing problem since one is not only interested in whether enforcements are *possible*, but also in the *effort needed* to enforce a set of arguments. One numerical measure which is frequently used for this effort corresponds to the number of additions or removals of attacks to reach such an enforcement. The main motivation behind this measure is that adding or removing an isolated argument does not contribute at all to solving or increasing a given conflict, i.e. the conflicting information remains the same. This means, the decrease or increase of a conflict is directly linked to upcoming or disappearing attacks and thus, counting attacks only is a reasonable approach. Regarding the introductory example we obtain a minimal effort of 1 if allowing arbitrary modifications.

In this chapter we give an overview over main variants of enforcement studied in the literature. We give a particular focus on strict and non-strict extension enforcement, whose aim is to modify an AF such that a desired set of arguments becomes exactly (or part of) an extension, under a semantics. A main distinguishing factor among the family of operators for extension enforcement is how an AF may be modified. We highlight here changes corresponding to expansions, i.e., additions of arguments and attacks such as the addition of argument d above, or local updates, i.e., modifying only the attack structure such as questioning the self-attack of c, but also discuss modifications to AFs more broadly, as well. Additionally, we consider as an instance of a change that does not affect the structure of the framework, modifications of the chosen semantics, in order to enforce a set of arguments.

We present main formal properties of extension enforcement derived in the literature, e.g., for impossibility and possibility results, and results for the minimal change problem of extension enforcement. We further survey results regarding the complexity of reasoning on enforcement and present algorithms based on declarative approaches to implement enforcement.

The chapter starts off with recalling formal preliminaries of AFs (Section 2) including types of modifications on AFs. The main section on extension enforcement is Section 3, which first introduces enforcement as a general problem, and focuses on the extension enforcement variant. In this section, we present expansion-based extension enforcement and extension enforcement based on locally updating an attack structure without modifying the set of arguments. Further, minimal change, semantics change, complexity results, and algorithms, are presented. In Section 4 we survey related notions to enforcement, and we close with a discussion of related works (Section 4.5) and with conclusions (Section 5).

2 Formal Preliminaries

In order to keep the chapter self-contained we review all relevant definitions. We start with the basic notions of Dung's abstract argumentation theory [54].

2.1 Argumentation Frameworks and Semantics

An *abstract argumentation framework (AF)* is just a directed graph $F = (A, R)$ where a node $a \in A$ is called an *argument* and a pair $(a, b) \in R \subseteq A \times A$ is interpreted as an *attack* from argument a to argument b. We require that any AF

$F = (A, R)$ possesses arguments from a fixed reference set $\mathcal{U}$, i.e. $A \subseteq \mathcal{U}$. Moreover, in this chapter we restrict ourselves to finite AFs, i.e. any AF consists of finitely many arguments and attacks only. Note that this is a common restriction in the literature although actual and potential infinite AFs play an important role in practical applications as well as theoretical considerations (cf. [8; 22; 16] for more information). At the heart of Dung's abstract argumentation theory are *argumentation semantics* which formalize intuition of what should be acceptable in the light of conflicts. Two main approaches to argumentation semantics can be found, namely so-called *extension-based* and *labelling-based* versions (cf. [7] for an introduction and [10, Sections 2.2, 4.4] for further relations). In this chapter we concentrate on the former only. Consider the following generic definition. The set $\mathcal{F}$ refers to all considered AFs.

Definition 2.1. *A semantics is a total function*

$$\sigma : \mathcal{F} \to 2^{2^{\mathcal{U}}} \quad F = (A, R) \mapsto \sigma(F) \subseteq 2^{A}.$$

A set of arguments $E \in \sigma(F)$ is called a *σ-extension.* Moreover, we say that a semantics σ is *universally defined* if each AF admits at least one extension with respect to this semantics, i.e. for any $F \in \mathcal{F}$, $|\sigma(F)| \geq 1$. Furthermore, a semantics σ is said to be *uniquely defined* if always exactly one set of arguments is returned, i.e. $|\sigma(F)| = 1$ for any $F \in \mathcal{F}$.

Before presenting the relevant semantics for this chapter we have to introduce some further notation. Given an AF $F = (A, R)$ and a set $E \subseteq A$. We use E^+_F, or simply E^+, for $\{b \mid (a, b) \in R, a \in E\}$. Moreover, $E^{\oplus}_F$, or simply $E^{\oplus}$, is called the *range* of E and stands for $E^+ \cup E$. Analogously, E^-_F (or simply E^-) stands for $\{b \mid (b, a) \in R, a \in E\}$, and $E^{\ominus}_F$ (or simply $E^{\ominus}$) corresponds to $E^- \cup E$. An argument a is *defended* by E (in F) if for each $b \in A$ with $(b, a) \in R$, b is attacked by some $c \in E$. Finally, $\Gamma_F : 2^A \to 2^A$ with $I \mapsto \{a \in A \mid a \text{ is defended by } I\}$ denotes the so-called *characteristic function* (of F).

Besides conflict-free and admissible sets (abbreviated by *cf* and *ad*) we consider a large number of well-known semantics, namely naive, stage, stable, semi-stable, complete, preferred, grounded, ideal, and eager semantics (abbreviated by *na*, *stg*, *stb*, *sst*, *co*, *pr*, *gr*, *id*, *eg*, respectively).

Definition 2.2. *Let $F = (A, R)$ be an AF and $E \subseteq A$.*

1. *$E \in cf(F)$ iff for no $a, b \in E$, $(a, b) \in R$,*

2. *$E \in na(F)$ iff E is $\subseteq$-maximal in $cf(F)$,*

3. *$E \in stg(F)$ iff $E \in cf(F)$ and $E^{\oplus}$ is $\subseteq$-maximal in $\{I^{\oplus} \mid I \in cf(F)\}$,*

4. *$E \in stb(F)$ iff $E \in cf(F)$ and $E^{\oplus} = A$,*

5. *$E \in ad(F)$ iff $E \in cf(F)$ and $E \subseteq \Gamma_F(E)$,*

6. *$E \in sst(F)$ iff $E \in ad(F)$ and $E^{\oplus}$ is $\subseteq$-maximal in $\{I^{\oplus} \mid I \in ad(F)\}$,*

7. *$E \in co(F)$ iff $E \in cf(F)$ and $E = \Gamma_F(E)$,*

8. *$E \in pr(F)$ iff E is $\subseteq$-maximal in $co(F)$,*

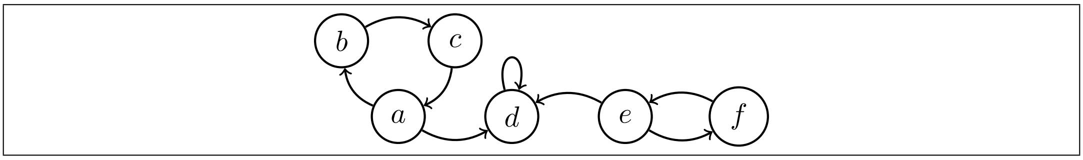
9. *$E \in gr(F)$ iff E is $\subseteq$-minimal in $co(F)$,*

10. *$E \in id(F)$ iff E is $\subseteq$-maximal in $\{I \mid I \in ad(F), I \subseteq \bigcap pr(F)\}$,*

11. *$E \in eg(F)$ iff E is $\subseteq$-maximal in $\{I \mid I \in ad(F), I \subseteq \bigcap sst(F)\}$.*

It has been shown that any of the introduced semantics is universally defined except the stable one and moreover, grounded, ideal and eager semantics are even uniquely defined (cf. [21] for an overview). In order to get familiar with the introduced definitions consider the following example taken from [15].

Example 2.3. *Consider the AF $F = (A, R)$ with $A = \{a, b, c, d, e, f\}$ and $R = \{(a, b), (a, d), (b, c), (c, a), (d, d), (e, d), (e, f), (f, e)\}$. The graphical representation of F is given below.*



The evaluation of F w.r.t. the introduced semantics is given in the following table. The entry "✓" in row "σ" and line "E" stands for $E \in \sigma(F)$.

	stb	*sst*	*stg*	*pr*	*ad*	*co*	*gr*	*id*	*eg*	*na*
$\emptyset$					✓	✓	✓	✓		
$\{e\}$		✓		✓	✓	✓			✓	
$\{f\}$				✓	✓	✓				
$\{a, e\}$			✓							✓
$\{b, e\}$			✓							✓
$\{c, e\}$			✓							✓
$\{a, f\}$			✓							✓
$\{b, f\}$										✓
$\{c, f\}$										✓

Table 1: Evaluation table of F

The AF F is an example for a collapse of stable semantics, i.e. $stb(F) = \emptyset$. The non-existence of stable extensions in F implies the occurrence of odd-length cycles like the 3-cycle $[a, b, c, a]$ or the self-loop $[d, d]$. More precisely, in case of finite AFs we have that being odd-cycle free is sufficient for warranting at least one stable extension [54; 81].

As already indicated in Table 1 there are several well-known subset relations between the considered semantics. For instance, for any AF F we have, $stb(F) \subseteq sst(F) \subseteq pr(F) \subseteq co(F) \subseteq ad(F)$ and $stb(F) \subseteq stg(F) \subseteq na(F)$.

2.2 Acceptance Modes and Structural Changes

In the following we present several acceptance modes and structural changes, that is, changes on the structure (addition or removal of arguments and attacks) of the AF, which can be used to specify a certain type of enforcement.

So-called *credulous* and *sceptical acceptance* are the most common reasoning types for abstract argumentation semantics. They are usually defined for single arguments only. We present their definitions for sets of arguments where the classical single argument acceptance can be obtained by considering the singleton of the argument in question. Moreover, since a non-universally defined semantics σ may return no σ-extension for a given AF F we consider so-called *non-empty sceptical reasoning* which avoids the (possibly) unintended situation that every argument is sceptically accepted due to the emptiness of $\sigma(F)$. A further frequently used acceptance mode is the requirement to be contained in at least one extension, so-called *covered acceptance*[1]. This notion plays a central role in the field of enforcement and is located in-between non-empty sceptical and credulous acceptance.

Definition 2.4. *Given a semantics σ, an AF $F = (A, R)$ and a set $E \subseteq A$. We say that E is*

1. *credulously accepted w.r.t. σ if $E \subseteq \bigcup \sigma(F)$,*
2. *sceptically accepted w.r.t. σ if $E \subseteq \bigcap \sigma(F)$,*
3. *non-empty sceptically accepted w.r.t. σ if $E \subseteq \bigcap \sigma(F)$ and $\sigma(F) \neq \emptyset$,*
4. *covered accepted w.r.t. σ if there is an $E' \in \sigma(F)$, s.t. $E \subseteq E'$.*

For convenience we introduce the following unified notation. We write $E \in \mathit{cred}(F, \sigma)$, $E \in \mathit{scep}(F, \sigma)$, $E \in \mathit{scep}^{\neq\emptyset}(F, \sigma)$ or $E \in \mathit{cov}(F, \sigma)$ for E is credulously, sceptically, non-empty sceptically or covered accepted, respectively. Moreover, for any given reasoning type r we use $E \in r_s(\sigma, F)$ to indicate that there is an equality instead of a subset relation only, e.g. there is an $E' \in \sigma(F)$, s.t. $E = E'$ in the case of covered acceptance (or, said otherwise, $E \in \sigma(F)$). In this case we say that the considered set E is *strictly* accepted. If E is non-empty sceptically accepted w.r.t. σ then E is covered accepted w.r.t. σ (since E must be part of all σ-extensions and there is at least one), and the latter implies that E is credulously accepted w.r.t. σ (since the witness for being covered accepted is a witness for credulous acceptance).

Let us proceed with the running AF exemplifying several acceptance modes.

Example 2.5 (Example 2.3 cont.). *Let $\sigma = \mathit{stb}$. Since $\mathit{stb}(F) = \emptyset$ we obtain $\bigcup \mathit{stb}(F) = \emptyset$ and $\bigcap \mathit{stb}(F) = \mathcal{U}$. Hence, any set $E \subseteq \mathcal{U}$ is sceptically, but not non-empty sceptically accepted, i.e. $E \in \mathit{scep}(F, \mathit{stb})$ and $E \notin \mathit{scep}^{\neq\emptyset}(F, \mathit{stb})$. Moreover, E is neither credulously, nor covered accepted, i.e. $E \notin \mathit{cred}(F, \mathit{stb})$ and*

[1] We mention that this notion is sometimes called credulous acceptance [55, p. 704]. This is due to the fact that that there are at least two options if generalizing credulous acceptance from arguments to sets of arguments.

$E \notin cov(F, stb)$.
Consider now $\sigma = pr$. Since $pr(F) = \{\{e\}, \{f\}\}$ we have $\bigcup pr(F) = \{e, f\}$ and $\bigcap pr(F) = \emptyset$. Thus, $\{e, f\}$ is credulously strict but neither sceptically nor non-empty sceptically accepted, i.e. $\{e, f\} \in cred_s(F, pr)$, $\{e, f\} \notin scep(F, pr)$ and $\{e, f\} \notin scep^{\neq\emptyset}(F, pr)$. Moreover, $\{e, f\}$ is not covered accepted whereas $\{e\}$ and $\{f\}$ are and this acceptance is even strict, i.e. $\{e, f\} \notin cov(F, pr)$ and $\{e\}, \{f\} \in cov_s(F, pr)$.

We now introduce typical structural changes. The most general form of dynamic scenarios are so-called *updates* where arguments and attacks can be deleted and added. If we do not delete any information we call the structural change an *expansion* [17; 76; 12]. The following kinds of expansions have received particular attention in the literature. *Normal expansions* add new arguments and possibly new attacks which concern at least one of the fresh arguments. Moreover, *local expansions* do not introduce any new arguments but possibly new attacks among the old arguments. Both types of expansions naturally occur in the context of instantiation-based argumentation [27; 35]. For instance, adding a new piece of information to the underlying knowledge base corresponds to a normal expansion on the AF level. Furthermore, changing the considered notion of attack left the constructed arguments untouched and results in a local expansion. Two further subconcepts of normal expansions are usually considered, so-called *strong* and *weak expansions*. Their names refer to properties of the additional arguments, namely arguments which are never attacked by former arguments (*strong* arguments) and arguments which do not attack former arguments (*weak* arguments). The former type typically occurs in a debate if one tries to strengthen the own point of view via rebutting the opponents arguments. Note that weak expansions seem to be more an academic exercise than a task with practical relevance with regard to real-world argumentation. However, they do play a decisive role in the context of *splittings* [11; 19; 6].

Consider the formal definition of the discussed types of expansions.

Definition 2.6. *An AF G is an expansion of AF $F = (A, R)$ (for short, $F \preceq_E G$) iff $G = (A \cup B, R \cup S)$ for some (maybe empty) sets B and S, s.t. $A \cap B = R \cap S = \emptyset$. An expansion is called*

1. *normal ($F \preceq_N G$) iff $\forall ab \ \ ((a, b) \in S \rightarrow a \in B \vee b \in B)$,*
2. *strong ($F \preceq_S G$) iff $F \preceq_N G$ and $\forall ab \ \ ((a, b) \in S \rightarrow \neg(a \in A \wedge b \in B))$,*
3. *weak ($F \preceq_W G$) iff $F \preceq_N G$ and $\forall ab \ \ ((a, b) \in S \rightarrow \neg(a \in B \wedge b \in A))$,*
4. *local ($F \preceq_L G$) iff $B = \emptyset$.*

Example 2.7. *Consider the following simple AF F. The presented AFs F_X represent examples for $F \preceq_X F_X$. This means, F_N is a normal expansion of F. Note that grey-highlighted arguments or attacks represent added information.*

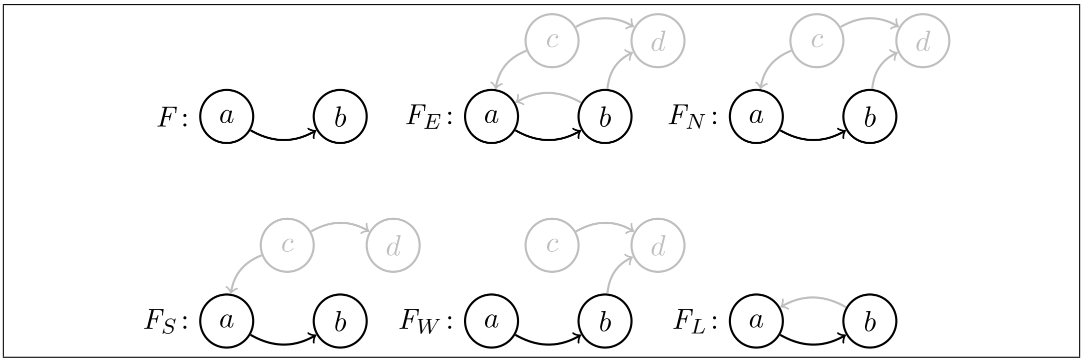


Figure 2: Different kinds of expansions

The natural counter-parts to expansions are so-called *deletions* where no further arguments and attacks are added [30; 28; 14]. We consider two sub-classes of *deletions* representing the inverse operations to normal and local expansions, namely *normal* and *local deletions*. Normal deletions retract arguments and their corresponding attacks. Such a kind of structural change occurs in the instantiation-based context if we delete information from the underlying knowledge base. Changing to a more restrictive notion of attack corresponds to a local deletion where only attacks are discarded.

In order to present the precise formal meaning of deletions we have to introduce some operations on directed graphs first. First, we use $F \sqcup H$ for the pointwise union of two AFs. In Definition 2.8, such an union is used in order to represent the addition of information (encoded in H) to an initial AF (F). Secondly, the restriction of $F = (A, R)$ to a set $B \subseteq A$ abbreviated as $F|_B$ is given via $(B, R \cap (B \times B))$.

Definition 2.8. *Given an AF $F = (A, R)$, a set of arguments B and a set of attacks S as well as a further AF H. The AF*

$$G = (F \setminus [B, S]) \sqcup H := \Big((A, R \setminus S)|_{A \setminus B}\Big) \sqcup H$$

is called an update of F (for short, $F \asymp_U G$). An update is called a

1. *deletion ($F \succeq_D G$) iff $H = (\emptyset, \emptyset)$,*

2. *normal deletion ($F \succeq_{ND} G$) iff $F \succeq_D G$ and $S = \emptyset$,*

3. local deletion ($F \succeq_{LD} G$) iff $F \succeq_D G$ and $B = \emptyset$.

Let us take a closer look at the definition of $G = (F \setminus [B, S]) \sqcup H$. The AF H plays the role of added information, i.e. it contains new arguments and attacks. Consequently, for all kind of deletions we have $H = (\emptyset, \emptyset)$ which leaves us with $G = F \setminus [B, S]$. The set B contains arguments which have to be deleted. Since attacks depend on arguments, we have to delete the attacks which involve arguments from B too. This operation is formally captured by the restriction of F to $A \setminus B$. Furthermore, the set S contains particular attacks which have to be deleted. This means, the pair $[B, S]$ does not necessarily have to be an AF. Therefore we use $[B, S]$ instead of (B, S). If clear from context we use B and S instead of $[B, \emptyset]$ or $[\emptyset, S]$, i.e. we simply write $F \setminus B$ as well as $F \setminus S$ for normal or local deletions, respectively. Note that the different kinds of expansion presented in Definition 2.6 can be captured by setting $B = S = \emptyset$. Deletions and expansions are dual concepts: $F \preceq_E G$ if and only iff $G \succeq_D F$, and similarly for the normal or local versions.

Example 2.9. *The AF F represents the initial situation. An update as well as arbitrary, normal or local deletion of it are given by F_U, F_D, F_{ND} and F_{LD}. Grey-highlighted arguments or attacks represent added information in contrast to dotted arguments and attacks which represent deleted objects.*[2] *More formally, in accordance with Definition 2.8 we have that $F_U = (F \setminus [B, S]) \sqcup H$, $F_D = F \setminus [B, S]$, $F_{ND} = F \setminus B$, $F_{LD} = F \setminus S$ where the set of arguments $B = \{c\}$, the set of attacks $S = \{(b, a)\}$ and the AF $H = (\{b, d, e, f\}, \{(d, b), (e, f), (f, d)\})$.*

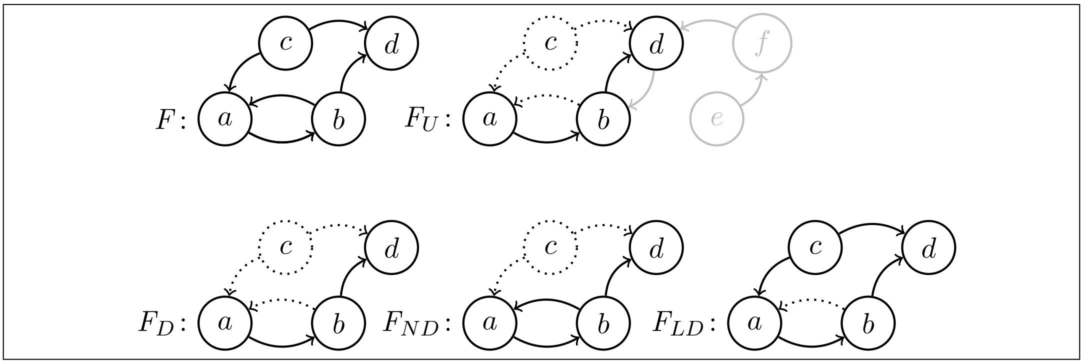


Figure 3: An update and different kinds of deletions

[2] This convention will be used throughout the whole chapter.

3 Enforcement

3.1 The General Setup

The starting point of any extension enforcement case is:

- an AF F,
- a semantics σ,
- a certain desired set of arguments E, together with
- a reasoning, acceptance mode r, e.g. credulous, sceptical, non-empty sceptical, covered, with a strict or non-strict goal achievement (cf. Section 2.2).

In addition, parameters indicating the way of achieving the enforcement can be specified, namely:

- allowed types of structural changes like update, expansion and deletion (cf. Section 2.2),
- allowed types of semantic changes, if any (cf. Section 3.2.4), and
- whether these changes would have to be minimal, and in which sense (cf. Section 3.2.3).

For illustrative purposes let us assume that r stands for credulous acceptance. Consequently, enforcement is needed if and only if E is not credulously accepted w.r.t. σ in F, i.e. $E \notin \mathit{cred}(F, \sigma)$. This is why we often speak of the *desired* set of arguments E since we want to fix the defect of non-acceptance. In order to achieve this goal we have two main options, namely structural changes and/or semantic changes. More precisely, we are looking for changes of AFs, from F to G, and/or semantics, from σ to τ, s.t. E is credulously accepted w.r.t. τ in G, i.e. $E \in \mathit{cred}(G, \tau)$. The way of how to perform the structural change is fixed in advance. For instance, one may require that only local expansions of F are allowed, i.e. $F \preceq_L G$. The same applies to the semantic change. One may allow changes to any kind of semantics or to admissibility-based ones only. Another option would be to completely forbid semantic changes, i.e. $\tau = \sigma$. In the following definition, we call a *modification type* $M \subseteq \mathcal{F} \times \mathcal{F}$ a relation such that $(F, G) \in M$ iff, when F is an initial AF, then G is a possible result of modifying F. For instance, $M = \preceq_L$ means that only local expansions are authorized.

Consider the following formal definition of an enforcement.

Definition 3.1. *Given two AFs F and G, two semantics σ and τ, a modification type $M \subseteq \mathcal{F} \times \mathcal{F}$, a set of argument E, and a reasoning mode r. A pair (G, τ) is called an (F, σ, M, r)*-enforcement *of E if*

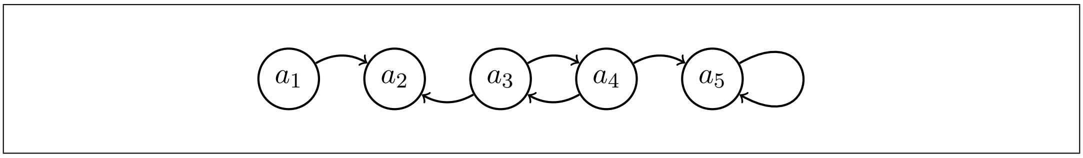
1. *$(F, G) \in M$ and*
2. *$E \in r(G, \tau)$.*

Moreover, we call G the τ-enforcing AF *and we say that E is τ*-enforced by *G.*

The different kinds of expansions and deletions presented in Definitions 2.6, 2.8 are captured by setting $M \in \{\preceq_E, \preceq_N, \preceq_S, \preceq_W, \preceq_L, \asymp_U, \succeq_D, \succeq_{ND}, \succeq_{LD}\}$. Note that $F \preceq_N G$ can be equivalently rewritten as $(F, G) \in \preceq_N$ since $\preceq_N$ is formally a binary relation over $\mathcal{F}$, i.e. $\preceq_N \subseteq \mathcal{F} \times \mathcal{F}$. Whenever F, σ, M and r are clear from context we simply speak of enforcements of E. If the set in question is strictly accepted we speak about a *strict* enforcement (for instance, $r = cov_s$), otherwise *non-strict* (for instance, $r = cov$). Moreover, we distinguish between *conservative* ($\sigma = \tau$) and *liberal* enforcements ($\sigma \neq \tau$). The latter may be interpreted as a change of proof standard or paradigm shift. Imagine a judicial proceeding. Here it is vitally important whether you are accused on the base of criminal or civil law. The required evidence is different and hence the acceptable sets of arguments differ.

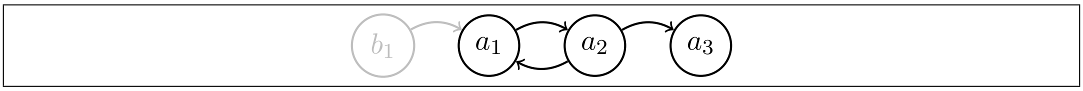
Consider the following two examples taken from [17].

Example 3.2 (liberal, strict)**.** *Given F as presented below, $\sigma = stb$, $M = \asymp_U$, $r = cov_s$ and the desired set $E = \{a_1, a_3\}$.*



Since $stb(F) = \{\{a_1, a_4\}\}$ we have $E \notin cov_s(stb, F)$. How to enforce E? Define an enforcement (G, τ) of E with $F = G$ and $\tau = pr$. Note that $pr(G) = \{\{a_1, a_3\}, \{a_1, a_4\}\}$ justifies the claim because $E \in cov_s(G, pr)$ holds. The considered enforcement is strict and liberal and F is the pr-enforcing AF.

Example 3.3 (conservative, non-strict)**.** *Given $\sigma = gr$, $M = \preceq_S$, $r = cov$, $E = \{a_2\}$ and $F = (\{a_1, a_2, a_3\}, \{(a_1, a_2), (a_2, a_1), (a_2, a_3)\})$ as presented below.*



Note that $gr(F) = \{\emptyset\}$. Hence, $E \notin cov(gr, F)$. In this example we allow strong expansions only. Is it possible to enforce E? The answer is "yes". Consider the enforcement (G, τ) of E with G defined as depicted above and $\tau = \sigma$. Since $gr(G) = \{\{b_1, a_2\}\}$ we deduce $E \in cov(G, gr)$. The considered enforcement is non-strict and conservative and G is the gr-enforcing AF.

3.2 Extension Enforcement with Structural Change

We start with a review of one of the most prominent enforcement operators in the literature, named extension enforcement [17; 13; 43; 52; 85; 60]. Extension enforcement refers to a family of enforcement operators that all deal with covered acceptance, i.e., the enforcement goal is to modify a given AF such that a desired set of arguments becomes an extension, or becomes part of an extension, under a semantics. Both strict and non-strict variants were studied.

The main distinguishing aspect of various extension enforcement operations is what kind of modification type is permitted. Concretely, we look at extension enforcement allowing only expansions (Section 3.2.1), permitting only local modifications (Section 3.2.2), i.e., changing the attack structure, restricting change to be minimal (Section 3.2.3), and changing semantics (Section 3.2.4).

3.2.1 Expansion-based enforcement

In this section we consider conservative (non-)strict enforcements w.r.t. covered reasoning mode under different forms of expansions. More precisely, for a given AF $F = (A, R)$, a semantics σ and a desired set of arguments $E \subseteq A$ we look at pairs (G, σ) being (F, σ, M, cov) enforcements of E. We allow $M \in \{\preceq_E, \preceq_N, \preceq_S, \preceq_W\}$, i.e. arbitrary, normal, strong, and weak expansions are considered. In the following, for the sake of brevity, we do not explicitly mention the covered acceptance mode as well as the conservativeness.

We have already seen a case of non-strict extension enforcement under strong expansions in Example 3.3. We now exemplify some properties of extension enforcement under expansions.

Example 3.4. *Let us consider an AF $F = (A, R)$ with $A = \{a, b, c, d\}$ and an attack relation as shown in Figure 4. Say we want to enforce $E = \{b, d\}$ to be part of an admissible extension in a non-strict manner, and allowing arbitrary expansions. An AF G that ad-enforces these constraints is shown, as well, in Figure 4. That is, expanding by two arguments e and f and adding attacks (b, f), (f, d), and (e, c) results in $\{e, b, d\} \in ad(G)$, and thus E is non-strictly enforced to be part of an admissible extension by G. Note that adding the single attack (d, c) only wouldn't* *do the job since we are interested in non-strict enforcements. However, there are many more ways to non-strictly enforce the desired set E. We encourage the reader to find other witnessing ad-enforcing expansions.*

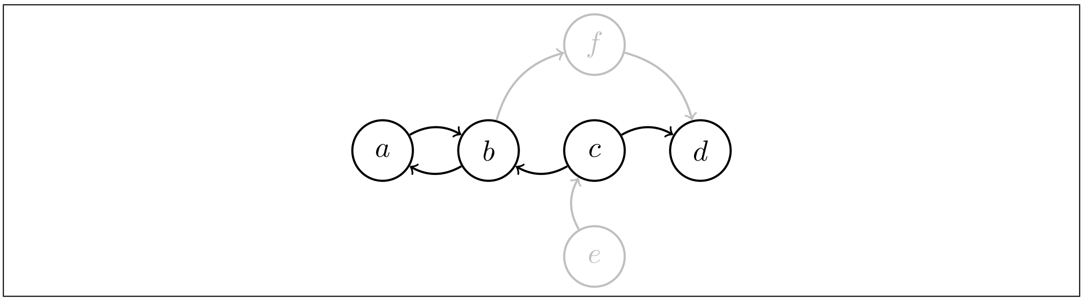


Figure 4: AF and expanded AF from Example 3.4

The preceding example illustrates the existence of enforcements. However, in general, desired enforcements might not exist. Consider the following example.

Example 3.5. *Consider again the AF $F = (A, R)$ of Figure 4. We illustrate now three different sources for the impossibility of enforcements.*

1. *Assume we aim to strictly ad-enforce $E = \{b, d\}$ under normal expansions. While non-strict enforcement of E was possible (cf. Example 3.4), strict enforcement is impossible under normal expansions. The intuition is that $\{b, d\}$ is not admissible in the original AF F (the attack from c to d is not defended) and this fact remains true in any normal expansion G of F. The reason is simply that any new attack in G involves at least one new argument and thus, E can not defend d in G. However, E can be strictly enforced when allowing arbitrary expansions, e.g. adding a defending attack (b, c) is an option.*

2. *Another reason for impossibility of enforcement occurs when considering enforcement of sets like $\{a, b\}$ under any semantics σ that preserves conflict-freeness, i.e. $\sigma \subseteq cf$. The reason is that $\{a, b\}$ is conflicting in F and thus, it remains conflicting regardless the considered type of expansion.*

3. *Even if the set E to be enforced is conflict-free and defends all its elements, enforcement is, under specific semantics, not always possible. Consider the aim to strictly co-enforce $E = \{c\}$ under weak expansions. In F the singleton E is not complete since it defends a and a $\notin E$. Now, weak expansions do not raise new attacks onto existing arguments which implies that former defense relations survive. Thus, for any weak expansion G of F we have E still defends a preventing it from being complete in G.*

The previous observations have been firstly formalized in [17, Proposition 1] and later considered further in [43, Proposition 1]. In the following we recall some results and generalize them to other semantics considered in this article.

Proposition 3.6. *Given an AF $F = (A, R)$ and $E \subseteq A$.*

- *If $E \notin ad(F)$ and $\sigma \subseteq ad$, then there is no AF G strictly σ-enforcing E under normal expansions.*
- *If $E \notin cf(F)$ and $\sigma \subseteq cf$, then there is no AF G (non-)strictly σ-enforcing E under arbitrary expansions.*
- *If E does not contain all defended arguments in F and $\sigma \subseteq co$, then there is no AF G that strictly σ-enforcing E under weak expansions.*
- *If $\sigma \in \{ad, cf, na, stb\}$ and $E \notin \sigma(F)$, then there is no AF G strictly σ-enforcing E under normal expansions.*

Despite several cases being impossible to enforce, there are interesting conditions under which an enforcement is always possible. As an illustration, consider the following example.

Example 3.7. *Say we desire to non-strictly ad-enforce $E = \{b, d\}$ under strong expansions. This means, we want E to be a strict subset of an admissible extension of the expanded framework. An example AF G ad-enforcing $\{b, d\}$ is shown in Figure 5. Here the new argument e is added which defends both b and d. Since $\{e\}$ is admissible in G we obtain via the famous Fundamental Lemma [54, Lemma 10] that $\{e, b, d\}$ is admissible as desired.*

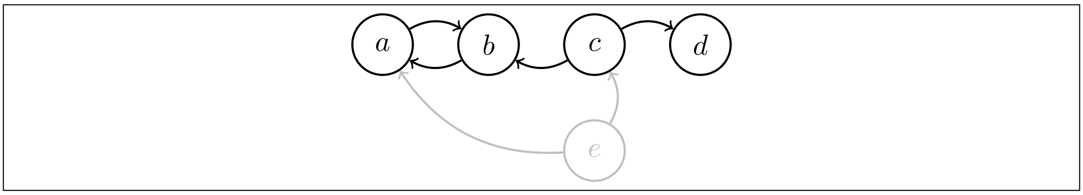


Figure 5: AF from Example 3.7

The observation from the example was generalized to further semantics [17, Theorem 4]. The main construction method is to extend the initial framework with a new argument which attacks all *undesired* arguments. We extend the already proven theorem to all semantics considered in this article.

Theorem 3.8. *Given an AF F, a desired set $E \in cf(F)$ and a semantics $\sigma \in \{ad, stb, pr, co, gr, id\, sst, eg, na, stg\}$. There is a strong expansion G of F non-strictly σ-enforcing E.*

Since strong expansions are particular cases of normal expansions as well as arbitrary expansions, we may state the following corollary.

Corollary 3.9. *Given an AF F, a desired set $E \in cf(F)$ and a semantics $\sigma \in \{ad, stb, pr, co, gr, id\, sst, eg, na, stg\}$. There are arbitrary as well as normal expansions G of F non-strictly σ-enforcing E.*

What about local expansions? Is it possible to (non-)strictly enforce a desired set E with local manipulations only? For most of the existing semantics we may act as follows: given the conflict-freeness of E we attack all remaining arguments first (this is sufficient for $\sigma \in \{ad, stb, pr, co, sst, na, stg\}$) and secondly, add self-loops to the remaining arguments (we additionally cover $\sigma \in \{id, eg\}$).

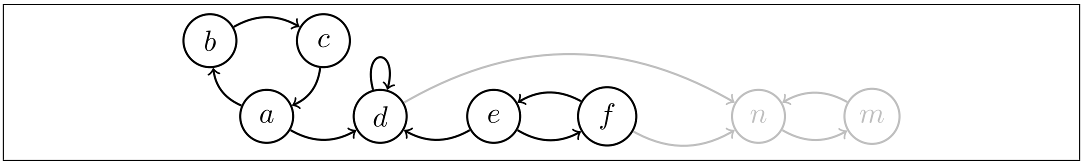
Theorem 3.10. *Given an AF F, a desired set $E \in cf(F)$ and a semantics $\sigma \in \{ad, stb, pr, co, id\, sst, eg, na, stg\}$. There is a local expansion G of F strictly σ-enforcing E.*

Note that grounded semantics is not included since it requires unattacked arguments which can not be “produced" with the help of local expansions. However, if there is an unattacked argument in the desired set E, then this unattacked argument can be used to attack all the arguments outside the directed set, leading to the strict *gr*-enforcement of E. Any unattacked argument in the AF can have a similar role for non-strict enforcement.

Theorem 3.11. *Given an AF F and a desired set $E \in cf(F)$, if there is an unattacked argument $a \in E$ (respectively $a \in A$), then there is a local expansion G of F strictly (respectively non-strictly) enforcing E under the grounded semantics.*

Let us turn now to a different aspect of enforcing, namely *how* exactly existing σ-extensions may change when expanding an AF. In general, the change is very much non-monotone: this means, arguments accepted earlier may become unaccepted, others become accepted; the number of extensions may shrink or increase, depending on the new arguments. For instance, it is easy to verify that we obtain a total collapse of stable extensions if we revise an AF by adding a self-defeating argument. Nevertheless, there are a few exceptions as illustrated in the following example taken from [15, Example 3.11]

Example 3.12. *Consider the weak expansion G of F as depicted below. In Example 2.3 we already observed that $pr(F) = \{\{e\}, \{f\}\} = \{E_1, E_2\}$.*



For the weak expansion G we find $pr(G) = \{E_1 \cup \{n\}, E_1 \cup \{m\}, E_2 \cup \{m\}\}$. Consequently, the following interrelations hold:

1. *the number of extensions increased*
2. *every old belief set is contained in a new one*
3. *every new belief set is the union of an old one and a new argument*

The previous example contrasts with the general observation that adding new arguments and attacks may change the outcome of an AF in a nonmonotonic fashion. Such a behaviour allows for reusing already computed extensions and has useful implications w.r.t. justification states. The following theorem [15, Theorem 3.2] shows that the class of weak expansions and semantics satisfying the directionality principle guarantee monotonic evolvements. Roughly speaking, the directionality criterion captures the idea that the evaluation of a certain argument should only be affected by its attackers and the attackers of its attackers and so on [9].

Theorem 3.13. *Given an AF $F = (A, R)$ and a semantics σ satisfying directionality, then for all weak expansions $G = (B, S)$ of F we have:*

1. $|\sigma(F)| \leq |\sigma(G)|$, *(cardinality)*
2. $\forall E \in \sigma(F)\ \exists E' \in \sigma(G)\ \exists C \subseteq B \setminus A,$ *s.t.* $E' = E \cup C$ *and* *(subset)*
3. $\forall E' \in \sigma(G)\ \exists E \in \sigma(F)\ \exists C \subseteq B \setminus A,$ *s.t.* $E' = E \cup C$. *(representation)*

It is well-known that admissible, complete, preferred, grounded and ideal semantics satisfy directionality (cf. [83] for an overview). Having the above theorem at hand we obtain the following relations regarding acceptance modes stating that credulously, sceptically as well as covered accepted sets persist.

Proposition 3.14. *Given an AF $F = (A, R)$ and $\sigma \in \{ad, co, pr, gr, id\}$. For any weak expansions G of F we have:*

- $cred(F, \sigma) \subseteq cred(G, \sigma)$,
- $scep(F, \sigma) \subseteq scep(G, \sigma)$ *and*
- $cov(F, ad) \subseteq cov(G, ad)$

3.2.2 Attack-based enforcement: Argument-fixed and Local Expansion-based Enforcement

We now turn to extension enforcement under a different kind of modifications to a given AF. In contrast to the previous section on expansion-based enforcement where expansion of the set of arguments and attacks, under certain conditions, was presented, we here look at changes that do not modify the set of arguments, but exclusively focus on updates of the attack structure.

Definition 3.15. *Let $F = (A, R)$ be an AF. We say that G is a local update of F, denoted by $F \asymp_L G$, if there is an AF G' such that $F \preceq_L G'$ and $G' \succeq_{LD} G$.*

In words, an AF $G = (A_G, R_G)$ is a local update of $F = (A_F, R_F)$ if there is an intermediate AF $G' = (A_{G'}, R_{G'})$ that is a local expansion of F (i.e., $A_{G'} = A_F$ and $R_F \subseteq R_{G'}$) and G is a local deletion of G' (i.e., $A_{G'} = A_G$ and $R_{G'} \supseteq R_G$). Put differently, G is a local update of F if the set of arguments stays the same, i.e., $A_G = A_F$, and the attack structure was changed arbitrarily: $R_G = (R_F \setminus R) \cup R'$ for some $R, R' \subseteq A_F \times A_F$.

In this section we consider extension enforcement under local updates [43]. An intuition of a local update is that the arguments are unmodified, but some new attacks are revealed (e.g., in presence of new information), and some attacks are disputed and discarded (e.g., due to the defeasibility of attacks). Modifying the attacks between existing arguments can also be seen as an update of the preferences between arguments [2].[3]

Example 3.16. *Let us look at the same AF F from the preceding section that we used to exemplify expansion-based enforcement. We recall this AF in Figure 6a.*

We begin with looking at enforcement of the set $\{b, d\}$. Say, we desire to have this set of arguments being part of an admissible extension. In F the set $\{b, d\}$ is conflict-free but not admissible: the attack from c onto both b and d is not countered. A local update, in fact a local expansion, that enforces $\{b, d\}$ to be part of an admissible extension is shown in Figure 6b. An attack from b to c suffices to have $\{b, d\}$ defend both b and d.

[3] Recall that in preference-based argumentation, the “success" of an attack (a, b) depends on the fact that b is not preferred to a.

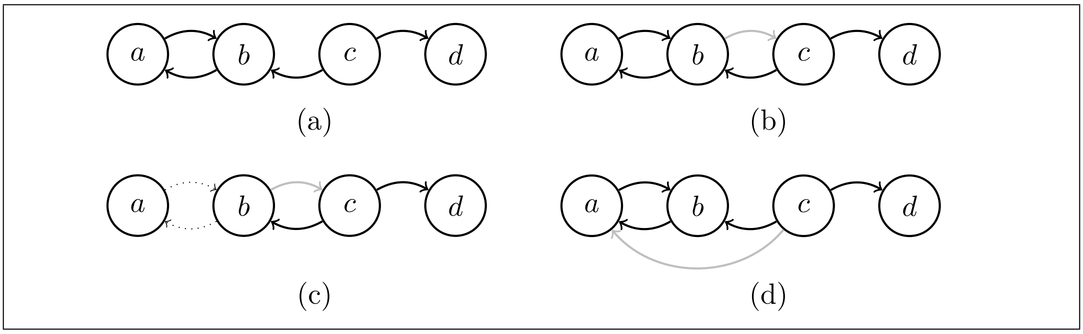


Figure 6: Enforcement by local updates

A different case is exhibited by aiming to have $\{a, b\}$ *being admissible: this set neither is conflict-free nor defends its arguments. A possible local update is shown in Figure 6c that enforces* $\{a, b\}$ *to be exactly an admissible extension, i.e., realizes strict extension enforcement under local updates and admissibility. Here, the conflicts between* a *and* b *are removed, to ensure conflict-freeness, and the attack from* b *to* c *is added, to ensure defense.*

Finally, consider strict enforcement of $\{c\}$ *under complete semantics. The set* $\{c\}$ *is admissible, yet defends* a *in* F*. A possible local update (local expansion) is shown in Figure 6d. Here one attack from* c *to* a *ensures that* $\{c\}$ *does not defend* a*.*

Inspection of the preceding example reveals that several impossible cases, when requiring certain expansions (see previous section), are, in fact, possible under local updates. This is no coincidence: enforcement under local updates is possible for all main semantics of AFs: if $E \neq \emptyset$ is to be enforced, for a given AF $F = (A, R)$ there is the (trivial) local update $G = (A, R')$ with $R' = \{(a, b) \mid a \in E, b \in A \setminus E\}$ (i.e., in G, every argument in E is non-attacked, and every argument in $A \setminus E$ is attacked by all arguments in E). We have $E \in gr(G)$, and since the graph structure of G is acyclic[4], most semantics coincide with the grounded semantics.

This observation is formalized next [43, Proposition 4].

Proposition 3.17. *Let* $F = (A, R)$ *be an AF and* $E \subseteq A$ *be a non-empty set of arguments. There exists a local update* G *that enforces* E *(non-)strictly to be (part of) a* σ*-extension, for all* σ *considered in this chapter.*

[4] In the case of finite AFs, acyclicity corresponds to the well-foundedness property defined by [54], which implies the coincidence of grounded, stable, preferred and complete semantics. We also refer the reader to [7] for more details on this topic.

Obviously, when $E = \emptyset$, it can always be non-strictly enforced with local updates, since E is included in any set of arguments. It is also the case that E can be strictly enforced with local update.[5] Indeed, for a given AF $F = (A, R)$ we can define the (trivial) local update $G = (A, R')$ with $R' = \{(a, a) \mid a \in A\}$ (i.e., in G, every argument is self-attacking). In this case, the empty set is the only conflict-free set, and thus the only extension for most semantics.

We have seen that enforcing a set of arguments with local updates is possible in general. Both the addition and the removal of attacks are necessary for this results. Indeed, if only local expansions are possible (i.e. removing attacks is not permitted), then a conflicting set E cannot be enforced under any semantics that requires conflict-freeness. Similarly, local deletions are not sufficient for strictly enforcing a set of arguments in all cases. As a matter of example, let us consider again the AF $F = (A, R)$ given at Figure 6a. The set $\{c\}$ cannot be enforced as a stable extensions by only deleting attacks: initially $\{c\}^{\oplus} = \{b, c, d\} \neq A$, and removing attacks cannot add arguments to the range of $\{c\}$.

3.2.3 Extension Enforcement and Minimal Change

Minimal change is an important topic in other domains of artificial intelligence, like belief change [1; 63]. In the context of extension enforcement, the question asked is "how much effort will it cost to perform the enforcement?". This effort is defined by [13] as the number of attacks that are modified (i.e. either added or removed). Formally,

Definition 3.18. *Given $F = (A, R)$ and $G = (A', R')$, the distance between F and G is $d(F, G) = |(R \setminus R') \cup (R' \setminus R)|$.*

In general, there may be several ways to enforce an extension, even for a fixed type of modification. In that case, minimal change enforcement consists in choosing one result that minimizes the distance d between the initial AF and the new one.

Example 3.19. *Figure 7 presents two examples of strong expansions of an AF $F = (A, R)$, with $A = \{a, b, c, d, e\}$ and $R = \{(b, a), (c, a), (d, b), (d, c), (e, d)\}$. This AF has a single stable extension: $stb(F) = \{\{b, c, e\}\}$. Both expansions succeed in non-strictly enforcing the set $\{a\}$ as a stable extension. However, we observe a difference in the number of attacks that have been added. The first one, F_1 (on the left side), adds two attacks, one from the new argument f_1 to c, and another one from f_2 to b; it has a single stable extension $stb(F_1) = \{\{a, e, f_1, f_2\}\}$. The second*

[5] Except for the stable semantics, since the empty set can never be a stable extension of a non-empty AF.

expansion, F_2 (on the right side), adds a single attack (f_3, e), and it also has a single stable extension: $stb(F_2) = \{\{a, d, f_3\}\}$. With $d(F, F_1) = 2$ and $d(F, F_2) = 1$, F_2 seems to be a more desirable result.

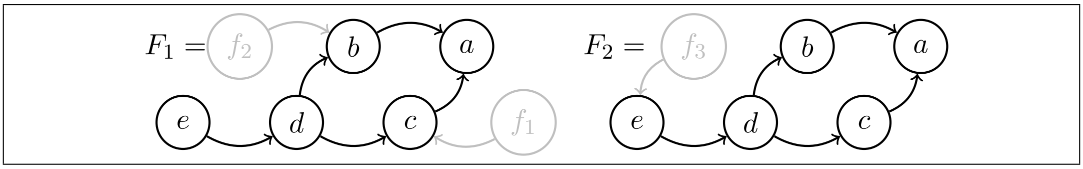


Figure 7: An example of (non-)minimal change

The question of minimal change in enforcement is studied in [13]. More specifically, it concerns the minimal change in non-strict enforcement based on normal expansions, as well as the special cases of strong and weak expansions. To do so, he defines the notion of characteristic of a set of arguments S, with respect to an AF F and a modification type $M \subseteq \mathcal{F} \times \mathcal{F}$. This characteristic corresponds to the minimal distance between F and an AF G such that S is included in an extension of G, and G is a possible result for the enforcement (i.e. $(F, G) \in M$). Strict enforcement can be considered as well [52].

Definition 3.20. *Given a semantics σ, a modification type $M \subseteq \mathcal{F} \times \mathcal{F}$, $x \in \{s, ns\}$ meaning strict or non-strict, and an AF $F = (A, R)$, the (σ, M, x)-characteristic of a set $S \subseteq A$ is:*

$$N^{F,x}_{\sigma,M}(S) = \begin{cases} 0 & \textit{if} \quad x = s, S \in \sigma(F) \\ 0 & \textit{if} \quad x = ns, \exists S' \in \sigma(F) \ \textit{s.t.} \ S \subseteq S' \\ k & \textit{if} \quad k = \min(\{d(F, G) \mid (F, G) \in M, N^{G,x}_{\sigma,M}(S) = 0\}) \\ +\infty & \quad \textit{otherwise} \end{cases}$$

Intuitively, the characteristic of a set of arguments S is 0 if this set is already (included in) an extension, k if k is the minimal distance between the initial AF and some AF that enforces S, and $+\infty$ if S cannot be enforced (under the the specified semantics and modification type).

Then, [13] introduces the notion of *value function*, that gives a constructive definition of how to compute the characteristic in a finite number of steps, based on properties of the initial AF. We use $V^{F,x}_{\sigma,M}(S)$ to denote this value function.

We start with the case of non-strict enforcement under weak expansion. Baumann shows that for most semantics, either the set S is already included in an extension, or it is impossible to enforce it with a weak expansion [13, Theorem 6]. Formally,

Proposition 3.21. *For $\sigma \in \{stb, ad\}$ a semantics, $F = (A, R)$ and AF and $S \subseteq A$ a set of arguments, the value function for non-strict enforcement under weak expansion and the semantics σ is*

$$V^{F,ns}_{\sigma,\preceq_W}(S) = \begin{cases} 0 & \text{if } \exists S' \in \sigma(F) \text{ s.t. } S \subseteq S' \\ +\infty & \text{otherwise} \end{cases}$$

Then, $N^{F,ns}_{stb,\preceq_W}(S) = V^{F,ns}_{stb,\preceq_W}(S)$ and $N^{F,ns}_{\sigma,\preceq_W}(S) = V^{F,ns}_{ad,\preceq_W}(S)$ for $\sigma \in \{ad, co, pr\}$.

Now, we turn to (non-strict) enforcement under strong expansion., i.e. we focus on defining $V^{F,ns}_{\sigma,\preceq_S}(S)$. This case is slightly more involved than the previous one, and it requires additional definitions.

Definition 3.22. *Given $F = (A, R)$ an AF and $X \in cf(F)$,*

- $ad(F, X) = X^{\ominus} \setminus X^{\oplus}$;
- $stb(F, X) = A \setminus X^{\oplus}$.

Intuitively, these sets correspond to the arguments that should be defeated in order to make X an admissible (respectively stable) extension of F. They can be used to define the value function for enforcement under strong expansion, for $\sigma \in \{stb, ad\}$. Interestingly, these value functions can be used also for enforcing a set of arguments under normal expansion or general expansions, as stated by [13, Theorem 9].

Proposition 3.23. *For $\sigma \in \{stb, ad\}$ a semantics, $F = (A, R)$ and AF and $S \subseteq A$ a set of arguments, the value function for non-strict enforcement under strong expansion and the semantics σ is*

$$V^{F,ns}_{\sigma,\preceq_S}(S) = \min(\{|\sigma(F, S')| \mid S \subseteq S' \text{ and } S' \in cf(F)\})$$

Then, $N^{F,ns}_{stb,M}(S) = V^{F,ns}_{stb,\preceq_S}(S)$ and $N^{F,ns}_{\sigma,M}(S) = V^{F,ns}_{\sigma,\preceq_S}(S)$ hold for $\sigma \in \{ad, co, pr\}$ and $M \in \{\preceq_E, \preceq_N, \preceq_S\}$.

This means that authorizing more kinds of modifications than the addition of strong arguments is useless regarding the issue of minimal change.

Then, an interesting result [13, Proposition 11] states that enforcement is always possible if arbitrary updates are permitted, i.e. attacks can also be deleted (contrary to expansions, where attacks can only be added).

Proposition 3.24. *For $\sigma \in \{stb, sst, pr, co, ad\}$ and any $F = (A, R)$,*

$$N^{F,ns}_{\sigma,U}(S) \leq |R \cap (S \times S)| + |A \setminus S|$$

Intuitively, it says that we can enforce S as (a subset of) an extension by making it conflict-free (i.e. removing the attacks in $R \cap (S \times S)$) and attacking every argument that is not in S (i.e. adding attacks from fresh arguments to arguments in $A \backslash S$). This finite upper bound guarantees that non-strict enforcement under arbitrary updates is always possible. But a more precise evaluation of the characteristics is given by this value function [13, Theorem 12]:

Proposition 3.25. *For $\sigma \in \{stb, ad\}$ a semantics, $F = (A, R)$ and AF and $S \subseteq A$ a set of arguments, the value function for non-strict enforcement under arbitrary updates and the semantics σ is*

$$V_{\sigma,U}^{F,ns}(S) = \min(\{|R \cap (S' \times S')| + |\sigma(F, S')| \mid S \subseteq S' \subseteq A\}$$

with $ad(F, S')$ and $stb(F, S')$ as in Definition 3.22. Then, $N_{stb,U}^{F,ns}(S) = V_{stb,U}^{F,ns}(S)$ and $N_{\sigma,U}^{F,ns}(S) = V_{\sigma,U}^{F,ns}(S)$ hold for $\sigma \in \{pr, co, ad\}$.

Finally, [52] presents characteristics for enforcement under local updates, i.e. when the set of arguments has to remain the same, but attacks between them can be added or deleted. The results are reminiscent of the ones described in this section.

3.2.4 Semantics-based Enforcement

Extension enforcement is usually defined as an operation where the target semantics is given as an input. We call it conservative enforcement when the target semantics is the same as the initial semantics, and liberal enforcement otherwise. On the contrary, [52] proposes to generalize enforcement, by enhancing operators with a set Σ of possible target semantics. Then, the chosen semantics is the one that allows to enforce the set of arguments with minimal change on the graph. More formally:

Definition 3.26. *For $F = (A, R)$ an AF, $S \subseteq A$ the set of arguments to be enforced and Σ a set of semantics, a strict (resp. non-strict) enforcement of S in F under a given modification type $M \subseteq \mathcal{F} \times \mathcal{F}$, is a pair (G, σ') such that*

1. *$(F, G) \in M$;*
2. *$\sigma' \in \Sigma$ and $S \in \sigma'(G)$ (resp. $S \subseteq S' \in \sigma'(G)$);*
3. *$\forall \sigma'' \in \Sigma$, $V_{\sigma',M}^{F,x}(S) \leq V_{\sigma'',M}^{F,x}(S)$ (with $x \in \{s, ns\}$).*

This means that the new semantics is chosen in a way that guarantees that the change on the graph is minimal. Since the characteristics can be the same for several semantics σ', additional criteria can be used in order to select the new semantics, like the distance between σ' and the initial semantics σ [51].

Finally, we already mentioned that [17, Section 3.1] discusses the tool of changing semantics in order to enforce a desired set. The authors presented two involved impossibility theorems specifying properties of initial extensions and desired sets, initial and target semantics as well as the considered type of structural change. Regarding the semantic change we have that possible target semantics were restricted to semantics satisfying well-known abstract criteria like admissibility or reinstatement (cf. [83] for an exhaustive overview). The mentioned theorems show either limitations for exchanging accepted arguments with formerly unaccepted ones (under normal expansions) or limitations for eliminating arguments of existing extensions (under weak expansions).

3.3 Complexity and Algorithms

We review complexity of enforcement problems, in particular expansion-based enforcement, and enforcement based on local updates [85; 43].

In several cases enforcement is, computationally speaking, straightforward if the task consists in checking whether there exists a modified AF that enforces a set of arguments under certain parameters. For instance, extension enforcement under normal expansions for admissible semantics is always possible if the set E to enforce is conflict-free in the given AF (see Section 3.2.1). That is why we look at extension enforcement that aims at minimizing the change induced by an enforcing AF. Concretely, given an AF $F = (A, R)$ we aim at finding an enforcing AF $G = (A', R')$ such that the distance $d(F, G)$ between them is minimal (see Definition 3.18).

Another important aspect for expansion-based enforcement is how many arguments shall be added. That is, if $G = (A', R')$ is an expansion of $F = (A, R)$, how to confine $|A'| - |A|$? This is important from a computational perspective, since allowing for unbounded expansions may complicate computation. We consider here only bounded expansions.

We define the computational problems next, for extension enforcement under bounded expansions and local updates. For local updates no bound is needed, since if the number of arguments $|A|$ does not change, the number of modifications to R is bounded quadratically by $|A|$.

For studying complexity of problems that are inherently optimization problems, such as enforcement when the goal is to find an enforcing AF with a minimum number of modifications to the attack structure, there are several ways to formally approach such problems. One standard way to reveal inherent complexities of optimization problems is to consider a natural decision variant: for a given integer $k \geq 0$, we ask whether there is an enforcing AF with at most k many modifications. We note that another way to study complexity of optimization problems is to consider functional problems instead of decision problems, which is an approach that may give more detailed complexity results (see, e.g., [64]). However currently no such analysis was carried out for enforcement.

First, we define a decision problem for extension enforcement under bounded expansions.

Extension enforcement under bounded expansions
Instance: an AF $F = (A, R)$, $E \subseteq A$, set A', integer $k \geq 0$, and a semantics σ.
Question: Does there exist an expansion $G = (A \cup A', R')$ of F such that $\exists E' \in \sigma(G)$ with $E \subseteq E'$ and $d(F, G) \leq k$?

In more words, given an AF F, a set $E \subseteq A$ of arguments to enforce, a set of arguments A', an integer $k \geq 0$ and a semantics σ, the task is to decide whether there exists an expansion G of F that enforces E non-strictly under σ, and, moreover, makes at most k many modifications to the attack structure. Note that the expansion G is bounded in the sense that the expanded arguments are already given beforehand, i.e., G has $A \cup A'$ as its arguments. The above definition gives a decision problem for non-strict enforcement. As before, we define strict enforcement analogously by replacing $\exists E' \in \sigma(G)$ and $E \subseteq E'$ with $E \in \sigma(G)$.

Next, we look at a decision problem variant for extension enforcement under local updates.

Extension enforcement under local updates
Instance: an AF $F = (A, R)$, $E \subseteq A$, integer $k \geq 0$, and a semantics σ.
Question: Does there exist a local update $G = (A, R')$ of F such that $\exists E' \in \sigma(G)$ with $E \subseteq E'$ and $d(F, G) \leq k$?

Strict enforcement is again defined as above.

We consider as fragments of these two enforcement problems those sub problems where a semantics is fixed, i.e., extension enforcement under bounded expansions (local updates) under a specific semantics σ, instead of having σ as part of the instance.

Finally, before delving into complexity results from the literature, we provide background on complexity classes used here, and related problems useful to understanding complexity of enforcement. For thorough introductions to computational complexity see, e.g., [3; 77]. We assume that the reader is familiar with concepts like complexity classes, reductions, completeness, and oracles. Complexity class P is composed of all decision problems which can be decided in polynomial time by a deterministic algorithm. Class NP contains all decision problems that can be decided by a non-deterministic polynomial time algorithm. Class coNP contains all problems that are complementary to a problem in NP. Class Σ_2^P contains all decision problems which can be decided via a non-deterministic polynomial time algorithm that can access an NP oracle. Class Π_2^P contains all problems that are complementary to some problem in Σ_2^P.

Two reasoning tasks on AFs in a static, i.e., non-dynamic setting, are useful to understand the complexity of enforcement. The first one is usually referred to as the Verification problem.

Verification
Instance: an AF $F = (A, R)$, $E \subseteq A$, and a semantics σ.
Question: Does $E \in \sigma(F)$ hold?

That is, the task is to check whether a given set E is a σ-extension. Another useful problem is credulous acceptance of arguments in AFs.

Credulous acceptance
Instance: an AF $F = (A, R)$, $a \in A$, and a semantics σ.
Question: Does $\{a\} \in \mathit{cred}(F, \sigma)$ hold?

In words, an argument is credulously accepted in case there is a σ-extension of a given AF that contains the queried argument.

Complexity of verification and credulous acceptance was established; we summarize complexity results for the main semantics in Table 2.

semantics σ	verification	credulous acceptance
cf	in P	in P
ad	in P	NP-complete
co	in P	NP-complete
stb	in P	NP-complete
pr	coNP-complete	NP-complete

Table 2: Complexity of verification and credulous reasoning in AFs (for an overview see [57])

3.3.1 Complexity of Enforcement

We illustrate two ways of showing complexity bounds that turn out to be tight in many, but not all, cases.

For an upper bound (i.e. membership in a complexity class), consider the following non-deterministic algorithm (sketch) given an AF F, a set E to enforce, and a semantics σ:

1. non-deterministically construct an AF $G = (A', R')$ that is a bounded expansion (or local update) of F;
2. non-deterministically construct an $E' \subseteq A'$ (for non-strict enforcement only); and
3. verify whether $E' \in \sigma(G)$ and $E \subseteq E'$ (for non-strict enforcement) or $E \in \sigma(G)$ (for strict enforcement).

In case the last step succeeds, then it holds that G enforces E to be a σ-extensions (non-)strictly. As can be seen from this algorithm sketch, a complexity upper bound can be derived from the complexity of the verification problem under σ. Take $\sigma = ad$, i.e., the verification problem under admissibility which is polynomial-time decidable. It follows that extension enforcement under bounded expansions (resp. local updates) is in NP under admissibility. The reason is that the above algorithm sketch directly witnesses membership in NP: one (resp. two) non-deterministic construction(s) and a check in polynomial time show membership for $\sigma = ad$. In the non-deterministic construction of the above algorithm the bound on the expansion is crucial, otherwise a non-bounded, and thus potentially non-polynomially bounded, structure would be constructed. However, this does not imply that enforcement under non-bounded expansions requires large expansions.

There is a similar approach to show lower bounds. Here we distinguish more between strict and non-strict variants. In particular, extension enforcement under bounded expansions (resp. local updates) and σ is C-hard

- if verification under σ is C-hard and the enforcement variant is strict; or
- if credulous acceptance under σ is C-hard and the enforcement variant is non-strict.

The underlying reason is as follows. One can reduce the verification problem to strict extension enforcement and the credulous acceptance problem to non-strict extension enforcement.

For the verification problem under σ, i.e., given an AF F and a set E, consider the extension strict enforcement problem under σ with F, E, and $k = 0$ as input (and $A' = \emptyset$ for expansion-based). Then we are not allowed to make modifications to F, and, therefore, F enforces E to be a σ-extension iff $E \in \sigma(F)$ iff this is a positive instance of the verification problem.

Similarly, the credulous acceptance problem under σ, with F and an argument a as input, is reduced to an instance of non-strict extension enforcement with input F, $E = \{a\}$, and $k = 0$ (and again $A' = \emptyset$). It follows that F enforces E non-strictly if there is an $E' \supseteq E$ with $E' \in \sigma(F)$, implying a positive instance of the credulous acceptance problem.

In several cases the two approaches to show upper and lower bounds yield tight bounds. However, there are notable exceptions.

Let us look first at results obtained for enforcement under bounded expansions, see Table 3. In this case only the non-strict variant was studied [85]. It can be observed that the complexity of this enforcement variant matches complexity of credulous reasoning in static AFs, i.e., the above approaches to show complexity bounds directly result in tight bounds. We remark that complexity of enforcement under conflict-free sets was not presented in [85], however it can be straightforwardly obtained: if the set is conflict-free then enforcement is trivial (and can be checked in polynomial time by scanning the input AF), otherwise, if the given set to enforce is conflicting, no expansion can remove such conflicts, and enforcement is impossible. Since (by definition) any conflict-free set is included in some naive extension, this result also holds for $\sigma = na$.

semantics σ	non-strict
cf	in P
na	in P
ad	NP-c
co	NP-c
stb	NP-c
pr	NP-c

Table 3: Complexity of non-strict extension enforcement under bounded expansions [85]

Let us turn to complexity of enforcement under local update [85; 43], summarized in Table 4. We see that complexity of non-strict enforcement, again, has the same complexity as credulous reasoning in static AFs, except for grounded semantics. Before discussing grounded semantics, let us turn to strict enforcement first.

To some extend surprising are the results for strict extension enforcement, which diverge from non-strict enforcement. For instance, for both admissible and stable semantics strict extension enforcement under local expansions is decidable in polynomial time. The underlying reason is that if E is to be an admissible set (a stable extension), then all conflicts inside the set have to be removed, and each attack

semantics σ	strict	non-strict
cf	in P	in P
na	in P	in P
ad	in P	NP-c
co	NP-c	NP-c
gr	NP-c	NP-c
stb	in P	NP-c
pr	Σ_2^P-c	NP-c
sst	Σ_2^P-c	Σ_2^P-c
stg	coNP-hard and in Σ_2^P	Σ_2^P-c

Table 4: Complexity of extension enforcement under local updates [85; 43]

from outside countered (each argument outside attacked). The latter means that one can choose an argument inside E to counter non-attacked attackers (to achieve defense) or remove the incoming attack. In both cases, it is sufficient to make at least one modification, however one modification is sufficient: adding an attack to counter an attacker (removing an incoming attack might not be sufficient if there are more incoming attacks). For stable semantics, similarly, one attack on unattacked arguments outside E is both necessary and sufficient, only the origin in E is flexible. Overall, this procedure sketches a polynomial-time deterministic algorithm (one can impose an ordering on arguments to make the choice of attacking arguments deterministic).

Finally, let us look at grounded semantics, for which NP-completeness was established for both non-strict and strict extension enforcement under local updates and complete semantics for the strict variant. Recall that both verification and credulous acceptance under grounded semantics is in P, and also verification for complete semantics is in P (Table 2). This means, the lower bounds established by the algorithms above do not result in tight bounds. The intuition behind this "complexity jump" for the strict variant under complete and grounded semantics is that when enforcing some set of arguments E to be complete, one has to be careful about what E defends. That is, enforcing E to be admissible is not the underlying reason for NP hardness, but to avoid having arguments defended that one desires to avoid being defended (as specified by strict enforcement, nothing outside the set E may be defended by E). In brief, addition or removal of attacks can make E admissible, but implying further arguments being defended. Finding an optimal assignment that accomplishes both having E admissible and nothing outside E being defended by E faces non-deterministic choices. However, the hardness construction to show NP-hardness is somewhat involved.

Finally, for grounded semantics and non-strict enforcement, the intuition for NP-completeness is a bit more direct: there could be a place in the AF to modify such that the grounded extension is significantly expanded and includes the desired E. However, choosing an adequate place in such a way is not direct to find.

Further semantics have been analyzed in [85].

3.3.2 Declarative Algorithms

Main approaches to compute optimal enforcing for AFs rely on declarative programming paradigms based on constraints, in particular maximum Satisfiability (MaxSAT) [68], answer set programming (ASP) [69; 58], and pseudo Boolean optimization (particularly integer linear programming [80]).

We present here some of the main ideas for algorithmic approaches to extension enforcement, focusing on the MaxSAT approach [85]. Enforcement via pseudo Boolean optimization is presented in [43], and via ASP in [74]. Systems using the MaxSAT approach are presented in [73; 43]. We present here encodings and an algorithm for some semantics, for further semantics and details we refer to the original papers.

We briefly recall background on MaxSAT. A literal is either a positive Boolean variable x or a negated Boolean variable $\neg x$. A clause is a disjunction of literals $l_1 \vee \cdots \vee l_n$ and a propositional formula is in conjunctive normal form (CNF) if the formula $\pi = c_1 \wedge \cdots \wedge c_m$ is a conjunction of clauses. Whenever convenient, we will view clauses as a set of literals and a formula in CNF as a set of clauses.

A truth assignment τ assigns either true (1) or false (0) to the Boolean variables. As usual, a truth assignment τ satisfies a variable x if $\tau(x) = 1$. Satisfaction is extended in the usual way to compound formulas, e.g., τ satisfies a literal l if $\tau(x) = 1$ and $l = x$ or $\tau(x) = 0$ and $l = \neg x$. A clause is satisfied by τ if at least one literal of the clause is satisfied, and a formula in CNF is satisfied if each clause is satisfied.

An instance of the partial MaxSAT problem is a pair $\phi = (\phi_h, \phi_s)$ with both ϕ_h and ϕ_s Boolean formulas in CNF (sets of clauses). The former is the set of hard clauses, while the latter is the set of soft clauses. A truth assignment τ is a solution to the partial MaxSAT instance if τ satisfies ϕ_h (the hard clauses). The cost of τ w.r.t. the instance ϕ is $\texttt{cost}(\phi, \tau) = \sum_{c \in \phi_s} 1 - \tau(c)$, i.e., the number of soft clauses not satisfied. A solution τ to ϕ is optimal if there is no solution τ' to ϕ with $\texttt{cost}(\phi, \tau') < \texttt{cost}(\phi, \tau)$. We refer to partial MaxSAT simply as MaxSAT.

We focus on an illustration of a MaxSAT approach to extension enforcement on the variant with local updates. Encoding extension enforcement under local updates can be achieved by encoding an AF, possible modifications, and semantics in MaxSAT. Before delving into encoding this enforcement variant, we recall an encoding of admissible semantics of static AFs similar as in [26]. Given an AF $F = (A, R)$ we define

$$\phi_{cf}(F) = \bigwedge_{(a,b)\in R} \neg a \vee \neg b.$$

We use here arguments as Boolean variables and vice versa. Satisfying truth assignments of this formula correspond directly to conflict-free sets of F in the way that $E \in cf(F)$ iff $\tau(x) = 1$ for $x \in E$ and $\tau(y) = 0$ if $y \notin E$ satisfies ϕ_{cf}. Admissibility can be encoded as follows:

$$\phi_{ad}(F) = \phi_{cf}(F) \wedge \bigwedge_{a\in A} (a \rightarrow (\bigwedge_{(b,a)\in R} (\bigvee_{(c,b)\in R} c)))$$

In words, if an argument a is in an admissible set (true in a satisfying assignment) then for each attacker b at least one defender c must be part of the admissible set (true in the assignment), as well, which directly captures the definition of admissibility.

Non-strict extension enforcement under local updates can be encoded by including variables for attacks. We first focus on how to encode constraints for the semantics, which we encode as hard clauses ϕ_h. For notation, for the encodings of conflict-free sets and admissible extensions above we used ϕ, for enforcement formulas we use ψ.

$$\psi_{cf}(F) = \bigwedge_{a,b\in A} (r_{a,b} \rightarrow (\neg a \vee \neg b))$$

In words, a new variable $r_{a,b}$ for each pair of arguments a, b is introduced denoting whether there is an attack from a to b. That is, a truth assignment includes now an assignment on the attacks, as well.

Moving on to enforcement under admissibility, which we encode as

$$\psi_{ad}(F) = \psi_{cf}(F) \wedge \bigwedge_{a,b\in A} ((a \wedge r_{b,a}) \rightarrow \bigvee_{c\in A} (c \wedge r_{c,b})).$$

That is, if a is assigned to be true and there is an attack (b, a), then this attack has to be defended against, by some c and the corresponding attack (c, b).

Let $F = (A, R)$, $E \subseteq A$ be given as an instance of the non-strict extension enforcement problem under local updates and admissibility. Defining a MaxSAT instance

$$\phi = (\psi_{ad} \wedge \bigwedge_{a \in E} a, \phi_s(F))$$

with

$$\phi_s(F) = \bigwedge_{(a,b) \in R} r_{a,b} \wedge \bigwedge_{(a,b) \notin R, a,b \in A} \neg r_{a,b}$$

results in optimal truth assignments τ to ϕ corresponding to AFs locally updated from an original AF $F = (A, R)$ with a minimum number of modifications that enforce S to be part of an admissible set. To see this, any solution to ϕ satisfies the hard clauses, implying that a truth assignment satisfying $\psi_{ad} \wedge \bigwedge_{a \in E} a$ assigns to true all variables (arguments) in E and possibly more arguments, and further assigns some of the attacks to be true (present in a modified AF) such that the argument variables assigned to true form an admissible set, thus enforcing E.

For strict extension enforcement under local updates, more variables can be fixed, since the set E to be enforced must be exactly a σ-extension, not just be part of one. That is, we can focus on variables only for attacks, since the other variables can be fixed (true for argument variables in E and false otherwise, i.e. $\bigwedge_{a \notin E} \neg a$ can be added to the hard part of the MaxSAT instance).

For instance, strict enforcement under conflict-free sets can be encoded as follows.

$$\psi^s_{cf}(F) = \bigwedge_{a,b \in E} \neg r_{a,b}$$

In more words, there cannot be any attack in the set S to be enforced, all other attacks remain unconstrained for conflict-free sets. Admissible extensions can be encoded by

$$\psi^s_{ad}(F) = \psi^s_{cf}(F) \wedge \bigwedge_{a \in E} \bigwedge_{b \in A \setminus E} (r_{a,b} \to \bigvee_{c \in E} r_{c,b}).$$

The MaxSAT instance is complete by setting

$$\phi = (\psi^s_{ad}(F) \wedge \bigwedge_{a \in E} a \wedge \bigwedge_{a \in A \setminus E} \neg a, \phi_s(F)).$$

We proceed to algorithmic approaches for problems “beyond NP”, e.g., strict extension enforcement under local updates and preferred semantics. One approach to such complex problems is to develop an algorithm that uses SAT solvers as subprocedures, and possibly calls a SAT solver multiple times (i.e., an iterative SAT-based procedure). We present an approach based (inspired by) the well-known CEGAR approach [40; 41] approach, where CEGAR stands for counterexample guided abstraction refinement. We remark that the term CEGAR is not used unambigously in the literature, and in some communities the term may refer to different concepts. Here, a CEGAR based algorithm works on an abstraction (approximation) of a solution space from which iteratively candidates are drawn. Importantly, due to the approximation, some solutions may be spurious. A SAT call determines whether a candidate is a solution or a spurious solution. In the latter case, the spurious solution is a "counterexample" which is used to refine the approximated solution space (removing as many as possible of the spurious solutions from the space) and a next candidate is produced, until a solution is reached.

For strict extension enforcement under local updates and preferred semantics, the solution space is approximated by considering initially strict extension enforcement under local updates and admissible or complete semantics. It holds that if an AF G enforces a set S strictly under preferred semantics, then G also enforces S strictly under admissible or complete semantics (since a preferred extension is complete and admissible). However, importantly, optimality is not guaranteed this way: an optimal solution to strict extension enforcement under local updates and preferred semantics might not be an optimal solution AF for admissible or complete semantics (since less modifications might be sufficient for admissible or complete semantics, but not for preferred semantics). Nevertheless, strict extension enforcement under complete or admissible semantics can act as an approximation. We focus for illustration on admissible semantics here.

The CEGAR-style algorithm is presented as Algorithm 1. When the loop is entered the first time, the MaxSAT call returns an optimal solution for strict extension enforcement under local updates and admissible semantics. To check whether the AF extractable from the truth assignment τ is a solution also under preferred semantics, we call a SAT solver to determine whether S is a preferred extension in the candidate AF. If so, we return this AF. Otherwise, we found a counterexample and the abstraction is refined. For correctness of the overall algorithm, it is important that a refinement does not remove (all) optimal solutions AFs for strict enforcement under local updates and preferred semantics. We refine here by removing the solution found in the MaxSAT call, i.e., by removing exactly τ from consideration when looking for the next candidate. This is a straightforward refinement. More sophisticated refinements are possible, but require care when designing them (e.g., in order not to violate correctness) [70]. For instance, in some cases refinements can be based on foundational results whether changes on AFs induce changes on semantics [31; 30].

The definitions for Algorithm 1 are as follows. From a truth assignment τ we can extract an AF by $\textsc{Extract}(A, \tau) = (A, R)$ with $R = \{(a, b) \mid \tau(r_{a,b}) = 1\}$. The

Algorithm 1 Strict extension enforcement under local updates and preferred semantics

```
1: $\varphi_h \leftarrow \psi'_{ad}(F)$
2: $\varphi_s \leftarrow \bigwedge_{(a,b)\in R} r_{a,b} \wedge \bigwedge_{(a,b)\notin R, a,b\in A} \neg r_{a,b}$
3: while true do
4:     $\tau \leftarrow$ MaxSAT$(\varphi_h, \varphi_s)$
5:     $result \leftarrow$ SAT(Check$(\tau, S)$)
6:     if $result = unsatisfiable$ then return $\tau$
7:     else $\phi_h \leftarrow \phi_h \wedge$ Refine$(\tau)$
8: end while
```

formula

$$\textsc{Check}(\tau, E) = \phi_{ad}(F') \wedge \bigwedge_{a\in S} a \wedge \bigvee_{b\in A\setminus S} b$$

can be used for checking whether an admissible extension E is a preferred extension of the modified AF: we guess a superset and check admissibility by the above sub formulas. Refinement is specified via

$$\textsc{Refine}(\tau) = \bigvee_{(a,b)\in R'} \neg r_{a,b} \vee \bigvee_{(a,b)\in(A\times A)\setminus R'} r_{a,b}.$$

In words, we exclude in a subsequent search for a solution candidate exactly the currently found candidate AF.

Example 3.27. *Consider the AF F from Figure 8(a). That is, we have a chain of attacked arguments from a to b to c, and c attacks both d and e. Further, f is unattacked and does not attack an argument. The unique preferred extension of F is $\{a, c, f\}$. Say we want to strictly enforce $\{b, f\}$ to be exactly a preferred extension, and use Algorithm 1 in order to achieve that. Initially, we solve, via MaxSAT, strict enforcement under admissible semantics to have $\{b, f\}$ being admissible. Say the result is as shown in Figure 8(b), i.e., an attack from f to a is added, resulting in $\{b, d, e, f\}$ being the unique preferred extension, and $\{b, f\}$ being admissible. As the SAT solver call in the algorithm verifies, this AF candidate is not a solution, since, e.g., $\{b, d, e, f\}$ is an admissible superset of $\{b, f\}$, implying that $\{b, f\}$ is not preferred. We exclude, via the refinement step, this candidate AF, and call the MaxSAT solver again. Note that the hard clauses refute the previous candidate AF (i.e., any truth assignment simulating that AF does not satisfy the hard clauses).*

In the next steps of the algorithm, all AFs that enforce $\{b, f\}$ to be admissible are checked which make at most one modification (in the above simple refinement).

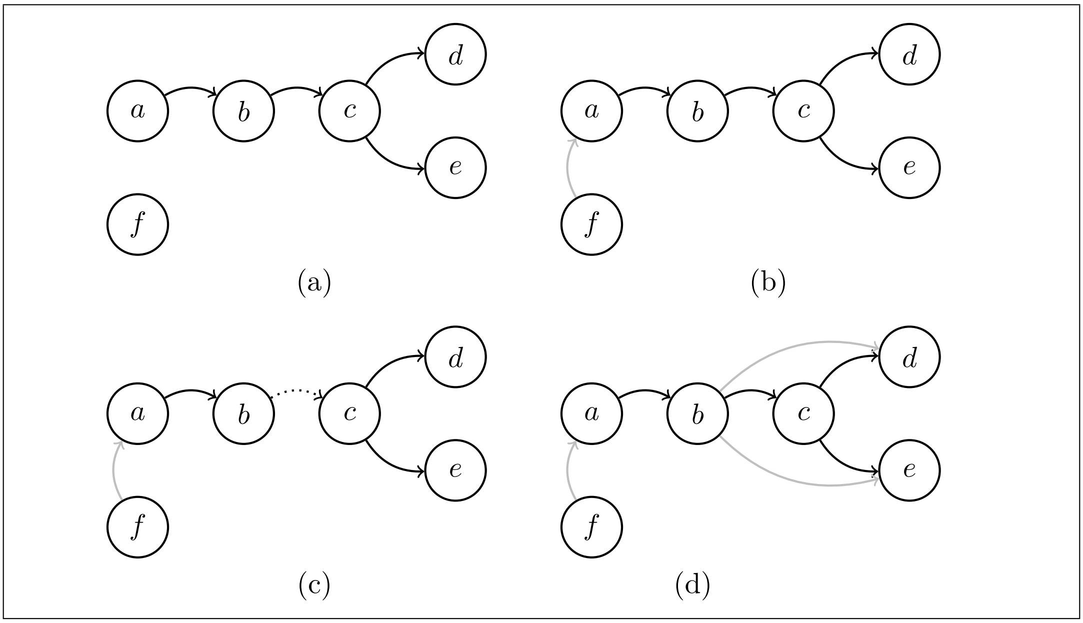


Figure 8: Example candidate AFs for Algorithm 1

After that it is verified that no modified AF with at most one modification (local update) achieves the strict enforcement under preferred semantics.

For two possible modifications, say the MaxSAT call returns an assignment corresponding to the AF in Figure 8(c). Due to the definition of the MaxSAT instance, we know that $\{b, f\}$ *is admissible in this candidate AF, which is like the previous one, except for removal of the attack from b to c. Here* $\{b, f\}$ *is again admissible, and the unique preferred extension is* $\{b, c, f\}$*, which is again verified not to be a solution. After checking all AFs that enforce* $\{b, f\}$ *to be admissible with at most one modification, the algorithm proceeds to at most three modifications, where a possible solution can be found, as illustrated in Figure 8(d).*

4 Related Notions to Enforcement

In this section, we overview several notions that are closely related to the enforcement setting described previously.

4.1 Update Using Logical Translations

YALLA (Yet Another Logic Language for Argumentation) [45] is a first-order logical language that allows to describe argumentation frameworks and their semantics. Then, operations related to enforcement can be defined through belief change theory, especially belief update [62].

Let us briefly describe the syntax and semantics of YALLA formulas. It is assumed that argumentation frameworks are built from a given *universe* $F_U = (A_U, R_U)$. This means that for any AF $F = (A, R)$, $A \subseteq A_U$ and $R \subseteq R_U \cap (A \times A)$. We write $k = |A_U|$ the number of arguments in the universe. A YALLA formula (or more precisely, YALLA$_U$) is a well-formed first order logic formula such that:

- the set of constant symbols is $V_{const} = \{c_\perp, c_1, \ldots, c_p\}$ where $p = 2^k - 1$;
- the set of function symbols is $V_{func} = \{\text{union}^2\}$;
- the set of predicate symbols is $V_{pred} = \{\text{on}^1, \rhd^2, \subseteq^2\}$.

The semantics of YALLA is defined through a structure associated with an AF $F = (A, R)$ built on the universe F_U. The domain of this structure is $D = 2^{A_U}$, and it is associated with an interpretation such that:

- the constant symbol $c_\perp$ is associated with the empty set; each constant symbol c_i ($i \in \{1, \ldots, 2^k - 1\}$) is associated with a different non-empty element of D;
- the union function symbol is associated with the binary set-theoretic union over D;
- the on predicate symbol is associated to the characterization function of subsets of A, i.e. on(S) is true if and only if $S \subseteq A$;
- the predicate symbol $\rhd$ is associated with the set-attack relation induced by R, i.e. $S_1 \rhd S_2$ if and only if $S_1 \subseteq A$, $S_2 \subseteq A$, and $\exists a_1 \in S_1, a_2 \in S_2$ such that $(a_1, a_2) \in R$;
- the predicate symbol $\subseteq$ is associated with the classical inclusion relation over D.

Some axioms are added to the theory in order to guarantee the meaning of the YALLA formulas. For instance, if a set S_1 is included in A, then any subset of S_1 is included in A as well: this is formalized by $\forall x, y, (\text{on}(x) \land y \subseteq x) \Rightarrow \text{on}(y)$. A full description of the YALLA axioms is out of the scope of this chapter; we refer the interested reader to [45] for more details on this topic.

An argumentation framework $F = (A, R)$ can be described with the formula

$$\Phi_F = \mathrm{on}(A) \land \bigwedge_{x \in A_U \setminus A} \neg\, \mathrm{on}(\{x\}) \land \bigwedge_{(x,y) \in R} (\{x\} \rhd \{y\}) \land \bigwedge_{(x,y) \in R_U \setminus R} \neg(\{x\} \rhd \{y\})$$

Then, the principles underlying extension-based semantics can also be encoded as YALLA formulas. Given the structure associated with an AF $F = (A, R)$,

- the term t is conflict-free if the formula $\Phi^{cf}_t = \mathrm{on}(t) \land \neg(\{t\} \rhd \{t\})$ is valid;
- the term t_1 defends the term t_2, denoted by $t_1 \rhd\!\rhd t_2$, if the formula $\forall t_3, ((\mathit{singl}(t_3) \land t_3 \rhd t_2) \to (t_1 \rhd t_3))$ is valid, where $\mathit{singl}(t)$ is a formula that is valid if t is a singleton.

The combination of these formulas allows to characterize the admissible sets (i.e. the terms that satisfy of $\Phi^{ad}_t = \Phi^{cf}_t \land (t \blacktriangleright t))$. This is the basics of YALLA encoding for the classical Dung's semantics. Additional constraints in the formulas yield encodings Φ^{σ}_t for the other semantics.

Then, belief update rationality postulates and operators [62] are adapted to take into account the universe $F_U = (A_U, R_U)$. A set of authorized transitions (corresponding to what we call a modification type) is $T \subseteq \Gamma_U \times \Gamma_U$, where Γ_U is the set of all AFs built on the universe F_U. Then, roughly speaking, an update operator $\diamond_T$ is such that, if ϕ is a YALLA formula characterizing an AF F, then for any formula α, $\phi \diamond_T \alpha$ characterizes an AF G such that $(F, G) \in T$.[6]

Finally, enforcing an extension in an AF F can be achieved by updating the formula Φ_F:

$$\Phi_F \diamond_T \Phi^{\sigma}_{c_i}$$

characterizes the AFs that enforce S_i in F, under the modification type T and the semantics σ, where c_i is the YALLA constant symbol that corresponds to the set of arguments S_i, and Φ^{σ}_t is valid if and only if the term t corresponds to a σ-extension (similarly to the way Φ^{cf}_t, described previously, characterizes conflict-free sets).

Another logic-based approach is that of [48], which proposes to translate the argumentation framework and the semantics into logic, to perform the enforcement. In this case, the Dynamic Logic of Propositional Assignments (DL-PA) by [5], is used to represent update operators as executable programs. The piece of information which causes the update is a formula about acceptance statuses, which should be satisfied by at least one extension of the result (credulous enforcement of the formula) or by each extension of the result (sceptical enforcement of the formula).

[6] This is actually slightly more subtle than that, since YALLA formulas can characterize sets of AFs.

Forbus' update operator is used to change minimally the attack relation such that the extensions of the new argumentation framework comply with the expected enforcement. An extension of [48] is proposed by [50], which considers also addition and removal of arguments, and by applying the framework to an access control case. Then, [49] generalizes the previous two approaches.

Let us mention that these kinds of approaches based on a belief update operation allow richer forms of enforcement, since complex information about the sets of arguments and the attacks in the AFs can be described. Also, other kinds of belief change operations (e.g. belief revision [63] or belief contraction [36]) could be defined in these contexts. We refer the interested reader to [53] for more details on the relation between belief change and argumentation.

4.2 Status Enforcement

Status enforcement [72] is defined as an operator where two sets of arguments are provided as input, that must be respectively positively and negatively enforced. This operation does not fit the framework described previously, since it is supposed that there is only one set of arguments given in input, that must have exactly one acceptance status with respect to some reasoning mode (see Definition 3.1).

Formally, given an AF $F = (A, R)$, P and N two subsets of A such that $P \cap N = \emptyset$, and σ a semantics,

- the AF $G = (A, R')$ is a credulous status enforcement of (P, N) in F with respect to σ if $P \subseteq \bigcup \sigma(G)$ and $N \cap \bigcup \sigma(G) = \emptyset$;
- the AF $G = (A, R')$ is a sceptical status enforcement of (P, N) in F with respect to σ if $P \subseteq \bigcap \sigma(G)$ and $N \cap \bigcap \sigma(G) = \emptyset$.

In words, status enforcement consists in finding G such that every argument in P is credulously (respectively sceptically) accepted in G, and every argument in N is not credulously (respectively sceptically) accepted in G.

Complexity issues for optimal status enforcement, i.e. finding G such that $d(F, G) = |(R \setminus R') \cup (R' \setminus R)|$ is minimal, have been investigated by [72]. Similarly to complexity for optimal extension enforcement (Section 3.3), the complexity results concern a decision problem related to the optimization problem under consideration.

Credulous status enforcement
Instance: an AF $F = (A, R)$, $P \subseteq A$ and $N \subseteq A$ s.t. $P \cap N = \emptyset$, integer $k \geq 0$, and a semantics σ.
Question: Does there exist an AF $G = (A, R')$ such that $P \subseteq \bigcup \sigma(G)$ and $N \cap \bigcup \sigma(G) = \emptyset$ and $d(F, G) \leq k$?

	$N = \emptyset$		N unrestricted	
semantics σ	credulous	sceptical	credulous	sceptical
cf	in P	trivial	in P	trivial
ad	NP-c	trivial	Σ_2^P-c	trivial
co	NP-c	NP-c	Σ_2^P-c	NP-c
gr	NP-c	NP-c	NP-c	NP-c
stb	NP-c	Σ_2^P-c	Σ_2^P-c	Σ_2^P-c
pr	NP-c	Σ_3^P-c	Σ_2^P-c	Σ_3^P-c

Table 5: Complexity of status enforcement

Sceptical status enforcement
Instance: an AF $F = (A, R)$, $P \subseteq A$ and $N \subseteq A$ s.t. $P \cap N = \emptyset$, integer $k \geq 0$, and a semantics σ.
Question: Does there exist an AF $G = (A, R')$ such that $P \subseteq \bigcap \sigma(G)$ and $N \cap \bigcap \sigma(G) = \emptyset$ and $d(F, G) \leq k$?

Two cases are considered: the general case, and the restricted case where $N = \emptyset$ (i.e. only positive arguments must be enforced). Table 5 presents the complexity of these problems for various semantics.

MaxSAT and CEGAR based algorithms in the same spirit as algorithms for extension enforcement (Section 3.3.2) are also provided.

4.3 Control Argumentation Frameworks

Now we introduce a concept that can be interpreted as a variant of enforcement under uncertain information. Control Argumentation Frameworks (CAFs) [46] are AFs where arguments and attacks are split in three distinct parts:

- the fixed part is made of arguments and attacks that are unquestionably in the system;
- the uncertain part is made of arguments and attacks that may belong to the system, as well as “undirected" attacks: in this case there is for sure a conflict between arguments, but the actual direction is uncertain;
- the control part is made of arguments and attacks that may be used by the agent.

The sets of fixed, uncertain and control arguments are disjoint, as well as the various sets of attacks. Roughly speaking, the fixed part corresponds to certain knowledge, i.e., elements that cannot be influenced neither by the agent nor by its environment (we use “environment" in a wide sense, it also includes other agents). On the contrary, the uncertain part models the agent’s knowledge (and beliefs) about the environment (and the other agents); in realistic scenarios, this knowledge is by nature uncertain. Finally, the control part corresponds to the agent’s possible actions. When the agent selects a subset of the control arguments and attacks (called a configuration), then it defines a *configured CAF*, that is the same CAF where the control arguments (and the associated attacks) that have not been selected have been removed. The uncertain part of the CAF induces a set of completions, i.e. classical AFs that are compatible with the knowledge encoded in the CAF. This notion is borrowed from Incomplete Argumentation Frameworks [42; 23; 25].

The notion of controllability of a CAF, with respect to a given target set of arguments, is directly related to enforcement. This target is defined as a subset of the fixed arguments, that is expected to belong to each (or some) extension of each completion. The agent needs to find a configuration that reaches this target. Let us exemplify these concepts.

Example 4.1. *Figure 9 describes a CAF, where the set of fixed arguments is $\{f_1, f_2, f_3, f_4, f_5\}$, the only uncertain argument is u (dashed square argument), and the control arguments are $\{c_1, c_2, c_3\}$ (bold square arguments). The plain arrows represent fixed attacks (e.g. (f_2, f_1) is fixed); the dotted arrow (f_5, f_1) means that it is uncertain whether f_5 actually attacks f_1 or not; the symmetric dashed arrow (u, f_4) means that there is for sure a conflict between u and f_4, but the actual direction is uncertain (it could be (u, f_4), or (f_4, u), or both at the same time). Finally, the bold arrows represent control attacks, they are related to the control arguments that can be selected by the agent.*

We suppose that the target $T = \{f_1\}$ must belong to each stable extension. Without control arguments, this is not possible: there are, for instance, completions where f_5 attacks f_1, and in this case f_1 is not defended. However, with the configuration $\{c_1, c_3\}$, f_1 will be defended against every possible threat coming from the uncertain part: c_1 defends f_1 against f_5, and c_3 defends f_1 against u (that is an undirect threat, since u may defeat f_4, making then f_2 and f_3 acceptable). Similarly, $\{c_2\}$ is a valid configuration, since it allows to guarantee that $\{f_1\}$ is included in every stable extension of every completion.

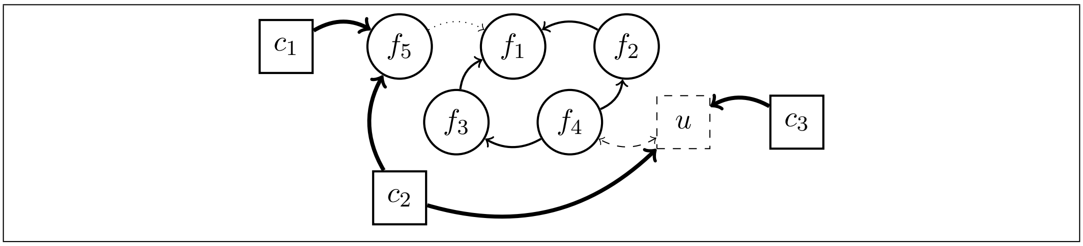


Figure 9: An example of control argumentation framework

Controlling a CAF can be seen as enforcing (non-strictly) an extension in presence of uncertainty. Intuitively, for an AF $F = (A, R)$ and a set of arguments E to be enforced through strong expansion, we can define a CAF that is controllable with respect to E if and only if it is possible to enforce E in F. Indeed, the arguments A and attacks R correspond to the fixed part of the CAF, while the uncertain part is empty. Then, for each $a \in A$, a control argument c_a with a control attack (c_a, a) is added. If E can be enforced in F, then the CAF is controllable (where the configuration to be chosen consists in the set of control arguments that do not attack E). On the opposite, if E cannot be enforced through a strong expansion, then the CAF is not controllable: indeed, the CAF configured by a control configuration is a strong expansion of F, thus E cannot be accepted in this configured CAF. We give a simple example of this transformation.

Example 4.2. *Let $F = (A, R)$ be the AF given at Figure 10a. We consider the grounded semantics: $gr(F) = \{\emptyset\}$. Let $E_1 = \{a\}$ be a set of arguments to be (non-strictly) enforced through a strong expansion. This enforcement is possible: for instance, the AF G that is a strong expansion of F where a new argument attacks b yields the expected result. Such an AF G corresponds to the CAF (given at Figure 10b) after it has been configured by $\{c_b\}$ (i.e. the argument c_a and the attack (c_a, a) are removed). So we observe that this CAF is controllable with respect to E_1 and the grounded semantics. On the opposite, $E_2 = \{a, b\}$ cannot be enforced in F with strong enforcement (since it is not conflict-free), and similarly there is no way to configure the CAF with respect to E_2 and the grounded semantics.*

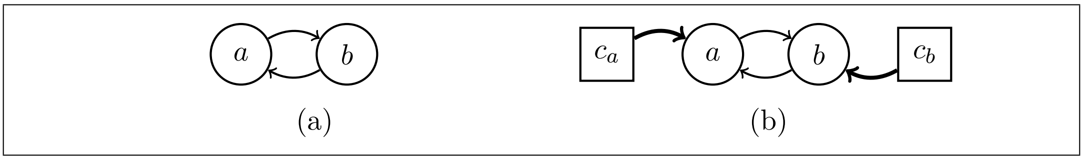


Figure 10: Transforming an AF into a CAF

In the previous example, we show how non-strict enforcement under strong expansion can seen as controlling a CAF. But more generally, since control arguments can attack each others, non-strict enforcement under a normal expansion can also be “translated" in controlling a CAF. On the opposite, configuring a CAF for controlling a target T can be interpreted as enforcing T in all the completions of the CAF with the same normal expansion (where the added arguments and attacks are chosen in the control part).

Let us also briefly mention that detailed complexity results and algorithms for reasoning with CAFs have been provided in [71], and [66] defines a weaker form of controllability, that relies on one completion instead of the whole set.

Applying CAFs to Automated Negotiation Let us briefly described how enforcement (or more precisely, CAFs) has been used in a context of automated negotiation [47]. The idea is to represent the (uncertain) knowledge of an agent about her opponent with a CAF. Indeed, negotiation has more chance to reach an agreement if agents have some knowledge about each other; however it is unrealistic to consider that opponent modelling can be done without incomplete or uncertain information. The theory of a negotiating agent is thus made of two parts: a classical AF that represents the agent's personal knowledge, and a CAF that represents her knowledge about her opponent.

It is supposed that agents negotiate about a set of (mutually exclusive) offers $\mathcal{O}$. Each offer may be supported, in AF_1 (the personal knowledge of agent 1), by 0, 1 or several *practical arguments*, i.e. arguments whose conclusions correspond to actions or decisions. The other arguments are *epistemic arguments*, they support knowledge and beliefs. The knowledge of agent 1 about agent 2 is represented in CAF_1^2. The fixed and uncertain parts are supposed to be built from the actual AF of agent 2: the assumption is made that there can be uncertainty (represented in the CAF), ignorance (some arguments or attacks of agent 2 may not appear in the CAF at all), but no mistake (there is no attacks or arguments that appear in CAF_1^2 but not in the personal AF of agent 2). Finally, the control part of CAF_1^2 is made of arguments and attacks chosen in AF_1, that are supposed to be used by agent 1 in order to make its target accepted. Similarly, AF_2 is the personal knowledge of agent 2, and CAF_2^1 represents the (uncertain) knowledge of agent 2 about agent 1.

Each agent selects its preferred offer $o \in \mathcal{O}$ according to its personal knowledge: o has to be supported by a practical argument that is accepted in AF_1; if several offers can be chosen, an assumption is made that the agent has a preference ranking over offers. When the preferred offer o of agent 1 is chosen, she uses her knowledge about agent 2 in order to persuade her to accept o: agent 1 searches for a practical argument in CAF_1^2 that supports o. If such an argument a exists, then three options are possible:

- if a is accepted in each completion of CAF_1^2 without using any control arguments, then agent 1 makes an offer to agent 2 (offer o, supported by argument a);
- otherwise, if a is accepted with the use of some control arguments $c_1, \ldots, c_k$, then agent 1 can again make an offer (offer o, supported by argument a, that is accepted because of $c_1, \ldots, c_k$);
- in the last case, a is not accepted even with control arguments, then agent 1 searches for another argument that supports offer o in CAF_1^2.

In the first two cases, if agent 2 accepts the argument a (with, or without control arguments), then the negotiation is a success: offer o is accepted. Otherwise, agent 2 gives to agent 1 the reasons why she rejects a (for instance, she knows some arguments that agent 1 does not know). If agent 1 knows other arguments that support o in CAF_1^2, the process is repeated. Otherwise, this is the end of the round: agents switch their roles, and now agent 2 will choose her preferred offer o', and use her CAF in order to persuade agent 1 to accept o'.

The whole process goes on, until either the agents agree on some offer (in that case, the negotiation is a success), or they do not have available offers (the negotiation fails).

Let us illustrate the process, with an example borrowed from [47].

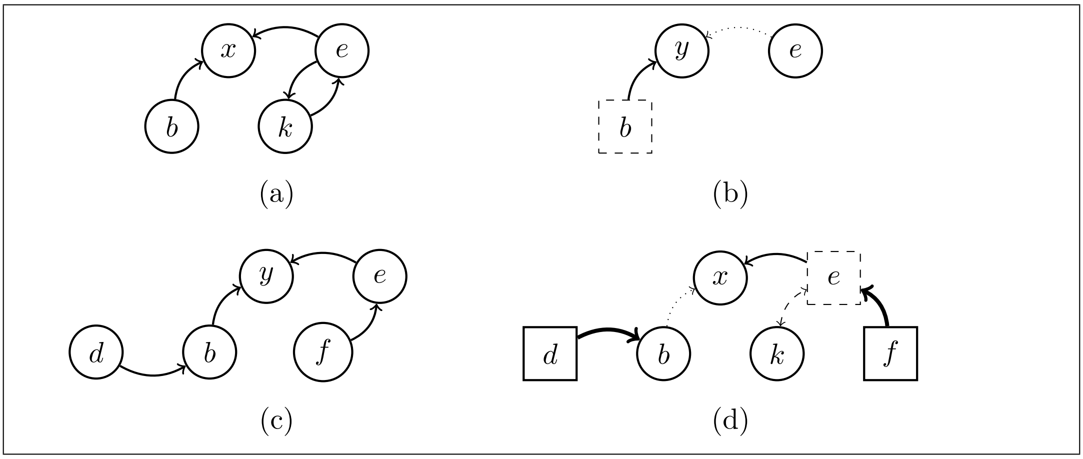


Figure 11: Initial theories of agents 1 and 2

Example 4.3. *Figure 11 describes the negotiation theories of two agents. More precisely, the AFs AF_1 and AF_2 (respectively Figure 11a and Figure 11c) correspond to the personal knowledge of (respectively) agent 1 and agent 2, while CAF_1^2 (Figure 11b) represents the (uncertain) knowledge of agent 1 about agent 2, and vice-versa for CAF_2^1 (Figure 11d). We suppose that both agents use the stable semantics for reasoning, and that there is one offer o, that is supported by arguments x and y. Before starting the negotiation, agent 1 has no reason to accept the offer o (since its supporting argument x is rejected in AF_1), while agent 2 accepts o since y is accepted in AF_2. If agent 1 starts the negotiation, she has no offer to propose (since there is no accepted argument in AF_1 that supports some offer), so the token has to go to agent 2.*

Agent 2 can make an offer. The goal of agent 2 is to persuade agent 1 to accept the offer o, using arguments that agent 1 already knows. This means that she needs to make agent 1 modify her AF in order to accept x (since x is the only argument that supports the offer o in CAF_2^1). This persuasion phase goes first through a step that do not use the control part of the CAF: if x is accepted in the CAF with no control argument, agent 2 can send to agent 1 the message "offer o, supported by the accepted argument x". In the present example, this is not the case: there are completions where x is rejected (for instance, the ones where the attack (b, x) exists). So, in the next step, agent 2 searches for a control configuration that allows to make x accepted in each completion. Here, the configuration is the full set of control arguments $\{d, f\}$. Agent 2 can then send the message "offer o, supported by the argument x, that is accepted because d attacks b and f attacks e".

Receiving this message triggers some updates in agent 1's knowledge. First, she can add the arguments d and f (as well as the attacks (d, b) and (f, e)) in CAF_1^2. Moreover, while argent b was initially uncertain in the CAF, it can now become a fixed argument: since agent 2 sends a message about the argument b, it certainly means that agent 2 knows this argument. Then, agent 1 can also add these arguments and attacks in her AF. The updated AF_1 and CAF_1^2 are shown at Figure 12. Since in the update AF_1, the argument x is accepted, agent 1 can stop the negotiation by accepting the offer o.

Let us suppose that agent 1 has, e.g., some argument i attacking d, then instead of accepting the offer o, she sends the message "reject the offer, because i attacks d". Then agent 2 updates her CAF, and the process continues as illustrated previously until reaching the negotiation success (if some offer can be accepted by both agents) or failure (if no offer can be accepted by both agents, even when exchanging arguments for defending them).

Experiments [47] have shown that control arguments and attacks help to increase

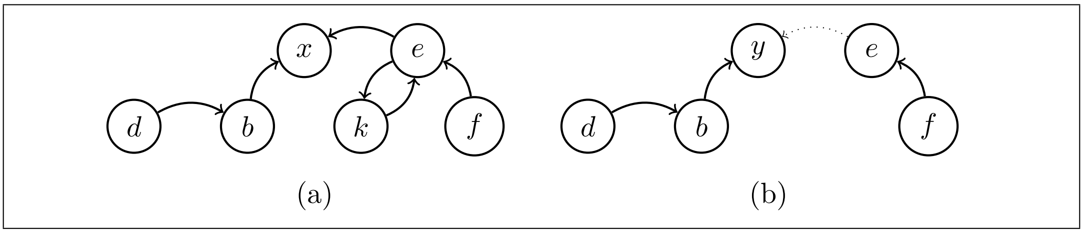


Figure 12: The updated theory of agent 1

the agreement rate, even when the percentage of uncertainty in CAFs is high.

4.4 Enforcement under Constraints

We have already seen several approaches to enforcement, and variants to enforcement. Common to many approaches are restrictions on the allowed modifications, such as allowing only expansions, local updates, other types of modifications, or defining allowed modifications via formulas, such as in YALLA.

More broadly, one can impose *constraints* on enforcements. Such constraints can have many different shapes or forms (expansions, deletions, specification as a formula, etc.). In general, constraints can be very useful for applications of enforcement operators. Going into a slightly different direction from before in this chapter, consider the following example.

Example 4.4. *Say we have knowledge about two arguments a and b, and we wish to enforce non-acceptability of a, e.g., because argument a counters a desirable argument. An expansion by c and attack (c, a) does the trick. However, say, in addition, that a is a sub argument of argument b, when inspecting the contents of the arguments. In such a case it seems adequate to require that c attacks the super argument of a, as well, i.e., we want also to add the attack (c, b).*

As suggested by the example, in some situations we may be faced with circumstances that may require specific expansions, or rather ruling out certain expansions. For instance, only considering those expansions that satisfy the condition that if an attack from some argument onto a is added, so must the same attacker also attack b.

Such conditions are not directly captured by the main types of modifications represented in this chapter, but can be incorporated into enforcement, as well. In [84], several families of constraints are considered, and the survey [53] discusses constraints of dynamics in argumentation, in general. We refer the reader for details to these papers, but highlight a particular type of constraint: implications of presence of arguments and attacks. By allowing constraints that take the form of implications, e.g., of the form mentioned above for attacks, one can specify that attacks on sub arguments must "propagate" to super arguments, which is present in many instances of formal approaches to structured argumentation [44; 67].

4.5 Other related works

Beside extension enforcement, adding or removing arguments or attacks to an AF can be seen as another form of enforcement, on the structure of the AF. This relates to dynamic aspects of argumentation. As already mentioned, [30; 38] are among the first approaches that studied the changes implied by such structural enforcements. Other similar approaches are detailed by [53].

In [82] the authors studied the question of how to *repair* an AF if nothing is credulously/sceptically accepted. More precisely, the main aim is to restore consistency via removing certain (minimal) sets of arguments or attacks. Note that enforcing a certain non-empty set can be seen as a special kind of repairing given that we are faced with no credulously accepted arguments. The notion of C-restricted semantics [20] is related to enforcement too. It can be shown that a set of arguments is a C-restricted extension if and only if it can be (non-strictly) enforced with a restricted form of expansion.

Normal expansions of AFs have been used for other purposes related to enforcement. For instance, [33] describes a framework where an agent's knowledge is represented by an AF F and a propositional formula ϕ that represents an integrity constraint about the complete labellings of the AF. The agent's knowledge is said to be inconsistent if none of the complete labellings satisfies the constraint. Two approaches are proposed for restoring consistency. The first one is a direct use of a normal expansion: the authors have proven that there exists a normal expansion of F that is consistent with ϕ (under some minimal assumption about the consistency of ϕ). The second approach also uses normal expansion, but only after a first step that consists in revising [63] the complete labellings of F by ϕ, in order to compute the so-called fallback beliefs of the agent. Then, a normal expansion allows to obtain a new AF that is consistent with the fallback beliefs. Contrary to the first approach (based only on an expansion), this one guarantees that the agent's complete labellings are as close as possible to the initial complete labellings.

Quite recently, the inverse problem to extension enforcement was studied, namely the problem of *extension removal* [18]. That is: given an AF F and a set of extensions $\mathcal{E}$, identify an AF H that is as close as possible to F but has none of the extensions in $\mathcal{E}$. In the same way as enforcement shifts revision to the level of extensions, extension removal shifts contraction to the level of extensions.

The approach by [56] aims at checking if a set E of sets of arguments can be the set of extensions of any argumentation framework F with respect to a given semantics σ. This property is named *realizability* of E with respect to σ. Realizability can be seen as a form of enforcement, where a set of extensions has to be enforced, and all the necessary structural changes on the argumentation framework (nothing is known about beforehand) can be done to achieve this.

A further related approach to enforcement is that of learning AFs or synthesis of AFs [79; 78; 75]. In brief, the aim is to construct an AF from certain information available. Different from deterministic logical approaches that construct an AF from a knowledge base (see structured argumentation approaches, e.g., in [6]), in AF learning or synthesis the information available might not uniquely determine an AF. In the AF synthesis problem [75], for instance, information about the semantics is given, and the task is to construct an AF that as best as possible matches the given semantic information. In this way, AF synthesis is related to realizability (see above), as well.

5 Conclusion

This chapter has offered an overview of the notion of enforcement in abstract, formal argumentation. A focus has been done on extension enforcement, on its general characterization, and on how it can be achieved: the various changes that can be applied to the structure of the argumentation framework, and/or to the semantics, considering that these changes should be minimal. Results about the complexity of enforcement, and algorithms, showing the feasibility of this approach, have also been presented.

If a general context and a number of specific approaches have been described, many additional proposals exist and keep on being proposed, showing the liveliness of the field. Applications of these formal approaches have also been outlined, and they should be developed in the future.

Regarding future work, several lines of research appear intriguing. Regarding formal foundations, we surveyed the state of the art, yet several directions are open, such as considering further argumentation semantics and their effect on possibility, impossibility, or (computational) cost of (optimal) enforcement. Moreover, different types of modifications can be considered as well, reflecting different updates on the given argumentation.

Beyond Dung's classical argumentation framework, the notion of enforcement can be defined and applied to any enriched argumentation framework, such as value-based argumentation frameworks (see Chapter 5 [4]), or frameworks with higher-order bipolar interactions (see Chapter 1 [37]), or with quantitative additions like probabilistic argumentation (see Chapter 7 [61]). The notion can also be extended to semantics other than extension-based, for instance their labelling-based counterparts [34], or ranking-based semantics [32].

Chapter 4 [24] studies (among other notions) Incomplete Argumentation Frameworks (IAFs), that are strongly related to CAFs described in this Chapter. Moreover, the possibility of enforcing a set of arguments can be intuitively associated with the notion of possible acceptance in IAFs.

Enforcement is also related to the notion of dialogue (see Chapter 9 [29]), where it can be put in practice, and to that of strategic argumentation (see Chapter 10 [59]). To go further, an empirical cognitive study of enforcement might be conducted, as it has been done for other argumentation notions (see Chapter 14 [39]).

Acknowledgments

This work was supported by the Austrian Science Fund (FWF, P30168-N31). Furthermore, this work was supported by the German Federal Ministry of Education and Research (BMBF, 01/S18026A-F) by funding the competence center for Big Data and AI "ScaDS.AI Dresden/Leipzig".

References

[1] Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50(2):510–530, 1985.

[2] Leila Amgoud and Claudette Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34(1-3):197–215, 2002.

[3] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach.* Cambridge University Press, 2009.

[4] Katie Atkinson and Trevor Bench-Capon. Value-based argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 5. College Publications, 2021.

[5] Philippe Balbiani, Andreas Herzig, and Nicolas Troquard. Dynamic logic of propositional assignments: a well-behaved variant of pdl. In *2013 28th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 143–152. IEEE, 2013.

[6] P. Baroni, M. Giacomin, and B. Liao. Locality and modularity in abstract argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der

Torre, editors, *Handbook of Formal Argumentation*, chapter 19. College Publications, February 2018.

[7] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In *Handbook of Formal Argumentation*, chapter 4. College Publications, February 2018.

[8] Pietro Baroni, Federico Cerutti, Paul E. Dunne, and Massimiliano Giacomin. Automata for infinite argumentation structures. *Artificial Intelligence*, 203:104–150, 2013.

[9] Pietro Baroni and Massimiliano Giacomin. On principle-based evaluation of extension-based argumentation semantics. *Artificial Intelligence*, 171(10-15):675–700, 2007.

[10] R. Baumann. On the nature of argumentation semantics: Existence and uniqueness, expressibility, and replaceability. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 14. College Publications, February 2018. also appears in IfCoLog Journal of Logics and their Applications 4(8):2779-2886.

[11] Ringo Baumann. Splitting an argumentation framework. In James P. Delgrande and Wolfgang Faber, editors, *Proc. LPNMR*, volume 6645 of *Lecture Notes in Computer Science*, pages 40–53. Springer, 2011.

[12] Ringo Baumann. Normal and strong expansion equivalence for argumentation frameworks. *Artificial Intelligence*, 193:18–44, 2012.

[13] Ringo Baumann. What does it take to enforce an argument? minimal change in abstract argumentation. In Luc De Raedt, Christian Bessiere, Didier Dubois, Patrick Doherty, Paolo Frasconi, Fredrik Heintz, and Peter J. F. Lucas, editors, *Proc. ECAI*, volume 242, pages 127–132. IOS Press, 2012.

[14] Ringo Baumann. Context-free and context-sensitive kernels: Update and deletion equivalence in abstract argumentation. In Torsten Schaub, Gerhard Friedrich, and Barry O'Sullivan, editors, *Proc. ECAI*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 63–68. IOS Press, 2014.

[15] Ringo Baumann. *Metalogical Contributions to the Nonmonotonic Theory of Abstract Argumentation.* College Publications, 2014.

[16] Ringo Baumann. *On the Existence of Characterization Logics and Fundamental Properties of Argumentation Semantics.* habilitation treatise, Leipzig University, November 2019.

[17] Ringo Baumann and Gerhard Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Proc. COMMA*, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 75–86. IOS Press, 2010.

[18] Ringo Baumann and Gerhard Brewka. Extension removal in abstract argumentation - an axiomatic approach. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 2670–2677, 2019.

[19] Ringo Baumann, Gerhard Brewka, and Renata Wong. Splitting argumentation frameworks: An empirical evaluation. In Sanjay Modgil, Nir Oren, and Francesca Toni, editors, *Proc. TAFA, Revised Selected Papers*, volume 7132 of *Lecture Notes in Computer Science*, pages 17–31. Springer, 2011.

[20] Ringo Baumann, Wolfgang Dvorák, Thomas Linsbichler, and Stefan Woltran. A general notion of equivalence for abstract argumentation. *Artif. Intell.*, 275:379–410, 2019.

[21] Ringo Baumann and Christof Spanring. Infinite argumentation frameworks - On the existence and uniqueness of extensions. In *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation - Essays Dedicated to Gerhard Brewka on the Occasion of His 60th Birthday*, pages 281–295, 2015.

[22] Ringo Baumann and Christof Spanring. A study of unrestricted abstract argumentation frameworks. In Carles Sierra, editor, *Proc. IJCAI*, pages 807–813. ijcai.org, 2017.

[23] Dorothea Baumeister, Daniel Neugebauer, and Jörg Rothe. Credulous and skeptical acceptance in incomplete argumentation frameworks. In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *Proc. COMMA*, volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 181–192. IOS Press, 2018.

[24] Dorothea Baumeister, Daniel Neugebauer, and Jörg Rothe. Collective acceptability in abstract argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 4. College Publications, 2021.

[25] Dorothea Baumeister, Daniel Neugebauer, Jörg Rothe, and Hilmar Schadrack. Verification in incomplete argumentation frameworks. *Artificial Intelligence*, 264:1–26, 2018.

[26] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In *Proc. NMR*, pages 59–64, 2004.

[27] Philippe Besnard and Anthony Hunter. *Elements of Argumentation.* The MIT Press, 2008.

[28] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Change in argumentation systems: Exploring the interest of removing an argument. In Salem Benferhat and John Grant, editors, *Proc. SUM*, volume 6929 of *Lecture Notes in Computer Science*, pages 275–288. Springer, 2011.

[29] Elizabeth Black, Nicolas Maudet, and Simon Parsons. Argumentation-based dialogue. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 9. College Publications, 2021.

[30] Guido Boella, Souhila Kaci, and Leendert W. N. van der Torre. Dynamics in argumentation with single extensions: Abstraction principles and the grounded extension. In Claudio Sossai and Gaetano Chemello, editors, *Proc. ECSQARU*, volume 5590 of *Lecture Notes in Computer Science*, pages 107–118. Springer, 2009.

[31] Guido Boella, Souhila Kaci, and Leendert W. N. van der Torre. Dynamics in argumentation with single extensions: Attack refinement and the grounded extension (extended version). In Peter McBurney, Iyad Rahwan, Simon Parsons, and Nicolas Maudet, editors, *ArgMAS Revised Selected and Invited Papers*, volume 6057 of *Lecture Notes in*

Computer Science, pages 150–159. Springer, 2009.

[32] Elise Bonzon, Jerome Delobelle, Sebastien Konieczny, and Nicolas Maudet. A comparative study of ranking-based semantics for abstract argumentation. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI'16)*, 2016.

[33] R. Booth, S. Kaci, T. Rienstra, and L. W. N. van der Torre. A logical theory about dynamics in abstract argumentation. In Weiru Liu, V. S. Subrahmanian, and Jef Wijsen, editors, *Proc. SUM*, volume 8078 of *Lecture Notes in Computer Science*, pages 148–161. Springer, 2013.

[34] Martin Caminada. On the issue of reinstatement in argumentation. In Michael Fisher, Wiebe van der Hoek, Boris Konev, and Alexei Lisitsa, editors, *Logics in Artificial Intelligence, 10th European Conference, JELIA 2006, Liverpool, UK, September 13-15, 2006, Proceedings*, volume 4160 of *Lecture Notes in Computer Science*, pages 111–123. Springer, 2006.

[35] Martin Caminada and Yining Wu. On the limitations of abstract argumentation. In *Benelux Conference on Artificial Intelligence*, 2011.

[36] Thomas Caridroit, Sébastien Konieczny, and Pierre Marquis. Contraction in propositional logic. *International Journal of Approximate Reasoning*, 80:428–442, 2017.

[37] Claudette Cayrol, Andrea Cohen, and Marie-Christine Lagasquie-Schiex. Higher-order interactions (bipolar or not) in abstract argumentation: A state of the art. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 1. College Publications, 2021.

[38] Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Change in abstract argumentation frameworks: Adding an argument. *Journal of Artificial Intelligence Research*, 38:49–84, 2010.

[39] Federico Cerutti, Marcos Cramer, Mathieu Guillaume, Emmanuel Hadoux, Anthony Hunter, and Sylwia Polberg. Empirical cognitive studies about formal argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 14. College Publications, 2021.

[40] E. M. Clarke, A. Gupta, and O. Strichman. SAT-based counterexample-guided abstraction refinement. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 23(7):1113–1123, 2004.

[41] E.M. Clarke, O. Grumberg, S. Jha, Y. Lu, and H. Veith. Counterexample-guided abstraction refinement for symbolic model checking. *Journal of the ACM*, 50(5):752–794, 2003.

[42] Sylvie Coste-Marquis, Caroline Devred, Sébastien Konieczny, Marie-Christine Lagasquie-Schiex, and Pierre Marquis. On the merging of dung's argumentation systems. *Artificial Intelligence*, 171(10-15):730–753, 2007.

[43] Sylvie Coste-Marquis, Sébastien Konieczny, Jean-Guy Mailly, and Pierre Marquis. Extension enforcement in abstract argumentation as an optimization problem. In Qiang Yang and Michael J. Wooldridge, editors, *Proc. IJCAI*, pages 2876–2882. AAAI Press, 2015.

[44] Kristijonas Cyras, Xiuyi Fan, Claudia Schulz, and Francesca Toni. Assumption-based argumentation: Disputes, explanations, preferences. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 7, pages 365–408. College Publications, 2018.

[45] Florence Dupin de Saint-Cyr, Pierre Bisquert, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. Argumentation update in YALLA (yet another logic language for argumentation). *International Journal of Approximate Reasoning*, 75:57–92, 2016.

[46] Yannis Dimopoulos, Jean-Guy Mailly, and Pavlos Moraitis. Control argumentation frameworks. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proc. AAAI*, pages 4678–4685. AAAI Press, 2018.

[47] Yannis Dimopoulos, Jean-Guy Mailly, and Pavlos Moraitis. Argumentation-based negotiation with incomplete opponent profiles. In Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew E. Taylor, editors, *Proc. AAMAS*, pages 1252–1260. International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[48] S. Doutre, A. Herzig, and L. Perrussel. A dynamic logic framework for abstract argumentation. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proc. KR*, pages 62–71. AAAI Press, 2014.

[49] Sylvie Doutre, Andreas Herzig, and Laurent Perrussel. Abstract argumentation in dynamic logic: Representation, reasoning and change. In *Dynamics, Uncertainty and Reasoning*, pages 153–185. Springer, 2019.

[50] Sylvie Doutre, Faustine Maffre, and Peter McBurney. A Dynamic Logic Framework for Abstract Argumentation: Adding and Removing Arguments (regular paper). In Salem Benferhat, Karim Tabia, and Moonis Ali, editors, *Advances in Artificial Intelligence: From Theory to Practice - International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE 2017)*, volume 10351 of *Lecture Notes in Computer Science*, pages 295–305. Springer-Verlag, 2017.

[51] Sylvie Doutre and Jean-Guy Mailly. Comparison criteria for argumentation semantics. In Francesco Belardinelli and Estefania Argente, editors, *Proc. EUMAS/AT, Revised Selected Papers*, volume 10767 of *Lecture Notes in Computer Science*, pages 219–234. Springer, 2017.

[52] Sylvie Doutre and Jean-Guy Mailly. Semantic change and extension enforcement in abstract argumentation. In Serafín Moral, Olivier Pivert, Daniel Sánchez, and Nicolás Marín, editors, *Proc. SUM*, volume 10564 of *Lecture Notes in Computer Science*, pages 194–207. Springer, 2017.

[53] Sylvie Doutre and Jean-Guy Mailly. Constraints and changes: A survey of abstract argumentation dynamics. *Argument & Computation*, 9(3):223–248, 2018.

[54] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[55] Paul E. Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artif. Intell.*, 171(10-15):701–729, 2007.

[56] Paul E Dunne, Wolfgang Dvořák, Thomas Linsbichler, and Stefan Woltran. Char-

acteristics of multiple viewpoints in abstract argumentation. *Artificial Intelligence*, 228:153–178, 2015.

[57] Wolfgang Dvořák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 13, pages 631–688. College Publications, 2018.

[58] Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *Proc. ICLP/SLP*, pages 1070–1080. MIT Press, 1988.

[59] Guido Governatori, Michael J. Maher, and Francesco Olivieri. Strategic argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 10. College Publications, 2021.

[60] Adrian Haret, Johannes Peter Wallner, and Stefan Woltran. Two sides of the same coin: Belief revision and enforcing arguments. In Jérôme Lang, editor, *Proc. IJCAI*, pages 1854–1860. ijcai.org, 2018.

[61] Anthony Hunter, Sylwia Polberg, Nico Potyka, Tjitze Rienstra, and Matthias Thimm. Probabilistic argumentation: A survey. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 7. College Publications, 2021.

[62] Hirofumi Katsuno and Alberto O. Mendelzon. On the difference between updating a knowledge base and revising it. In James F. Allen, Richard Fikes, and Erik Sandewall, editors, *Proc. KR*, pages 387–394. Morgan Kaufmann, 1991.

[63] Hirofumi Katsuno and Alberto O. Mendelzon. Propositional knowledge base revision and minimal change. *Artificial Intelligence*, 52(3):263–294, 1992.

[64] Mark W. Krentel. The complexity of optimization problems. *Journal of Computer and System Sciences*, 36(3):490–509, 1988.

[65] Bei Shui Liao, Li Jin, and Robert C. Koons. Dynamics of argumentation systems: A division-based method. *Artificial Intelligence*, 175(11):1790–1814, 2011.

[66] Jean-Guy Mailly. Possible controllability of control argumentation frameworks. In *Computational Models of Argument - Proceedings of COMMA 2020, Perugia, Italy, September 4-11, 2020*, pages 283–294, 2020.

[67] Sanjay Modgil and Henry Prakken. Abstract rule-based argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 6, pages 287–364. College Publications, 2018.

[68] António Morgado, Federico Heras, Mark H. Liffiton, Jordi Planes, and João Marques-Silva. Iterative and core-guided MaxSAT solving: A survey and assessment. *Constraints*, 18(4):478–534, 2013.

[69] Ilkka Niemelä. Logic programs with stable model semantics as a constraint programming paradigm. *Annals of Mathematics and Artificial Intelligence*, 25(3-4):241–273, 1999.

[70] Andreas Niskanen and Matti Järvisalo. Strong refinements for hard problems in argumentation dynamics. In *Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020)*, pages 841–848, 2020.

[71] Andreas Niskanen, Daniel Neugebauer, and Matti Järvisalo. Controllability of control argumentation frameworks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 1855–1861, 2020.

[72] Andreas Niskanen, Johannes P. Wallner, and Matti Järvisalo. Optimal status enforcement in abstract argumentation. In Subbarao Kambhampati, editor, *Proc. IJCAI*, pages 1216–1222. IJCAI/AAAI Press, 2016.

[73] Andreas Niskanen, Johannes P. Wallner, and Matti Järvisalo. Pakota: A system for enforcement in abstract argumentation. In Loizos Michael and Antonis C. Kakas, editors, *Proc. JELIA*, volume 10021 of *Lecture Notes in Computer Science*, pages 385–400. Springer, 2016.

[74] Andreas Niskanen, Johannes Peter Wallner, and Matti Järvisalo. Extension enforcement under grounded semantics in abstract argumentation. In Michael Thielscher, Francesca Toni, and Frank Wolter, editors, *Proc. KR*, pages 178–183. AAAI Press, 2018.

[75] Andreas Niskanen, Johannes Peter Wallner, and Matti Järvisalo. Synthesizing argumentation frameworks from examples. *Journal of Artificial Intelligence Research*, 66:503–554, 2019.

[76] Emilia Oikarinen and Stefan Woltran. Characterizing Strong Equivalence for Argumentation Frameworks. *Artificial Intelligence*, 175(14-15):1985–2009, 2011.

[77] Christos H. Papadimitriou. *Computational complexity.* Academic Internet Publ., 2007.

[78] Régis Riveret. On learning abstract argumentation graphs from bivalent statement labellings. In *Proc. ICTAI*, pages 190–195. IEEE Computer Society, 2016.

[79] Régis Riveret and Guido Governatori. On learning attacks in probabilistic abstract argumentation. In Catholijn M. Jonker, Stacy Marsella, John Thangarajah, and Karl Tuyls, editors, *Proc. AAMAS*, pages 653–661. ACM, 2016.

[80] Gerard Sierksma and Yori Zwols. *Linear and integer optimization: theory and practice.* CRC Press, 2015.

[81] Christof Spanring. Hunt for the collapse of semantics in infinite abstract argumentation frameworks. In Claudia Schulz and Daniel Liew, editors, *Proceedings of the 2015 Imperial College Computing Student Workshop, ICCSW 2015*, volume 49 of *OASICS*, pages 70–77. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

[82] Markus Ulbricht and Ringo Baumann. If nothing is accepted - repairing argumentation frameworks. *J. Artif. Intell. Res.*, 66:1099–1145, 2019.

[83] Leon van der Torre and Srdjan Vesic. The principle-based approach to abstract argumentation semantics. *The IfCoLog Journal of Logics and their Applications*, 4(8):2727–2780, 2017.

[84] Johannes P. Wallner. Structural constraints for dynamic operators in abstract argumentation. *Argument & Computation*, 11(1–2):151–190, 2020.

[85] Johannes P. Wallner, Andreas Niskanen, and Matti Järvisalo. Complexity results and algorithms for extension enforcement in abstract argumentation. *Journal of Artificial Intelligence Research*, 60:1–40, 2017.

Received May 2021

Strategic Argumentation

Guido Governatori
Data61-CSIRO, Brisbane, Australia
guido.governatori@data61.csiro.au

Michael J. Maher
Reasoning Research Institute, Canberra, Australia
michael.maher@reasoning.org.au

Francesco Olivieri
Griffith University, Brisbane, Australia
francesco.olivieriphd@gmail.com

Abstract

Dialogue games are a dynamic form of argumentation, with multiple parties pooling their arguments with the intention of settling an issue. Such games can have a variety of structures, and may be collaborative or competitive, depending on the motivations of the parties. Strategic argumentation is a class of competitive dialogue games in which two players take turns in contributing their arguments, each attempting to have an issue settled in the way that they would prefer. Thus strategic argumentation games are less about discovering a joint truth than about a player imposing their view on an opponent. They are reflective of legal argumentation.

In the games we study, players have perfect information of the moves players make, but incomplete information on the possible moves (arguments) that other players have available to them. We look both at games using logically structured arguments and games using abstract arguments. We show that playing these games can be computationally hard. We also examine issues of corruption in such games, and discuss approaches to foiling it.

1 Introduction

When two or more parties are engaged in a debate, it is often the case that each party has some information they are not willing to disclose to the other parties. Also, in some cases, the disclosure of some piece of information by one party could prove detrimental for the party, in the sense that the information could be used to prevent the party to reach their aim in the debate, or some of the information disclosed can help the other party to achieve their goal. Accordingly we can provide the following (informal) definition of strategic argumentation.

Definition 1.1. Strategic argumentation *is the problem of determining what arguments (pieces of information) to disclose during a debate in order to achieve the aim a party has in the debate and to prevent the other party from gaining an undesired advantage.*

To illustrate the issue, consider the following argument exchange, first proposed in [124]:

Example 1.2. *Let* Pr *and* Op *be the players involved in the following argumentation dialogue (*Pr *and* Op *denote, respectively, the proponent and the opponent):*

Pr_0 : *"You killed the victim."*
Op_1 : *"I did not commit murder! There is no evidence!"*
Pr_1 : *"There is evidence. We found your ID card near the scene."*
Op_2 : *"It is not evidence! I had my ID card stolen!"*
Pr_2 : *"It is you who killed the victim. Only you were near the scene at the time of the murder."*
Op_3 : *"I did not go there. I was at facility A at that time."*
Pr_3 : *"At facility A? Then, it is impossible to have had your ID card stolen since facility A does not allow a person to enter without an ID card."*

We can easily represent arguments of this example with a rule-based formalism as follows. We have rules R:

$$\begin{array}{rrcl}
r_{\mathsf{Pr}_0}: & & \Rightarrow & murderer(X)\\
r'_{\mathsf{Op}_1}: & & \Rightarrow & \neg evidence\_Against(X)\\
r''_{\mathsf{Op}_1}: & \neg evidence\_Against(X) & \Rightarrow & \neg murderer(X)\\
r_{\mathsf{Pr}_1}: & ID(X)\_at\_crime\_scene & \Rightarrow & evidence\_Against(X)\\
r_{\mathsf{Op}_2}: & ID(X)\_stolen & \Rightarrow & \neg evidence\_Against(X)\\
r'_{\mathsf{Pr}_2}: & & \Rightarrow & only(X)\_at\_crime\_scene\\
r''_{\mathsf{Pr}_2}: & only(X)\_at\_crime\_scene & \Rightarrow & murderer(X)\\
r_{\mathsf{Op}_3}: & at\_facility\_A(X) & \Rightarrow & \neg only(X)\_at\_crime\_scene\\
r_{\mathsf{Pr}_3}: & at\_facility\_A(X) & \Rightarrow & \neg ID(X)\_stolen
\end{array}$$

and a priority relation $>= \{r_{\mathsf{Op}_2} > r_{\mathsf{Pr}_1}\}$, *where the notation* $r_i : A(r) \Rightarrow C(r)$ *identifies that* r_i *is the name of the rule,* $A(r)$ *is the set of antecedents (possibly empty)* *while $C(r)$ is the conclusion, symbol $\Rightarrow$ denotes that the conclusion may be defeated by contrary evidence, as for instance the conflict between r_{Op_2} and r_{Pr_1}, resolved by $>$ (the superiority relation) which allows us to conclude that $\neg evidence\_Against(X)$ is the case.*

A feature of this dialogue is that the exchange of arguments reflects an asymmetry of information between the two parties. Each player does not know the other player's knowledge, thus they cannot predict which arguments will be attacked, nor which counterarguments may be employed for attacking their own arguments. In addition, the private information disclosed by a party might eventually be used by the adversary to construct and play justified counterarguments. Thus, Pr_3 attacks Op_2, but only after Op_3 has been given. Thus, the attack Pr_3 of the proponent is possible *only when* the opponent discloses some private information through the move Op_3 (in this setting, after Op let Pr know that Op was at facility). If we assume that Pr wishes to expose Op's guilt, and Op wishes to hide it, then we can view this dialogue as a game, where a move consists of stating an argument.

This example illustrates a scenario where some of the information disclosed by a party could be detrimental to their aim. This is a common phenomenon in many applications that are suitable to be formally represented by argumentation such as negotiation [117], brokering [10], and in the legal domain [114; 63]. In a negotiation, the other party could use the information to gain some advantage either on the issue of the negotiation (e.g., price of an item) or on some side effects; in a legal proceeding the opposite party could use the information to win the case. Hence, players in such an argumentation game must be strategic in what arguments they expose, to put themselves in the best position. We refer to such games as *strategic argumentation* games.

Furthermore, in such applications the parties can be represented by agents acting and debating on behalf of their clients, but these agents might not have their client's best interests at heart. This can corrupt the dialogue. For example, suppose the agent for Pr was bribed by Op to omit the claim Pr_2. Then Op_3 would have remained private, and Op's lie would be undiscovered. Similar issues occur whenever we employ an agent, whether human or software.

Technically, games involving privacy are called games of *incomplete information.* As argued in [67], argument games with incomplete information can be modelled by stating that each player has a logical theory, constituting their private knowledge, and which is unknown by the opposite party, and there is an additional theory shared by all parties with the information that is public. A player may build an argument that supports their claim by using some of their private knowledge and the common information; in turn, the other party may construct new arguments by re-using the adversary's disclosed information (along with other pieces of their own private knowledge) in order to defeat the opponent's arguments. In a legal proceeding, we can distinguish between two types of information: the norms in force in the underlying jurisdiction, which are assumed to be known by both parties, and the information, private to each party, on the facts of the case. Accordingly, the legal proceeding can be modelled by three theories, a public one with the common knowledge, encoding the norms of the underlying jurisdiction, plus two private theories: one for each party.

When working with logically structured arguments, the different logical theories are represented by sets of rules (which may include unconditional facts). So, the set R of all rules used to build arguments is partitioned into three (distinct) subsets: a set R_{Com} known by both players, and two subsets R_{Pr} and R_{Op} corresponding, respectively, to Pr's and Op's private knowledge. While the game is evolving, at each turn, a party discloses some of their private arguments and, by doing so, the player reduces their private information ($R_{\mathsf{Pr}}/R_{\mathsf{Op}}$ decreases) with what now becomes part of the new common knowledge base (R_{Com} increases). Consider a setting where $F = \{a, d, f\}$ is the known set of facts (categorical statements), $R_{\mathsf{Com}} = F$ (the facts are common knowledge), and the players have the following sets of rules:

$$R_{\mathsf{Pr}} = \{r_0 : a \Rightarrow b;\ r_1 : d \Rightarrow c;\ r_2 : c \Rightarrow b\} \quad R_{\mathsf{Op}} = \{r_3 : c \Rightarrow e;\ r_4 : e, f \Rightarrow \neg b\}.$$

If Pr's intent is to prove b and plays $\{a \Rightarrow b\}$, then Pr wins the game. In fact, Op has no way to prove e and thus r_4 is not active. If, on the other hand, Pr plays $\{d \Rightarrow c,\ c \Rightarrow b\}$ (or even the whole R_{Pr}), this allows Op to succeed. Here, a minimal subset of R_{Pr} is successful. The situation can be reversed for Pr. Replace the sets of private rules with

$$R_{\mathsf{Pr}} = \{a \Rightarrow b;\ d \Rightarrow \neg c\} \qquad R_{\mathsf{Op}} = \{d, c \Rightarrow \neg b;\ f \Rightarrow c\}.$$

Playing $\{a \Rightarrow b\}$ is now not successful for Pr, while the whole R_{Pr} ensures victory.

Example 1.2 brings out the issues we will address in this chapter: formalizing such dialogues as strategic argumentation games, addressing the difficulty of making a move in a game, and examining the possibility of corruption in such games and means to foil it. We will look at both defeasible logics [6] and ASPIC-style structured argumentation [2; 111] as languages for expressing arguments. We will also show that the same issues arise if we formulate strategic argumentation in terms of abstract arguments [41]. In looking at corruption, we consider two forms: *espionage* and *collusion*. To counter these possibilities, we examine the use of standards and audit to limit the ability of players to behave corruptly, and the idea of computational resistance to corruption to discourage corruption.

The layout of this chapter is as follows. Section 2 describes the general setting of argumentation and dialogue games. Section 3 provides some technical background on computational complexity, elements of abstract argumentation [41], and a framework for argumentation with logically structured arguments. Section 4 outlines Defeasible Logic and its four main variants. Section 5 presents an instance of the strategic argumentation game with Defeasible Logic as the basis for argumentation, and proves the computational difficulty of playing the game. It extends this result to an instance of structured argumentation under the grounded semantics. Section 6 extends the idea of strategic argumentation further, to abstract argumentation over a variety of semantics. Section 7 investigates how corruption can affect argumentation games, and how it can be countered. Section 8 discusses related work and Section 9 considers possible future directions of this research. Section 10 ends the chapter.

2 Argumentation and Dialogue Games

In this section we briefly describe a general setting of argumentation and dialogue games. In doing so we will not bind concepts such as *argument*, *aim*, *acceptance* or *extension* to a specific meaning, nor specify all details of concepts like argumentation framework. They will be specified more precisely later.

Definition 2.1 (Argumentation framework)**.** *An* argumentation framework *AF is a tuple $(\mathcal{A}, \mathcal{R})$, where $\mathcal{A}$ is a set of arguments, and $\mathcal{R}$ is a collection of relations over $\mathcal{A}$.*

The literature in argumentation theory flourishes with different frameworks describing *what* arguments are, where the two main school of thoughts see them as either *monads* (with no internal structure), or *structured* (made of sub-parts). We will address both schools. For now, we are only interested in saying that there is a function mapping arguments to elements of the language, referred to as *conclusions* (or theses, claims).

Definition 2.2 (Conclusions)**.** *Given an argumentation framework AF and a language of expressions L, the function $conc : \mathcal{A} \mapsto 2^L$ maps each argument to a set of elements of L. If $c_A \in conc(A)$, then we call c_A a* conclusion *of argument A.*

In the monadic view, each argument might have a single, distinct conclusion. In that case, conclusions add nothing to the argumentation framework. In the structured view, an expression might be a conclusion of several arguments, and its negation might also be a conclusion of arguments. Any structured argumentation framework with conclusions can be abstracted to a monadic argumentation framework by simply ignoring its internal structure (and retaining the conclusion function).

For the purposes of this chapter, a semantics maps an argumentation framework to a set of extensions. Each semantics implicitly expresses a criterion for how arguments can coherently be adjudicated together, given an argumentation framework. Each extension in the semantics represents a "reasonable" adjudication, according to that criterion, of the arguments in the argumentation framework. We leave open the details of what an extension is and how it might be represented, but commonly it is a set of arguments or a labelling for arguments (see Section 3.2 for more details of these common representations).

Definition 2.3 (Semantics)**.** *A* semantics *is a function σ mapping argumentation frameworks to a set of extensions.*

There is a rich array of interactions that are considered dialogues in the argumentation literature [24] but, as can be seen from the introduction, we have a specific kind of dialogue in mind. We define a *dialogue* as the exchange of arguments between two (or more) parties. We talk of *dialogue games* when we want to analyse the formal properties of the dialogue, using criteria from game theory.

At the beginning of a dialogue game, each agent starts with a private set of arguments but they also share a (possibly empty) set of arguments that are common knowledge[1] to all players. This shared knowledge among the agents will be enriched throughout the game with the arguments played at each turn, as will be clear in the following.

Each player also has an *aim*, the details of which we leave open. Aims might be to have a particular argument accepted in at least one extension, under a particular semantics, or to have the cardinality of each extension, under a given semantics, be a prime number[2].

Our dialogue games consist of a state and possible changes of state.

Definition 2.4 (Dialogue Game State)**.** *Given a set of agents $Pl_1, \dots, Pl_n$ (referred to as* players*), a* dialogue game state *is an argumentation framework $(\mathcal{A}, \mathcal{R})$ where $\mathcal{R}$ contains unary relations $\xi_1, \dots \xi_n$ on $\mathcal{A}$, one for each player, as well as ξ_{Com} and, possibly, other relations.*

Each unary relation ξ_i defines a subset S_i of $\mathcal{A}$: $S_i = \{a \mid a \in \mathcal{A}, \xi_i(a)\}$. Similarly, $S_{\mathsf{Com}} = \{a \mid a \in \mathcal{A}, \xi_{\mathsf{Com}}(a)\}$. S_{Com} is the set of arguments that are common knowledge to all players, while S_i is the additional set of arguments that player Pl_i knows, but other players don't know she knows (they are *private*).

[1] By common knowledge we mean, not only that all players have knowledge of the arguments, but also each player knows that the others know, and each knows that the others know that she knows, and so on. [49]

[2] Admittedly, the latter example is not likely to arise in practice.

Thus, a dialogue game state can equally be viewed as a *split argumentation framework* $(\mathcal{A}, S_{\mathsf{Com}}, S_1, \ldots, S_n, \mathcal{R}')$, where $S_{\mathsf{Com}} \cap (\cup_{i=1}^{n} S_i) = \emptyset$ and $\mathcal{R}'$ is $\mathcal{R} \backslash \{\xi_{\mathsf{Com}}, \xi_1, \ldots, \xi_n\}$.

A dialogue game is a collection of players, each with their own aim, making moves, in turn, to achieve their aim[3].

Definition 2.5 (Dialogue Game)**.** *Given a set of players $Pl_1, \ldots, Pl_n$ and an aim for each player, a* dialogue game *consists of an initial dialogue game state in the form of a split argumentation framework $(\mathcal{A}, S_{\mathsf{Com}}, S_1, \ldots, S_n, \mathcal{R})$, and state transition rules (or* moves*) defined below.*

1. *Players take turns, meaning that only a single player can act at a given turn*[4]*.*
2. *At a given turn k, player Pl_i advances a subset T of its private arguments in order to achieve their aim. If S_{Com}^{k-1} and S_i^{k-1} denote, respectively, the common shared argumentation framework and Pl_i's private argumentation framework at turn $k-1$, then*
   - $S_{\mathsf{Com}}^{k} = S_{\mathsf{Com}}^{k-1} \cup T$
   - $S_i^{k} = S_i^{k-1} \setminus T$
   - $S_j^{k} = S^{k-1}$ *for* $j \neq i$
3. *The game ends at turn $k+1$, when either: (i) the aim of each player is satisfied, so no player has an incentive to change the state of the game, or (ii) no player with an unsatisfied aim is able to satisfy that aim by making a move.*

The state of the dialogue game after turn k is $(\mathcal{A}, S_{\mathsf{Com}}^{k}, S_1^{k}, \ldots, S_n^{k}, \mathcal{R})$. The common argumentation framework at that point is $CAF^k = (S_{\mathsf{Com}}^{k}, \mathcal{R})$.

According to the typology of argumentation games in [128], these dialogue games have a dialectical argumentation mechanism and players have no awareness of other players' arguments; agent type is not specified. The games we define below (Definitions 2.6 and 2.7) have an indicator agent type.

[3] Many different types of dialogue have been classified and many protocols have been provided for them; we refer to Chapter 9 of the present volume [24] for in depth analysis of the various alternatives. In this chapter we restrict ourselves to a minimal and limited view of dialogue games, suitable to define strategic argumentation.

[4] We shall not dwell on the details of how/which players are selected to act at a given turn, as it is outside the scope of this chapter. [128] discusses some other possibilities.

If we ignore turn-taking, our dialogue games are *memoryless*: the permitted moves are determined by the current dialogue state, independent of how that state was reached. Other forms of dialogue game may not have this property.

Note that, although the set of common arguments increases monotonically, this game is non-monotonic, meaning that, at any given turn, aims that were satisfied at the previous turn might now be unsatisfied.

Also note that we are considering the relations $\mathcal{R}$ to have a fixed meaning, independent of player's beliefs or perceptions. The *omniscient* argumentation framework corresponding to a dialogue game is $(\mathcal{A}, \mathcal{R})$.

We now formulate a specific type of dialogue games, namely *strategic argumentation dialogues*. In a strategic argumentation dialogue game, we have only *two* players, who take turns in exchanging arguments to accept/reject a topic φ, where $\varphi \in L$. We name one player *Proponent* (Pr), and the other *Opponent* (Op). We shall consider two variants of the strategic argumentation dialogue game: the *symmetric*, and the *asymmetric* strategic argumentation dialogue game. In the symmetric variant, both parties have the burden of proof, that is, the proponent has to establish φ, where the opponent has to establish $\neg\varphi$. (With $\neg\varphi$, we denote the contrary of φ.) In the asymmetric variant, the proponent still has to establish φ, whereas the opponent aims to prevent this.

In the symmetric variant, at one turn, either φ, or $\neg\varphi$, is accepted. If φ is accepted, then it is the opponent's turn; if $\neg\varphi$ is accepted, then is the proponent's turn. At a given turn, the player has two possible courses of action. First, they play a subset of their private argumentation framework (i.e., a non-empty set of arguments). By doing so, they increment the shared argumentation framework with the arguments just played. Second, they pass and admit defeat. This happens when they are not able to change the status of the conclusion. The game ends when a player passes.

Definition 2.6 (Symmetric Strategic Argumentation Dialogue Game)**.** *Consider two players, a proponent* Pr *and an opponent* Op*, an expression* $\varphi \in L$*, a dialogue game state in the form of a split argumentation framework* $(\mathcal{A}, S_{\mathsf{Com}}, S_{\mathsf{Pr}}, S_{\mathsf{Op}}, \mathcal{R})$*, and a conclusion function conc. Suppose that there is an argument* $a \in S_{\mathsf{Pr}}$ *such that* $\varphi \in conc(a)$*.*

Let S^k_{Com}*,* S^k_{Pr}*, and* S^k_{Op} *denote, respectively, the common knowledge arguments,* Pr*'s private arguments and* Op*'s private arguments after turn* k*. (In particular,* $S^0_{\mathsf{Com}} = S_{\mathsf{Com}}$*,* $S^0_{\mathsf{Pr}} = S_{\mathsf{Pr}}$*, and* $S^0_{\mathsf{Op}} = S_{\mathsf{Op}}$*.)*

We define a symmetric strategic argumentation dialogue game *as a dialogue game where:*

1. *The players take turns; if* φ *is accepted by* CAF^0 *under semantics* σ*, then* Op

begins; otherwise Pr *does so.*

2. *At turn k, if* $\neg\varphi$ *is accepted in* CAF^{k-1}*, then it is* Pr*'s turn to play, as follows*
   - Pr *advances a subset of its private arguments* $T \subseteq S^{k-1}_{\mathsf{Pr}}$ *so that* φ *is accepted in* CAF^k*. As a result*
     - $S^k_{\mathsf{Com}} = S^{k-1}_{\mathsf{Com}} \cup T$;
     - $S^k_{\mathsf{Pr}} = S^{k-1}_{\mathsf{Pr}} \setminus T$.
     - $S^k_{\mathsf{Op}} = S^{k-1}_{\mathsf{Op}}$
3. *At turn k, if* φ *is accepted in* CAF^{k-1}*, then it is* Op*'s turn to play, as follows*
   - Op *advances a subset of its private arguments* $T \subseteq S^{k-1}_{\mathsf{Op}}$ *so that* $\neg\varphi$ *is accepted in* CAF^k*. As a result*
     - $S^k_{\mathsf{Com}} = S^{k-1}_{\mathsf{Com}} \cup T$;
     - $S^k_{\mathsf{Pr}} = S^{k-1}_{\mathsf{Pr}}$
     - $S^k_{\mathsf{Op}} = S^{k-1}_{\mathsf{Op}} \setminus T$.
4. *The game ends at turn* $k+1$*, when either (i) it is* Pr*'s turn and there is no move for* Pr *such that* CAF^{k+1} *accepts* φ*, in which case* Op *wins, or (ii) it is* Op*'s turn and there is no move for* Op *such that* CAF^{k+1} *accepts* $\neg\varphi$*, in which case* Pr *wins.*

The only difference in the asymmetric variant with respect to the symmetric one is that, the opponent no longer has the burden of proof: during her turn, Op proposes arguments in order to prevent acceptance of φ, rather than to accept $\neg\varphi$ (see point 3).

Definition 2.7 (Asymmetric Strategic Argumentation Dialogue Game)**.** *Consider two players, a proponent* Pr *and an opponent* Op*, an expression* $\varphi \in L$*, a dialogue game state in the form of a split argumentation framework* $(\mathcal{A}, S_{\mathsf{Com}}, S_{\mathsf{Pr}}, S_{\mathsf{Op}}, \mathcal{R})$*, and a conclusion function conc.*

Let S^k_{Com}*,* S^k_{Pr}*, and* S^k_{Op} *denote, respectively, the common knowledge arguments,* Pr*'s private arguments and* Op*'s private arguments after turn k. (In particular,* $S^0_{\mathsf{Com}} = S_{\mathsf{Com}}$*,* $S^0_{\mathsf{Pr}} = S_{\mathsf{Pr}}$*, and* $S^0_{\mathsf{Op}} = S_{\mathsf{Op}}$*.)*

We define an asymmetric strategic argumentation dialogue game *as a dialogue game where:*

1. *The players take turns; if* φ *is accepted by* CAF^0 *under semantics* σ*, then* Op *begins; otherwise* Pr *does so.*

2. *At turn k, if φ is not accepted in CAF^{k-1}, then it is* Pr*'s turn to play, as follows*

   - Pr *advances a subset of its private arguments $T \subseteq S^{k-1}_{\mathsf{Pr}}$ so that φ is accepted in CAF^k. As a result*
     - $S^k_{\mathsf{Com}} = S^{k-1}_{\mathsf{Com}} \cup T$;
     - $S^k_{\mathsf{Pr}} = S^{k-1}_{\mathsf{Pr}} \setminus T$.
     - $S^k_{\mathsf{Op}} = S^{k-1}_{\mathsf{Op}}$

3. *At turn k, if φ is accepted in CAF^{k-1}, then it is* Op*'s turn to play, as follows*

   - Op *advances a subset of its private arguments $T \subseteq S^{k-1}_{\mathsf{Op}}$ so that φ is not accepted in CAF^k. As a result*
     - $S^k_{\mathsf{Com}} = S^{k-1}_{\mathsf{Com}} \cup T$;
     - $S^k_{\mathsf{Pr}} = S^{k-1}_{\mathsf{Pr}}$
     - $S^k_{\mathsf{Op}} = S^{k-1}_{\mathsf{Op}} \setminus T$.

4. *The game ends at turn $k+1$, when either (i) it is* Pr*'s turn and there is no move for* Pr *such that CAF^{k+1} accepts φ, in which case* Op *wins, or (ii) it is* Op*'s turn and there is no move for* Op *such that CAF^{k+1} does not accept φ, in which case* Pr *wins.*

Thus both variants are dialogue games between two players arguing about a conclusion φ on the basis of their common argumentation framework. They leave open the notion of acceptance and the details of the set of relations $\mathcal{R}$, but specify more precisely the aims of the players. From now on, we will use the abbreviations SSA for Symmetric Strategic Argumentation, and AsSA for Asymmetric Strategic Argumentation.

The asymmetric game can be seen in situations where the parties have different proof standards. For example, in a criminal proceeding the prosecution must prove its case “beyond a reasonable doubt”, while the defence has only to prevent this. An asymmetric dialogue game was presented in [48].

The problems that the players must solve in order to move vary slightly according to the kind of game played (SSA vs. AsSA) and the players (Pr and Op). We formulate them as decision problems as follows:

SSA Problem under Semantics σ

Let $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ be the split argumentation framework as in Definition 2.6 after turn k, and $\varphi \in L$ be the content of the dispute.

> Pr's instance for turn $k+1$: A split argumentation framework $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ and an expression $\varphi \in L$.
>
> Question: Does there exist a subset T of S^k_{Pr} such that φ is accepted by CAF^{k+1} under semantics σ?
>
> Op's instance for turn $k+1$: A split argumentation framework $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ and an expression $\varphi \in L$.
>
> Question: Does there exist a subset T of S^k_{Op} such that $\neg\varphi$ is accepted by CAF^{k+1} under semantics σ?

Analogously, we can formalise the AsSA Problem.

AsSA Problem under Semantics σ

Let $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ be the split argumentation framework as in Definition 2.7 after turn k, and $\varphi \in L$ be the content of the dispute.

> Pr's instance for turn $k+1$: A split argumentation framework $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ and an expression $\varphi \in L$.
>
> Question: Does there exist a subset T of S^k_{Pr} such that φ is accepted by CAF^{k+1} under semantics σ?
>
> Op's instance for turn $k+1$: A split argumentation framework $(\mathcal{A}, S^k_{\mathsf{Com}}, S^k_{\mathsf{Pr}}, S^k_{\mathsf{Op}}, \mathcal{R})$ and an expression $\varphi \in L$.
>
> Question: Does there exist a subset T of S^k_{Op} such that φ is not accepted by CAF^{k+1} under semantics σ?

In Section 5, we will give an implementation of the strategic argumentation game with *Defeasible Logic* (DL) [104] as the underlying logical framework, and assess the complexity of these problems.

3 Background

In this section we outline the concepts we use involving computational complexity, abstract and structured argumentation. This is not intended to be an introduction to these topics, it is simply a sketch of the concepts, assuming a familiarity with the more common elements. Those with less familiarity with these topics might want to read an introduction first, such as [75; 45] for computational complexity, [13; 12] for abstract argumentation, and [112] for structured argumentation.

$$\begin{tikzcd}
 & \mathrm{NP} \arrow[dr, phantom, sloped, "\supseteq"] & & & & \Sigma_2^p \arrow[dr, phantom, sloped, "\supseteq"] & & & \\
\mathrm{P} \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \mathrm{D}^p \arrow[r, phantom, "\subseteq"] & \Theta_2^p \arrow[r, phantom, "\subseteq"] & \Delta_2^p \arrow[ur, phantom, sloped, "\subseteq"] \arrow[dr, phantom, sloped, "\subseteq"] & & \mathrm{D}_2^p \arrow[r, phantom, "\subseteq"] & \Delta_3^p \arrow[r, phantom, "\subseteq"] & \cdots \\
 & \mathrm{coNP} \arrow[ur, phantom, sloped, "\subseteq"] & & & & \Pi_2^p \arrow[ur, phantom, sloped, "\subseteq"] & & &
\end{tikzcd}$$

$$\cdots \subseteq PH \subseteq \mathrm{P}^{\mathrm{PP}} \subseteq \mathrm{NP}^{\mathrm{PP}} \subseteq \mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}} \subseteq \mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}} \cdots \subseteq \mathrm{PSPACE}$$

Figure 1: Some complexity classes in the polynomial counting hierarchy, ordered by containment.

3.1 Complexity Classes

When addressing computational complexity we will focus on decision problems, because of their more familiar complexity classes, rather than their functional counterparts, which are more appropriate for many of the computational tasks we will address. We assume familiarity with the polynomial time complexity hierarchy but we will introduce some other complexity classes that we will need, and computational problems that are complete for each class. As is usual, $\mathcal{D}^{\mathcal{C}}$ denotes the class of problems that can be solved with complexity $\mathcal{D}$ if given an oracle for a problem in $\mathcal{C}$.

Within the polynomial hierarchy, a complete problem for Σ_n^p (Π_n^p) is the satisfiability of quantified Boolean formulas (QBF) with quantifiers in the form $\exists\forall\exists\cdots\exists$ (respectively, $\forall\exists\forall\cdots\exists$) with n alternations of quantifiers. PSPACE is the class of decision problems solvable in polynomial space. It contains the entire polynomial hierarchy PH. A complete problem for PSPACE is satisfiability of all quantified Boolean formulas.

D^p is the complexity class of problems that can be expressed as the conjunction of a problem in NP and a problem in coNP. A complete problem for D^p asks, given Boolean formulas ϕ and ψ, is ϕ unsatisfiable and ψ satisfiable? $\mathrm{NP}^{D^p} = \Sigma_2^p$. Similarly D_2^p is the conjunction of problems in Σ_2^p and Π_2^p.

Θ_2^p is the class of decision problems solvable by a deterministic polynomial algorithm with $\mathrm{O}(log\, n)$ calls to an NP oracle. It is equal to $\mathrm{P}_{||}^{\mathrm{NP}}$, the class of problems solvable by a deterministic polynomial algorithm with non-adaptive calls to an NP oracle. Non-adaptive refers to the restriction that oracle calls cannot depend on the outcome of previous calls. $\mathrm{NP}^{\Theta_2^p} = \Sigma_2^p$.

Δ_2^p is equal to P^{NP}. A complete problem for Δ_2^p is, given a Boolean formula ψ, does the lexicographically last satisfying assignment for ψ end with a 1?

PP is, roughly, the class of decision problems that have more accepting paths than rejecting paths. It can be thought of as a decision problem version of the more familiar counting complexity class #P, which addresses absolute counting, while PP addresses relative size of counts. We have $\mathrm{P}^{\#\mathrm{P}} = \mathrm{P}^{\mathrm{PP}}$ and $\mathrm{NP}^{\#\mathrm{P}} = \mathrm{NP}^{\mathrm{PP}}$. The entire polynomial hierarchy is contained within $\mathrm{NP}^{\mathrm{PP}}$. A complete problem for PP, called MAJSAT, is to decide whether a given Boolean formula is satisfied by more than half of the assignments to its variables. This can be expressed via a "counting" quantifier C as satisfying $\mathsf{C}X\ \psi$. Similarly, a complete problem for $\mathrm{NP}^{\mathrm{PP}}$, called E-MAJSAT is satisfying formulas $\exists X \mathsf{C} Y\ \psi$. And so on.

The *counting polynomial hierarchy* [137] extends the polynomial hierarchy by incorporating PP, P^{PP}, $\mathrm{NP}^{\mathrm{PP}}$, $\mathrm{coNP}^{\mathrm{PP}}$, etc. Figure 1 displays containment relations among relevant complexity classes. In addition to the containments displayed, $\Theta_2^p \subseteq \mathrm{PP} \subseteq \mathrm{P}^{\mathrm{PP}}$.

3.2 Abstract Argumentation

Definition 3.1 (Abstract Argumentation Framework)**.** *An abstract argumentation framework is a pair $(\mathcal{A}, \gg)$ where $\mathcal{A}$ is a set of arguments and $\gg$ is a subset of $\mathcal{A} \times \mathcal{A}$, where $(a, b) \in \gg$ denotes that a attacks b.*

An abstract argumentation framework can be represented as a directed graph, where each vertex is an argument, and a directed edge from a to b if a attacks b. An argumentation framework is acyclic if the corresponding directed graph is acyclic.

For the purposes of this chapter, a semantics maps an argumentation framework to a set of extensions, each extension being a set of arguments (the set of arguments accepted in that extension)[5]. When representing the state of an argument in an extension, we will use the labelling approach (see, for example, [13; 12]) in which the argument is labelled either `in`, `out`, or `undec`. That is, an extension E is defined as a function $\mathcal{L}ab_E : \mathcal{A} \to \{\texttt{in}, \texttt{out}, \texttt{undec}\}$. Then we can define an extension E as $\{a \in \mathcal{A} \mid \mathcal{L}ab_E(a) = \texttt{in}\}$.

Given an argumentation framework $\mathcal{AF} = (\mathcal{A}, \gg)$, an argument a is said to be *accepted* in an extension E if $\mathcal{L}ab_E(a) = \texttt{in}$, *rejected* in E if $\mathcal{L}ab_E(a) = \texttt{out}$, and *undecided* in E if $\mathcal{L}ab_E(a) = \texttt{undec}$. An extension E is *conflict-free* if no accepted argument is attacked by an accepted argument. An argument a is *defended* by E if every argument that attacks a is attacked by some argument accepted in E. An extension E of $\mathcal{AF}$ is *stable* if it is conflict-free and for every argument $a \in \mathcal{A} \backslash E$

[5] Thus we will not address the gradual and ranking semantics discussed in [15; 1].

there is an argument in E that attacks a. An extension E of $\mathcal{AF}$ is *complete* if it is conflict-free and, $a \in E$ iff a is defended by E.

The set of complete extensions forms a lower semi-lattice under the containment ordering, and many semantics can be defined directly in terms of this semi-lattice. The least complete extension under the containment ordering exists and is called the *grounded* extension. The *preferred* extensions are the maximal complete extensions under the containment ordering. The *semi-stable* extensions are the complete extensions where the set of arguments labelled with `in` or `out` is maximal under the containment ordering. The *ideal* extension is the maximal complete extension contained in all preferred extensions. Similarly, the *eager* extension is the maximal complete extension contained in all semi-stable extensions. These are not necessarily the original definitions of these extensions, but they are equivalent definitions.

We will use $\mathcal{GR}$ to denote the grounded semantics, $\mathcal{ST}$ for the stable semantics, $\mathcal{CO}$ for the complete semantics, $\mathcal{PR}$ for the preferred semantics, $\mathcal{ST}$ for the stable semantics, $\mathcal{SST}$ for the semi-stable semantics, $\mathcal{EA}$ for the eager semantics, and $\mathcal{ID}$ for the ideal semantics.

We say a semantics is *completist* if every argumentation framework is mapped to a set of complete extensions. These semantics will be our main focus. A semantics is *strongly completist* if it is completist and the set of extensions is determined by the semi-lattice structure of the complete extensions. Among the completist semantics are the grounded, preferred, stable, semi-stable, ideal, eager, and complete semantics. All except the stable semantics are strongly completist. Stable extensions are defined by a property of the individual extension, rather than by a structural property within the semi-lattice of complete extensions, and it turns out there is no equivalent structural definition [90]. Stable semantics is also exceptional in that some argumentation frameworks have no stable extensions.

Each semantics implicitly expresses a criterion for what arguments can coherently be accepted together, given an argumentation framework. Each extension in the semantics represents a “reasonable” adjudication, according to that criterion, of the arguments in the argumentation framework.

Our restriction to completist semantics is, then, an implicit requirement that reasonable adjudications are conflict-free, defend all the accepted arguments, and accept all the defended arguments.[6] Each of the semantics, except (obviously) the complete semantics, imposes extra requirements, reflecting different emphases: the grounded semantics is highly sceptical, requiring a minimal set of accepted arguments[7]; the preferred semantics requires maximal sets of accepted arguments;

[6] However, we make this restriction in this chapter only for simplicity, and not on the basis that this implicit requirement is justified.

[7] Or, equivalently, accepting only arguments that are accepted in all complete extensions.

the stable semantics requires that no argument is left undecided; the semi-stable semantics requires minimal sets of undecided arguments; the ideal semantics requires accepting only arguments that are accepted in all preferred extensions, and accepting as many of these as possible; the eager semantics requires accepting only arguments that are accepted in all semi-stable extensions, and accepting as many of these as possible.

The grounded, ideal and eager semantics are *unitary*: they contain exactly one extension. Such semantics limit, somewhat, the range of possible strategic aims of players in strategic argumentation, as we will see later.

Structural properties of an argumentation framework can influence the relationship between various semantics, which can make proving the computational complexity of some problems easier. An argumentation framework is *well-founded* if there is no infinite sequence of arguments $a_1, a_2, \ldots, a_i, a_{i+1}, \ldots$ such that, for each i, a_{i+1} attacks a_i. Such argumentation frameworks have a single complete extension which must be the grounded extension [41], in which every argument is either accepted or rejected. Every completist semantics for such argumentation frameworks consists of this single extension.

An argument framework is *coherent* if every preferred extension is stable. An argument b *indirectly attacks* an argument a if there is a path of odd length from b to a, and *indirectly defends* a if there is a path of even length from b to a. An argument b is *controversial* wrt a if b indirectly attacks a and indirectly defends a. An argument is *controversial* if it is controversial wrt some argument. An argument framework is *uncontroversial* if there is no controversial argument. An argument framework is *limited controversial* if there is no infinite sequence of arguments $a_1, a_2, \ldots, a_i, \ldots$ such that a_{i-1} is controversial wrt a_i. Dung shows that (Theorem 33 of [41]) every limited controversial argument framework is coherent, and every uncontroversial argument framework is also relatively grounded. An argument framework is *relatively grounded* if intersection of all preferred extensions coincides with the grounded extension.

3.3 Structured Argumentation

Argumentation takes place over a language of expressions, most commonly a language of literals. For definiteness, in this chapter we consider propositional literals.

Definition 3.2 (Language)**.** *The* language L *of expressions consists of a set of literals. Given a set* PROP *of propositional atoms, the set of* literals *is* Lit $=$ PROP $\cup \{\neg p \mid p \in \text{PROP}\}$. *We denote with* $\sim p$ *the* complementary *of literal* p*; if* p *is a positive literal* q*, then* $\sim p$ *is* $\neg q$*, and if* p *is a negative literal* $\neg q$*, then* $\sim p$ *is* q*.*

Rules are built out of these expressions. Rules have labels to name them, but these are completely separate from labels used in abstract argumentation.

Definition 3.3 (Rules)**.** *Let* Lab *be a set of rule labels. A* rule *r with $r \in$* Lab *describes the relation between a set of expressions, called the* antecedent *(or* body *or the* premise*) of r and denoted by $A(r)$ (which may be empty) and an expression, called the* consequent*, or* head*, of r and denoted by $C(r)$. Three kind of rules are allowed:* strict rules *of the form $r : A(r) \rightarrow C(r)$,* defeasible rules *of the form $r : A(r) \Rightarrow C(r)$, and* defeaters *of the form $r : A(r) \leadsto C(r)$.*

A strict rule is a rule in the classical sense: whenever the antecedent holds, so does the conclusion. We call a strict rule without antecedent a *fact*, but we often distinguish facts from "true" strict rules that have an antecedent. A defeasible rule is allowed to assert its conclusion unless there is contrary evidence to it. A defeater is a rule that cannot be used to draw any conclusion, but can provide contrary evidence to complementary conclusions. A defeater in this sense [102] can be considered an instance of the general notion of defeater in epistemology: evidence that counts against a belief.

Definition 3.4 (Argumentation Theory)**.** *An* argumentation theory *D is a structure $(R, >)$, where R is a (finite) set of rules and $> \subseteq R \times R$ is a binary relation on R called the* superiority relation*.*

The relation $>$ describes the relative strength of rules, that is to say, when a single rule may override the conclusion of another rule, and is required to be irreflexive, asymmetric, and acyclic (i.e., its transitive closure is irreflexive). To simplify discussion, we assume no strict rule is inferior to another rule. We use the following abbreviations on R: the set of strict rules in R is denoted by R_s, the set of strict and defeasible rules in R by R_{sd}, the set of defeasible rules by R_d, the set of defeaters by R_{dft}, and $R[q]$ is the set of rules in R whose head is q.

To demonstrate these definitions, we look at a time-honoured example of defeasible reasoning.

Example 3.5. *Consider an argumentation theory consisting of the following rules*

$$\begin{array}{rrcl}
r_1 : & bird(X) & \Rightarrow & fly(X) \\
r_2 : & penguin(X) & \Rightarrow & \neg fly(X) \\
r_3 : & penguin(X) & \rightarrow & bird(X) \\
r_4 : & injured(X) & \leadsto & \neg fly(X) \\
f : & penguin(tweety) & & \\
g : & bird(freddie) & & \\
h : & injured(freddie) & &
\end{array}$$

and a priority relation $r_2 > r_1$.

Here r_1, r_2, r_3, r_4, f are labels and r_3 is (a reference to) a strict rule, while r_1 and r_2 are defeasible rules, r_4 is a defeater, and f, g, h are facts. Thus $R_s = \{r_3, f, g, h\}$ and $R_{sd} = R = \{r_1, r_2, r_3\}$ and $>$ consists of the single tuple (r_2, r_1). The rules express that birds usually fly (r_1), penguins usually don't fly (r_2), that all penguins are birds (r_3), and that an injured animal may not be able to fly (r_4). In addition, we are given the facts that tweety is a penguin, and $freddie$ is an injured bird. Finally, the priority of r_2 over r_1 expresses that when something is both a bird and a penguin (that is, when both rules can fire) it usually cannot fly (that is, only r_2 may fire, it overrules r_1).

By combining the rules in a theory, we can build arguments (we adjust the definition in [112] to meet Definition 3.4). In what follows, for a given argument A, Conc returns its conclusion, Sub returns all its sub-arguments, Rules returns all the rules in the argument and, finally, TopRule returns the last inference rule in the argument.

Definition 3.6 (Argument)**.** *Let $D = (R, >)$ be an argumentation theory and $\Rrightarrow \in \{\rightarrow, \Rightarrow, \leadsto\}$. An argument A constructed from D has the form $A_1, \ldots, A_n \Rrightarrow_r \psi$, where*

- *A_k is an argument constructed from D, for $1 \leq k \leq n$, and*
- *$r : \mathsf{Conc}(A_1), \ldots, \mathsf{Conc}(A_n) \Rrightarrow \psi$ is a rule in R.*

The set of arguments constructed from D is the smallest set of arguments satisfying this condition.

With regard to argument A, the following holds:

$$\begin{array}{l}
\mathsf{Conc}(A) = \psi \\
\mathsf{Sub}(A) = \mathsf{Sub}(A_1) \cup \cdots \cup \mathsf{Sub}(A_n) \cup \{A\} \\
\mathsf{TopRule}(A) = r : \mathsf{Conc}(A_1), \ldots, \mathsf{Conc}(A_n) \Rrightarrow \psi \\
\mathsf{Rules}(A) = \mathsf{Rules}(A_1) \cup \cdots \cup \mathsf{Rules}(A_n) \cup \{\mathsf{TopRule}(A)\} \\
(\mathsf{Rules}(A_1) \cup \cdots \cup \mathsf{Rules}(A_n)) \cap R_{dft} = \emptyset
\end{array}$$

If $\mathsf{Rules}(A) \subseteq R_s$ *then argument A is* strict*, otherwise A is* defeasible*. If* $\mathsf{Rules}(A) \cap R_{dft} \neq \emptyset$ *then argument A is* non-supportive*, otherwise it is* supportive*.*

Conflicts between contradictory argument conclusions are resolved on the basis of preferences over arguments using a simple last-link ordering. An argument A is stronger than another argument B (written $A > B$) iff B is defeasible, and either A is strict or $\mathsf{TopRule}(A)$ is stronger than $\mathsf{TopRule}(B)$ ($\mathsf{TopRule}(A) > \mathsf{TopRule}(B)$).

Definition 3.7 (Attacks)**.** *An argument B attacks an argument A iff $\exists A' \in \mathsf{Sub}(A)$ such that $\mathsf{Conc}(B) = {\sim}\mathsf{Conc}(A')$, and $A' \not> B$.*

We can now define the argumentation framework that is determined by an argumentation theory.

Definition 3.8 (AF determined by an argumentation theory)**.** *Let $D = (R, >)$ be an argumentation theory. The* argumentation framework determined by D *is $(\mathcal{A}, \gg)$, where $\mathcal{A}$ is the set of all arguments constructed from D, and $\gg$ is the attack relation defined above.*

Given this definition of argumentation framework, if D is an argumentation theory, we can abuse notation somewhat and write $\mathcal{GR}(D)$ to denote the grounded extension of the argumentation framework determined by D.

Definition 3.9 (Justified Conclusion)**.** *Given an argumentation theory D, we say a conclusion ψ is* justified *by D under the grounded semantics iff there exists a supportive argument a in $\mathcal{GR}(D)$ such that $\mathsf{Conc}(a) = \psi$.*

The following example illustrates the notions just introduced.

Example 3.10. *Using the rules from Example 3.5, we have arguments:*

$$\begin{array}{lll} A_1 : & \rightarrow_f penguin(tweety) & (strict\ argument) \\ A_2 : & A_1 \rightarrow_{r_3} bird(tweety) & (strict\ argument) \\ A_3 : & A_2 \Rightarrow_{r_1} fly(tweety) & (defeasible\ argument) \\ A_4 : & A_1 \Rightarrow_{r_2} \neg fly(tweety) & (defeasible\ argument) \end{array}$$

among others.

If we consider the argument A_3, we have

$$\begin{array}{ll} \mathsf{Conc}(A_3) & = fly(tweety) \\ \mathsf{Sub}(A_3) & = \{A_1, A_2, A_3\} \\ \mathsf{TopRule}(A_3) & = r_1 \\ \mathsf{Rules}(A_3) & = \{f, r_1, r_3\} \end{array}$$

A_4 attacks A_3 because the two arguments have contradictory conclusions and $r_1 \not> r_2$. On the other hand, A_3 does not attack A_4 because $r_2 > r_1$.

In the argumentation framework determined by this theory there is no argument attacking A_4. Hence A_4 appears in the grounded extension. Since A_4 is a supportive argument, its conclusion $\neg fly(tweety)$ is justified under the grounded semantics.

4 Defeasible Logic

Defeasible Logic (DL) [103] is a rule-based sceptical approach to non-monotonic reasoning. It is based on a logic programming-like language and is a simple, efficient but flexible formalism capable of dealing with many intuitions of non-monotonic reasoning in a natural and meaningful way [4].

Defeasible rule languages like defeasible logic have been shown to be useful in representing legal documents and reasoning [113; 9; 118; 68; 66; 74; 72]. There are a variety of defeasible logics, which have been argued to represent the different proof standards that apply in legal systems [62; 64].

Defeasible logics have much in common with argumentation, but there is only little work substantiating the relationship. [65] characterizes inference in two defeasible logics in terms of argumentation. [62] maps proof in Carneades [59] at a given proof standard into proof in a defeasible logic. [79] showed how to map one instance of $ASPIC^{+}$ into a defeasible logic. [93] gave two embeddings of abstract argumentation frameworks $\mathcal{AF}$ into a small subset of defeasible rule languages, implying, in particular, that acceptance in the grounded extension of $\mathcal{AF}$ can be implemented in a wide variety of defeasible logics and other concrete defeasible reasoning formalisms.

In this section we define two defeasible logics, but first we introduce defeasible logic in general.

4.1 Defeasible logic

The language of DL consists of literals and rules. To avoid notational redundancies, we use the same definitions of PROP, Lit, complementary literal, and the same rule types, structure and notation as already introduced in Definition 3.2.

A defeasible theory D is a triple $(F, R, >)$, where $F \subseteq \text{Lit}$ is a set of indisputable statements called *facts*, R is a (finite) set of rules, and $> \subseteq R \times R$ is a superiority relation on R as introduced in Definition 3.4.

A *derivation* (or *proof*) is a finite sequence $P = P(1), \ldots, P(n)$ of *tagged literals* of the type $+\Delta q$ (q is definitely provable), $-\Delta q$ (q is definitely refuted), $+d\,q$ (q is defeasibly provable) and $-d\,q$ (q is defeasibly refuted). The proof conditions below define the logical meaning of such tagged literals. Given a proof P, $P(n)$ denotes the n-th element of the sequence, and $P(1..n)$ denotes the first n elements of P. $\pm\Delta$ and $\pm df$ are called *proof tags*. Given # a proof tag, the notation $D \vdash \pm\#q$ means that there is a proof P in D such that $P(n) = \pm\#q$ for an index n.

In the remainder, we only present the proof conditions for the positive tags: the negative ones are obtained via the principle of *strong negation*. This is closely related to the function that simplifies a formula by moving all negations to an inner most position in the resulting formula, and replaces the positive tags with the respective negative tags, and the other way around [5].

The proof conditions for $+\Delta$ describe just forward chaining of strict rules.

$+\Delta$: If $P(n+1) = +\Delta q$ then either
  (1) $q \in F$ or
  (2) $\exists r \in R_s[q]$ s.t. $\forall a \in A(r). +\Delta a \in P(1..n)$.

Literal q is definitely provable if either (1) is a fact, or (2) there is a strict rule for q, whose antecedents have all been definitely proved. Literal q is definitely refuted if (1) is not a fact and (2) every strict rule for q has at least one definitely refuted antecedent. Conceptually, strict derivations are much stronger than defeasible ones: the superiority relation plays no part in them. If we have two strict rules for opposite conclusions whose antecedents are all proven, then the logic will derive both conclusions, which signals an inconsistency within the theory itself.

The conditions to establish a defeasible proof $+d$ have a structure similar to arguments, and are formalised by the following schema.

$+d$: If $P(n+1) = +d\,q$ then either
  (1) $+\Delta q \in P(1..n)$ or
  (2) (2.1) $-\Delta{\sim}q \in P(1..n)$ and
    (2.2) $\exists r \in R_{sd}[q]$ s.t. r is applicable, and
    (2.3) $\forall s \in R[{\sim}q]$. either
      (2.3.1) s is unsupported, or
      (2.3.2) s is defeated.

Intuitively, a rule is applicable if all the literals in the antecedent have previously been proven. Clause (2.3) considers the possible counter-arguments. To derive q, each such counter-argument must be either unsupported, or defeated. A rule is unsupported if it is not possible to give a (valid) justification for at least one of the premises of the rule. The degree of provability of the conclusion we want to obtain determines the meaning of valid justification for a premise. This could vary from a derivation for the premise to a simple chain of rules leading to it. Finally, a rule is defeated if there is an applicable rule stronger than it.

By instantiating the abstract definitions of applicable, supported and defeated, the above structure defines several variants of DL. In particular, we address the distinction between *ambiguity blocking* and *ambiguity propagation*. A literal q is *ambiguous* if (i) there is a chain of reasoning that supports a conclusion q, (ii) one (chain) supporting the complementary conclusion ${\sim}q$, and (iii) the superiority relation does not resolve this conflict.

Example 4.1. *Consider the defeasible theory $D = (\emptyset, R, \emptyset)$, such that*

$$R = \{r_1 : \Rightarrow a, \quad r_2 : \Rightarrow b, \quad r_3 : \Rightarrow \neg a, \quad r_4 : a \Rightarrow \neg b\}.$$

Here a is ambiguous since both r_1 and r_3 are applicable, and there is no superiority between them.

In what follows we shall introduce two variants of DL, the first one supporting ambiguity blocking, and the second one supporting ambiguity propagation. We explain the intuitions behind the two variants by referring to Example 4.1, where a is ambiguous. In a setting where ambiguity is blocked, b is not ambiguous because rule r_2 for b is applicable, whilst r_4 for $\neg b$ is not, since we cannot prove a. On the other hand, in an ambiguity propagating setting, b is ambiguous because a is not disproved, and so the applicability of r_4 is not denied. In this way, the ambiguity is propagated to b.

The ambiguity blocking and ambiguity propagation is a clash in intuitions in non-monotonic reasoning [130]. However, [62] argues that the distinction can be used to characterise different proof standards, where ambiguity blocking corresponds to the proof standard of *preponderance of evidence* while ambiguity propagation captures the *beyond reasonable doubt* proof standard. Furthermore, there are scenarios where both intuitions are needed (for different conclusions), and the reasoning for conclusions requiring one of the two proof standard depends on conclusions obtained using the other proof standard. See [64] for the details and how to combine the two intuitions.

In the remainder, we shall use ∂ for the proof tag to indicate that a conclusion is defeasibly provable (refutable) under ambiguity blocking, and δ for the corresponding notions under ambiguity propagation.

4.2 Ambiguity Blocking Defeasible Logic

The ambiguity blocking variant of DL was introduced in [7] and is captured by the following instantiation of $+d$:

$+\partial$: If $P(n+1) = +\partial q$ then either
- (1) $+\Delta q \in P(1..n)$ or
- (2) (2.1) $-\Delta{\sim}q \in P(1..n)$ and
  - (2.2) $\exists r \in R_{sd}[q]$ s.t. $\forall a \in A(r) + \partial a \in P(1..n)$ and
  - (2.3) $\forall s \in R[{\sim}q]$ either
    - (2.3.1) $\exists a \in A(s)$ s.t. $-\partial a \in P(1..n)$ or
    - (2.3.2) $\exists t \in R_{sd}[q]$ s.t.
      $\forall a \in A(t) + \partial a \in P(1..n)$ and $t > s$.

To prove $+\partial q$, we have to show that either (1) q is already definitely provable, or (2.2) there is an applicable rule for q and (2.3) for very rule attacking q either (2.3.1) at least one antecedent has been defeasibly refuted, or (2.3.2) the rule is defeated by a (stronger) rule for q.

In other terms, a rule is applicable if all the elements of the body are defeasibly provable. A rule is unsupported if there is an element of the body that is defeasibly refuted. A rule is defeated if it is weaker than an applicable rule. We use DL(∂) to denote the ambiguity blocking defeasible logic variant.

4.3 Ambiguity Propagating Defeasible Logic

Ambiguity propagation describes a behaviour where ambiguity of a literal is propagated to dependent literals. This is achieved in DL by separating the invalidation of a counterargument from the derivation of tagged literals. To do so, another kind of conclusion, called *support* and denoted by Σ, is introduced [8].

$+\Sigma$: If $P(n+1) = +\Sigma q$ then either
  (1) $+\Delta q \in P(1..n)$ or
  (2) (2.1) $-\Delta{\sim}q \in P(1..n)$ and
    (2.2) $\exists r \in R_{sd}[q]$ s.t.
      (2.2.1) $\forall a \in A(r) + \Sigma a \in P(1..n)$ and
      (2.2.2) $\forall s \in R[{\sim}q]$ either
        $\exists a \in A(s)$ s.t. $-\delta a \in P(1..n)$, or $s \not> r$.

The condition for $+d$ is thus instantiated as follows:

$+\delta$: If $P(n+1) = +\delta q$ then either
  (1) $+\Delta q \in P(1..n)$ or
  (2) (2.1) $-\Delta{\sim}q \in P(1..n)$ and
    (2.2) $\exists r \in R_{sd}[q]$ s.t. $\forall a \in A(r) + \delta a \in P(1..n)$ and
    (2.3) $\forall s \in R[{\sim}q]$ either
      (2.3.1) $\exists a \in A(s)$ s.t. $-\Sigma a \in P(1..n)$ or
      (2.3.2) $\exists t \in R_{sd}[q]$ s.t.
        $\forall a \in A(t) + \delta a \in P(1..n)$ and $t > s$.

The idea is that a conclusion q is supported if (2.1) there is a rule for q such that (2.2.1) all the elements in the antecedent are (at least) supported, and that (2.2.2) all rules for the opposite conclusion have (at least) one premise that has been refuted, or such a rule is not stronger than the rule for q. This means that there is an undefeated argument supporting the conclusion. Then to affirm that a conclusion is provable, we have to provide an argument/rule where all the antecedents are provable, and there is no argument/rule for the opposite that is at least supported. We refer to the ambiguity propagating variant by using DL(δ).

Example 4.1 (Continued)**.** *Consider, again, the theory $D = (\emptyset, R, \emptyset)$, where*

$$R = \{r_1 : \Rightarrow a, \quad r_2 : \Rightarrow b, \quad r_3 : \Rightarrow \neg a, \quad r_4 : a \Rightarrow \neg b\}.$$

By definition of $+\partial$, we obtain the following conclusions from D: $-\partial a$, $-\partial\neg a$, $+\partial b$, $-\partial\neg b$, capturing the ambiguity blocking behaviour of DL*(∂). On the other hand, if we compute the consequences of D by using the proof conditions for Σ and δ, we obtain $+\Sigma a$, $+\Sigma\neg a$, $+\Sigma b$, $+\Sigma\neg b$ and thus also $-\delta a$, $-\delta\neg a$, $-\delta b$ and $-\delta\neg b$. In this way, we capture the ambiguity propagation feature of* DL*(δ).*

4.4 Team or Individual Defeat?

The defeasible logics defined above have the property of *team defeat*: the rules for a literal q are compared with the rules for $\sim q$. If each applicable rule for $\sim q$ is inferior to some applicable rule for q, then the rules for q, as a team, overcome the rules for $\sim q$. Thus, q is inferred. In comparison, under *individual defeat* there must be an applicable rule for q that is superior to *all* applicable rules for $\sim q$ in order to overcome the rules for $\sim q$ and infer q. Clearly, any time individual defeat overcomes the rules for $\sim q$, so does team defeat.

To get some intuition about these two forms of defeat we use a variation of an example from [7].

Example 4.2. *Consider some rules of thumb about animals and, particularly, mammals. An egg-laying animal is generally not a mammal. Similarly, an animal with webbed feet is generally not a mammal. On the other hand, an animal with fur is generally a mammal. Finally, the monotremes are a subclass of mammal. These rules are represented as defeasible rules below.*

Furthermore, animals with fur and webbed feet are generally mammals, so r_2 should overrule r_4. And monotremes are a class of egg-laying mammals, so r_1 should overrule r_3.

Finally, it happens that a platypus is a furry, egg-laying, web-footed monotreme. Is it a mammal? (That is, is $mammal(platypus)$ a consequence of the defeasible theory below?)

$$\begin{array}{llll}
r_1: & monotreme(X) \Rightarrow mammal(X) & r_3: & laysEggs(X) \Rightarrow \neg mammal(X) \\
r_2: & hasFur(X) \Rightarrow mammal(X) & r_4: & webFooted(X) \Rightarrow \neg mammal(X) \\
\multicolumn{2}{l}{r_1 > r_3} & \multicolumn{2}{l}{r_2 > r_4}
\end{array}$$

$monotreme(platypus)$	$laysEggs(platypus)$
$hasFur(platypus)$	$webFooted(platypus)$

It is obvious that all four rules are applicable to the question of $mammal(platypus)$*. Under team defeat, each rule for* $\neg mammal(platypus)$ *is overcome by some rule for* $mammal(platypus)$*, so* $mammal(platypus)$ *is inferred. However, there is no single rule for* $mammal(platypus)$ *that overcomes all rules for* $mammal(platypus)$*, so under individual defeat we cannot infer* $mammal(platypus)$ *(nor* $\neg mammal(platypus)$*).*

Thus, we see that team defeat can be useful in making a justified inference that otherwise would not be made. On the other hand, most expressions of structured argumentation employ individual defeat.

Fortunately, it is easy to adjust the inference conditions for the two logics defined above to obtain individual defeat: we simply replace the sub-conditions (2.3.2) by $r > s$. We denote the individual defeat logics by $\mathrm{DL}(\partial^*)$ and $\mathrm{DL}(\delta^*)$. For more discussion of the four variants of defeasible logic discussed here, see [23].

Finally, we consider the relationship between these logics. A series of papers [84; 85; 86; 87] investigates the relative expressiveness of variants of Defeasible Logic. In brief, two (defeasible) logics L_1 and L_2 have the same expressiveness iff the two logics simulate each other (where a defeasible logic L_2 simulates a defeasible logic L_1 if there is a polynomial time transformation T that transforms a theory D_1 of L_1 in a theory $D_2 = T(D_1)$ of L_2 such that, for any addition of facts A, all strict and defeasible conclusions of $D_1 \cup A$ are the same as those of $D_2 \cup A$ in L_1). [84; 85] provide polynomial time transformations between each of the four logics defined above.

Theorem 4.3. *[85] Each of* $\mathrm{DL}(\partial)$, $\mathrm{DL}(\delta)$, $\mathrm{DL}(\partial^*)$, *and* $\mathrm{DL}(\delta^*)$ *simulates the others.*

5 Strategic Argumentation for Defeasible Logic and Structured Argumentation

We now propose a Defeasible Logic instantiation of the games introduced in Section 2. We shall hence specialise Definitions 2.6 and 2.7 for the instance at hand, and then proceed with the formulation of two problems.

Given a defeasible theory $D = (F, R, >)$, we define the corresponding *split defeasible theory* as $SD = (F_{\mathsf{Com}}, F_{\mathsf{Pr}}, F_{\mathsf{Op}}, R_{\mathsf{Com}}, R_{\mathsf{Pr}}, R_{\mathsf{Op}}, >)$ with $F = F_{\mathsf{Com}} \cup F_{\mathsf{Pr}} \cup F_{\mathsf{Op}}$ and $R = R_{\mathsf{Com}} \cup R_{\mathsf{Pr}} \cup R_{\mathsf{Op}}$. We call the content of dispute discussed by the players the *critical literal*, and note that the arguments brought about by the players will be in the form of defeasible derivations. We assume that each player is informed about the restriction of $>$ to their private rules,

We will have three instances of the definitions of Section 2, owing to the extra expressivity of defeasible logic. Defeasible logic offers the following three ways to express a contrary to $D \vdash +d\,q$: the negation of q can be proved ($D \vdash +d\,{\sim}q$); within the logic we can prove that that $+d\,q$ cannot be proved ($D \vdash -d\,q$); and, we cannot prove $+d\,q$ ($D \not\vdash +d\,q$). Thus, if Pr wants to prove q, Op has three possible levels of opposition. The first will lead to a symmetric game, and the third to an asymmetric game. The second falls somewhere in between, and we will call it a *semi-symmetric* game. In the semi-symmetric game Op shoulders a burden of proof, but only to prove that Pr's aim cannot be proved, not to prove the negation of q.

If we consider the asymmetric case corresponds to the Scottish verdict of not proven[8] and the symmetric case corresponds to not guilty, then what is the semi-symmetric case? Technically, in defeasible logic, the distinction between semi-symmetric and asymmetric opposition is caused by a circularity or infinite regress in an argument. Abstractly, it might represent unknowability, or an incapacity of the proceedings/inference rules – inability to decide that l is not provable, even though l, in fact, is not provable (a little bit like Gödel's incompleteness theorem).

The game rules discussed in Section 2 are instantiated as follows. The parties start the game by choosing the critical literal l. Pr has the burden to prove $+d\,l$ by using the remainder of its private rules along with those that currently have been played; Op's final aim is to prove $+d\,{\sim}l$ in the symmetric version of the game, to prove $-d\,l$ in the semi-symmetric game, and simply to prevent the proof of $+d\,l$ in the asymmetric game.

Note that, when putting forward an argument, the players: (1) may propose, along with a subset of their private rules, a subset of their private facts to support such rules (see Example 5.2 at the end of this section), and (2) may play an argument whose terminal literal differs from l or ${\sim}l$ (with the aim to attack/disprove one of the premises of a rule in the proof proving $l/{\sim}l$).

As the semi-symmetric and asymmetric games differ from the symmetric one only in Op's final aim, to avoid pedantic redundancies we shall provide a single definition for the three games.

Definition 5.1 (SSA (SSSA, AsSA) Game for Defeasible Logic)**.** *Consider two players, a proponent* Pr *and an opponent* Op*, a split defeasible theory* $SD =$

[8] Roughly, under this verdict the jury considers the prosecution has not made the case for "guilty", beyond a reasonable doubt, but the defence has not made the case for "innocent". A verdict of *guilty* is given when the jury considers the prosecution has made its case, and *not guilty* when the defence has made its case. See [11] or the Wikipedia entry for *Not proven*.

$(F_{\mathsf{Com}}, F_{\mathsf{Pr}}, F_{\mathsf{Op}}, R_{\mathsf{Com}}, R_{\mathsf{Pr}}, R_{\mathsf{Op}}, >)$, and a critical literal $l \in L$.

Let F^k_{Com}, R^k_{Com}, F^k_{Pr}, R^k_{Pr}, F^k_{Op}, and R^k_{Op} denote, respectively, the common (knowledge) facts and rules, Pr*'s private facts and rules, and* Op*'s private facts and rules, after turn k. (In particular, $F^0_{\mathsf{Com}} = F_{\mathsf{Com}}$, $R^0_{\mathsf{Com}} = R_{\mathsf{Com}}$, $F^0_{\mathsf{Pr}} = F_{\mathsf{Pr}}$ $R^0_{\mathsf{Pr}} = R_{\mathsf{Pr}}$, $F^0_{\mathsf{Op}} = F_{\mathsf{Op}}$, and $R^0_{\mathsf{Op}} = R_{\mathsf{Op}}$.) The* common defeasible theory *at that point is $D^k = (F^k_{\mathsf{Com}}, R^k_{\mathsf{Com}}, >)$.*

We define a symmetric (resp. semi-symmetric, asymmetric) strategic argumentation game for Defeasible Logic *as a dialogue game where:*

1. *The players take turns. If $D^0 \vdash +d\,l$ then* Op *begins; otherwise* Pr *does so.*
2. *At turn k, if $D^{k-1} \vdash +d\,\neg l$ (resp. $D^{k-1} \vdash -d\,l$ for the semi-symmetric version, $D^k \nvdash +d\,l$ for the asymmetric version), then it is* Pr*'s turn to play, as follows*
   - Pr *advances a subset of its private facts $\Phi \subseteq F^{k-1}_{\mathsf{Pr}}$ and rules $\rho \subseteq R^{k-1}_{\mathsf{Pr}}$ so that $D^k \vdash +d\,l$. As a result*
     - *$F^k_{\mathsf{Com}} = F^{k-1}_{\mathsf{Com}} \cup \Phi$ and $R^k_{\mathsf{Com}} = R^{k-1}_{\mathsf{Com}} \cup \rho$;*
     - *$F^k_{\mathsf{Pr}} = F^{k-1}_{\mathsf{Pr}} \setminus \Phi$ and $R^k_{\mathsf{Pr}} = R^{k-1}_{\mathsf{Pr}} \setminus \rho$;*
     - *$R^k_{\mathsf{Op}} = R^{k-1}_{\mathsf{Op}}$.*
3. *At turn k, if $D^{k-1} \vdash +d\,l$, then it is* Op*'s turn to play, as follows*
   - Op *advances a subset of its private $\Phi \subseteq F^{k-1}_{\mathsf{Op}}$ and rules $\rho \subseteq R^{k-1}_{\mathsf{Op}}$ so that $D^k \vdash +d\,\neg l$ (resp. $D^k \vdash -d\,l$ for the semi-symmetric version, $D^k \nvdash +d\,l$ for the asymmetric version). As a result*
     - *$F^k_{\mathsf{Com}} = F^{k-1}_{\mathsf{Com}} \cup \Phi$ and $R^k_{\mathsf{Com}} = R^{k-1}_{\mathsf{Com}} \cup \rho$;*
     - *$R^k_{\mathsf{Pr}} = R^{k-1}_{\mathsf{Pr}}$;*
     - *$F^k_{\mathsf{Op}} = F^{k-1}_{\mathsf{Op}} \setminus \Phi$ and $R^k_{\mathsf{Op}} = R^{k-1}_{\mathsf{Op}} \setminus \rho$.*
4. *The game ends at turn $k+1$, when either (i) it is* Pr*'s turn and there is no move for* Pr *such that the common defeasible theory $D^{k+1} \vdash +d\,l$, in which case* Op *wins, or (ii) it is* Op*'s turn and there is no move for* Op *such that the common defeasible theory $D^{k+1} \vdash +d\,\neg l$ (resp. $D^{k+1} \vdash -d\,l$ for the semi-symmetric version, $D^k \nvdash +d\,l$ for the asymmetric version), in which case* Pr *wins.*

The corresponding decision problems are as follows.

SSA (SSSA, AsSA) Problem for Defeasible Logic

Let SD^k be a split defeasible theory as in Definition 5.1 after turn k, D^{k+1} be the corresponding common defeasible theory after turn $k+1$, and $l \in L$ be the critical literal.

Pr's instance for turn $k+1$: Let F^k_{Pr} and R^k_{Pr} be, respectively, the set of Pr's private facts and rules after turn k, and that the common defeasible theory assume $D^k \vdash +d\,\neg l$ (resp. $D^k \vdash -d\,l$ and $D^k \not\vdash +d\,l$ for the semi-symmetric and asymmetric problems).

Question: Do there exist Φ subset of F^k_{Pr} and ρ subset of R^k_{Pr} such that the common defeasible theory $D^{k+1} \vdash +d\,l$?

Op's instance for turn $k+1$: Let F^k_{Op} and R^k_{Op} be, respectively, the set of Op's private facts and rules after turn k, and assume that the common defeasible theory $D^k \vdash +d\,l$.

Question: Do there exist Φ subset of F^k_{Op} and ρ subset of R^k_{Op} such that the common defeasible theory $D^{k+1} \vdash +d\,\neg l$ (resp. $D^{k+1} \vdash -d\,l$ and $D^{k+1} \not\vdash +d\,l$, for the semi-symmetric and asymmetric problems)?

We explore how these games are played through an example theory that shows how different moves by the players may lead to different result of the game in the symmetric and semi-symmetric/asymmetric variants.

Example 5.2. *Consider $SD = (F_{\mathsf{Com}}, F_{\mathsf{Pr}}, F_{\mathsf{Op}}, R_{\mathsf{Com}}, R_{\mathsf{Pr}}, R_{\mathsf{Op}}, >)$ such that*

- $F_{\mathsf{Com}} = \{a\}$ *and* $R_{\mathsf{Com}} = \emptyset$*;*
- $F_{\mathsf{Pr}} = \{d\}$ *and* $R_{\mathsf{Pr}} = \{r_1 : a \Rightarrow p,\ r_2 : b, d \Rightarrow p\}$*;*
- $F_{\mathsf{Op}} = \{b, c\}$ *and* $R_{\mathsf{Op}} = \{r_3 : c \Rightarrow \neg p,\ r_4 : b \Rightarrow \neg p\}$*; and*
- $> \{(r_4, r_1), (r_2, r_4)\}$.

The critical literal is p. Pr *starts the game and can only advance r_1; the fact that b is not proven makes r_2 unsupported. Consequently, for both variants, $SD^1 \vdash +d\,p$. We now detail the different scenarios for* Op *wrt the symmetric, semi-symmetric, and asymmetric games.*

Symmetric variant. Op *considers playing r_3 but realises that is not a legal move. In fact, as r_3 is neither stronger than r_1 nor r_2, by playing it* Op *would not prove $+d\,\neg p$. By playing r_4,* Op *must also advance r_4's only premise, b ($SD^2 \vdash +d\,\neg p$ and $SD^2 \vdash +d\,b$). This makes r_2 applicable and allows* Pr *to play it and win the game.*

Semi-symmetric variant. *For this variant of the game,* Op *has the burden to prove $-d\,p$ and plays, again, r_4 ($SD^2 \vdash +d\,\neg p$ and $+d\,\neg p$ implies $-d\,p$).* Pr *can again play r_2 leading to $SD^3 \vdash +d\,p$, but now if* Op *plays r_3 (along with c), then $SD^4 \vdash -d\,p$.* Pr *has no more rules to play and this time* Op *wins.*

Asymmetric variant. *This variant of the game unfolds in the same way as the semi-symmetric variant because, for every k, $SD^k \vdash -\partial p$ implies $SD^k \not\vdash +\partial p$.*

We can modify the above example to demonstrate the distinction between the semi-symmetric and asymmetric games.

Example 5.3. *Consider the modification of Example 5.2 where r_3 in R_{Op} is replaced by*

$$r_3 : c, \neg p \Rightarrow \neg p$$

Symmetric variant. *This variant unfolds in exactly the same way as Example 5.2.* Op *does not play r_3.*

Semi-symmetric variant. *For this variant of the game,* Pr *plays r_1,* Op *plays r_4, and* Pr *plays r_2, just as in the symmetric variant. At this stage* Op *would like to play r_3 but, again, this is not a legal move: playing it would not achieve $SD^4 \vdash -\partial p$. Thus* Pr *wins.*

Asymmetric variant. *Again,* Pr *plays r_1,* Op *plays r_4, and* Pr *plays r_2. However, in this variant* Op *can play r_3, because then $SD^3 \not\vdash +\partial p$.* Pr *has no more moves, so* Op *wins. Alternatively,* Op *could simply play r_3 on her first move, to which* Pr *has no response. Thus* Op *wins without exposing r_4 and b (and without inducing* Pr *to expose r_2 and d).*

We end this subsection with a brief discussion of fact-based strategic argumentation [88], a refinement of the strategic argument games where players can only play facts. That is, strategic argument games where $R_{\mathsf{Pr}} = \emptyset$ and $R_{\mathsf{Op}} = \emptyset$. While general strategic argumentation can be a model for legal argumentation in general, this refinement reflects argument about whether regulations have been adhered to. The players are the party subject to the regulations, and the enforcement body for the regulations. R_{Com} represents the regulations, which are fixed. The players can only generate arguments by marshalling facts that support the applicability of clauses in the regulations (i.e. rules) that, in turn, support the player's contentions. This refinement could also be considered a crude partial model for pleadings in civil law (in that it elicits claimed facts from parties), although different in many ways from Gordon's Pleadings Game [58].

Although this refinement appears to simplify the reasoning required to play the game, in one sense it is no simpler [88]. Any general strategic argumentation game $SD = (F_{\mathsf{Com}}, F_{\mathsf{Pr}}, F_{\mathsf{Op}}, R_{\mathsf{Com}}, R_{\mathsf{Pr}}, R_{\mathsf{Op}}, >)$ can be reduced to the "simpler" game as follows: for each rule $r_i : \beta \Rightarrow \varphi$ in R_{Pr} we add the rule $r_i : \beta, \alpha(r_i) \Rightarrow \varphi$ to R_{Com} and add the fact $\alpha(r_i)$ to F_{Pr}, where $\alpha(r_i)$ is a new proposition. And similarly for Op. Every move in the resulting game $SD' = (F_{\mathsf{Com}}, F'_{\mathsf{Pr}}, F'_{\mathsf{Op}}, R'_{\mathsf{Com}}, \emptyset, \emptyset, >)$ corresponds to a move of SD, and vice versa.

5.1 Computational Results

We are now ready to show that deciding what arguments to play at a given turn of a dialogue game under Dung's grounded semantics is an NP-complete problem even when the problem of deciding whether a conclusion follows from an argument is computable in polynomial time.

[67] proved that this problem is NP-complete for DL with ambiguity blocking, i.e., DL(∂). We present here an outline of the proof in [88]. Theorem 5.4 is provided from the viewpoint of Pr. The same result holds for Op.

Theorem 5.4. *The SSA Problem under* DL(∂) *is* NP*-complete.*

Proof. First, the SSA Problem is polynomially solvable on non-deterministic machines. Consider a dialogue game with sets R^0_{Com}, R^0_{Pr}, R^0_{Op} and the defeasible theory $D^{i-1} = (\emptyset, R^{i-1}_{\mathsf{Com}}, >)$, the theory at turn $i-1$ of a dialogue game. An oracle guesses a set of rules $R^i \subseteq R^{i-1}_{\mathsf{Pr}}$, we compute the consequences of the argumentation theory $D^i = (\emptyset, R^{i-1}_{\mathsf{Com}} \cup R^i, >)$, and we check whether the critical literal is a positive or negative consequence. The computation of consequences can be done in polynomial time [83; 23].

Second, we reduce 3SAT to the SSA Problem, proving therefore that the problem is NP-hard. Consider a 3SAT formula $\varphi = \bigwedge_{j=1}^{n} C_j$ such that $C_j = \bigvee_{k=1}^{3} x_j^k$. R^i is defined as follows:

1. For each proposition x occurring in φ, R^{i-1}_{Pr} and R^{i-1}_{Op} both contain

$$t_x : \quad \Rightarrow x$$

$$t_{\neg x} : \quad \Rightarrow \neg x.$$

2. For each clause C_j, R^{i-1}_{Com} contains

$$r_j^k : x_j^k \Rightarrow c_j$$

   where x_j^k is either a positive literal (x), or a negative literal ($\neg x$).

3. R^{i-1}_{Com} also contains

$$r_{sat} : c_1, \ldots, c_n \Rightarrow sat.$$

For any assignment θ of values to the Boolean variables in φ, let S_θ be the set of x literals that evaluate to true under θ. And for any consistent subset S of x literals, let θ_S be an assignment that evaluates all elements of S to true. We leave it for the reader to verify that if θ satisfies φ then choosing the move S_θ wins for Pr, and if S is a winning move for Pr then S is consistent and θ_S satisfies φ. □

The same result holds for the semi-symmetric and asymmetric games.

Theorem 5.5. *The SSSA and AsSA problems under* DL(∂) *is* NP*-complete.*

Proof. The proof is essentially the same as that of Theorem 5.4 except for the case when, at turn i, Op must play. In that case, the reduction is identical to the one proposed above, with the only difference that Point 3. now also adds to R^i_{Com} the following rule

$$r_{nsat}: \quad \Rightarrow \neg sat$$

It is trivial to prove that an interpretation satisfies φ iff r_{sat} is applicable iff sat and $\neg sat$ are ambiguous. Thus φ is satisfied iff $-\partial\, sat$ is proved iff $\neg sat$ is not proved. □

While it is possible to define DL(∂) in terms of an argumentation semantics, the logic corresponding to Dung's grounded semantics is ambiguity-propagating [65; 79].

The next step is to determine the computational complexity of the problem at hand for the ambiguity propagating variant of DL. The NP-completeness of the strategic argumentation problem under DL(δ^*) follows immediately from Theorems 4.3, 5.4, and 5.5.

Theorem 5.6. *The SSA, SSSA, and AsSA problems under* DL(δ^*) *are* NP*-complete.*

We have the same results for DL(∂^*) and DL(δ).

In [79], it is shown that the conclusions of an $ASPIC^+$ argumentation theory under grounded semantics are the same as those in DL(δ^*) (after minor changes to the superiority relation).

Theorem 5.7. *[79] Given an $ASPIC^+$ argumentation theory AT, there is a defeasible theory $T(AT)$ such that p is derived under the grounded semantics from AT iff $+\delta^*p$ can be derived from $T(AT)$. Furthermore, all consequences of AT can be computed in time polynomial in the size of AT.*

Thus we can use implementations of DL(δ^*) to implement $ASPIC^+$ argumentation theories that employ the last-link ordering of arguments and the grounded semantics.

We can solve the strategic argumentation problem by non-deterministically choosing a set R^i of rules and then verifying whether the critical literal p is justified in the argumentation framework determined by D^i, or not. Further, the literals justified by the grounded semantics are computable in polynomial time, as shown above. The strategic argumentation problem is thus in NP.

Now, from Theorems 5.6 and 5.7, we obtain the following result.

Theorem 5.8. *The strategic argumentation problems under the grounded semantics are* NP*-complete.*

6 Strategic Abstract Argumentation

In this section we look beyond the grounded semantics to a wide range of other semantics for abstract argumentation frameworks. After exploring the range of dialogue games that can be played in the context of abstract argumentation, we investigate the possibilities for player aims, and identify the complexity of two computational problems related to playing strategic abstract argumentation games, for selected aims and semantics.

6.1 Strategic Argumentation in the Abstract

We formulate a split argumentation framework in this abstract sense as a tuple $(\mathcal{A}, \mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ where $\mathcal{A}_{Com}$ is a set of abstract arguments that are common knowledge to the players, $\mathcal{A}_{\mathsf{Pr}}$ ($\mathcal{A}_{\mathsf{Op}}$) is the set of arguments known to Pr (Op), and $\gg$ is the attack relation over all arguments. Each player knows $\gg$ restricted to the set of arguments the player knows. For example, Pr knows $\gg$ restricted to $(\mathcal{A}_{Com} \cup \mathcal{A}_{\mathsf{Pr}}) \times (\mathcal{A}_{Com} \cup \mathcal{A}_{\mathsf{Pr}})$. Each player has a *strategic aim* or *desired outcome* (the two terms will be treated as equivalent) that expresses their desired property of the state of the argument framework at the end of the strategic argumentation game.

A strategic abstract argumentation game consists of alternating moves by Pr and Op until one player cannot make a move. In that case the other player wins. Pr starts the game by playing a set of arguments, including a mutually agreed *critical argument* which is the subject of the two players' strategic aims[9]. By "playing a set of arguments" we refer to the transfer of a set of arguments from the player's set of arguments to $\mathcal{A}_{Com}$ such that the revised common argumentation

[9] Aims will be discussed in the next subsection.

framework $(\mathcal{A}_{Com}, \gg)$ satisfies the player's strategic aim. Thus a move by Pr replaces a split argument framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ by a new framework $(\mathcal{A}_{Com} \cup X, \mathcal{A}_{\mathsf{Pr}} \backslash X, \mathcal{A}_{\mathsf{Op}}, \gg)$, where $X \subseteq \mathcal{A}_{\mathsf{Pr}}$ is the set of arguments played by Pr in that move, and the new framework achieves Pr's strategic aim. Similarly, a move by Op transfers arguments from $\mathcal{A}_{\mathsf{Op}}$ to $\mathcal{A}_{Com}$. Clearly, if $\mathcal{A}_{\mathsf{Pr}}$ or $\mathcal{A}_{\mathsf{Op}}$ is finite then the game terminates. We will only consider games where $\mathcal{A}_{Com}$, $\mathcal{A}_{\mathsf{Pr}}$ and $\mathcal{A}_{\mathsf{Op}}$ are finite.

Thus, a strategic abstract argumentation game is a dialogue game played by two players (Pr and Op). Let *conc* map arguments to distinct propositions, and let φ be the conclusion of the critical argument. Then the game is an asymmetric strategic argumentation game, as defined in Definition 2.7, where "φ is accepted" is defined as: Pr's aim wrt the critical argument is satisfied.

We assume that the players agree on what is an argument, and whether one argument attacks another. This is implicit in the formulation as a split argumentation framework. But, in theory, there is no reason why the two players should employ the same semantics when they play a strategic argumentation game. For example, Pr might formulate her aim in terms of the preferred semantics, while Op's aim is expressed in terms of the eager semantics. Indeed, it is quite reasonable that different players might perceive the world differently. This is no impediment to the players playing a strategic argumentation game, since the definition of the game only describes moves a player may make, and not the interpretation she puts on the game.

However, there has not been any work on such situations. This is not so surprising when we consider that strategic argumentation is primarily treated as an adversarial game. Real world situations that are modelled by strategic argumentation may need the presence of an adjudicator to enforce any conclusions that result from the game. Such an adjudicator might bring their own perceptions and semantics to the game. Thus, playing in a common semantics could be considered as both players adopting the adjudicator's view of the world.

Similarly, there is no *prima facie* reason why the two players should focus on a single critical argument, rather than have individual, separate foci. The literature has rarely addressed this possibility ([71] is an exception). However, once we assume that the players agree on a focus, the use of a single critical argument for each player implies no loss of generality. Straightforward constructions can map a disjunction or conjunctions of arguments to a single argument in most semantics[10]. In particular, the arguments supporting the same conclusions can be united in a single argument.

In any case, many of the computational issues discussed in this and the next section depend only on the semantics and the player's aim, and so are still applicable to these less-well-studied forms of strategic argumentation.

[10] For example, see Proposition 2 of [90].

Finally, even when addressing the same semantics and critical argument, there is some freedom in the strategic aims of the two players. At one extreme the players might have the same aim and, on the other extreme, have diametrically opposed aims. In between these extremes the players might have different but compatible aims, or have incompatible aims. Aims are discussed in detail in the next subsection. In this chapter we assume that the two players have incompatible aims: it is not possible for both players to achieve their aims simultaneously.

In previous sections we have discussed both symmetric and asymmetric forms of strategic argumentation. In abstract argumentation there is no explicit notion of conclusion and, therefore, no notion of an argument supporting the negation of the conclusion of another. Consequently, symmetric strategic argumentation is not available, in general. We will focus on asymmetric strategic argumentation. That is, whatever Pr's aim is, Op's aim is to prevent it.

In summary, a *strategic abstract argumentation dialogue game* consists of a split abstract argumentation framework, a critical argument, an abstract argumentation semantics, and aims for both Pr and Op. The *play* of the game is a sequence of moves such that each player leaves the game in a state where her strategic aim is satisfied.

6.2 Players' Aims

The range of strategic aims a player might have is limited under the grounded semantics. But once we consider semantics with multiple extensions a player has a much wider range.

Initially, work on abstract argumentation focussed on credulous and skeptical acceptance. An argument a in argumentation framework AF under semantics σ is *credulously accepted* if it is labelled `in` in at least one σ-extension. a is *skeptically accepted* if it is labelled `in` in every σ-extension. These two statuses were inherited from the field of non-monotonic reasoning.

[142] extended this work with the notion of justification status. The justification status of an argument a in an argument framework AF is the set of labels a receives in complete extensions. Thus a justification status is a subset of $\{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$. In general this might lead to $2^3 = 8$ different statuses, but only 6 are possible for the complete semantics [142]. Obviously, this approach can be extended to any extension-based semantics [44].

[91; 90] further extended the range of argument statuses to the following, casting these as possible aims of a proponent

1. **Existential:** a is labelled `in` in at least one σ-extension

2. **Universal:** a is labelled `in` in all σ-extensions

3. **Unrejected:** a is not labelled `out` in any σ-extension
4. **Uncontested:** a is labelled `in` in at least one σ-extension and is not labelled `out` in any σ-extension
5. **Plurality:** a is labelled `in` in more σ-extensions than it is labelled `out`
6. **Majority:** a is labelled `in` in more σ-extensions than it is not labelled `in`
7. **Supermajority:** a is labelled `in` in at least twice as many σ-extensions than it is not labelled `in`

The last three are called *counting aims*, distinct from the first four which are based on zero/non-zero number of labels, like the justification statuses[11]. In addition, the negation of such conditions and their dual (exchanging the role of `in` and `out`), which are plausible aims for the opponent, have also been considered [90].

But clearly there are many more possibilities. Each of the first four strategic aims can be formulated as a disjunction of justification statuses. So we might consider any disjunction of justification statuses as a potential strategic aim. This would give us $2^8 = 256$ strategic aims. Many of these will be unrealizable under some semantics and/or unrealistic in practice. Under the stable semantics, aims that the argumentation framework has at least one extension or has no extension are also sensible. Further possibilities are aims such as: a is accepted in at least 2 extensions or is universally accepted. There are also many variations possible for the counting aims. For example, [91] contemplates a weighting on all extensions, with the arguer's aim that the sum of the weights of extensions in which a is labelled `in` is greater than the sum of weights of the remaining extensions.

Some of the aims seem similar to the ideas behind proof standards that are formalized in [60], although those proof standards are formalized in a very different setting. The Existential aim is similar to a *scintilla of evidence*, the Majority and Supermajority correspond to *preponderance of the evidence* and *clear and convincing evidence*, respectively, while the Uncontested aim is like *beyond a reasonable doubt.*[12] The Universal aim corresponds to *beyond a doubt*, in the phrasing of [51].

There are some obvious close relationships between these different concepts. a is skeptically accepted iff a has justification status $\{$`in`$\}$ iff a satisfies the Universal aim. Similarly, a is credulously accepted iff a's justification status contains `in` iff

[11] A counting utility function was defined in [128], but it counts the number of desired conclusions that appear in all σ-extensions rather than counting the number of σ-extensions in which a conclusion appears.

[12] The Uncontested aim is also similar to the notion of *argumentative inference* in paraconsistent reasoning from maximally consistent sets [20].

	$\mathcal{GR}$	$\mathcal{ST}$	$\mathcal{CO}$	$\mathcal{PR}$	$\mathcal{SST}$	$\mathcal{EA}$	$\mathcal{ID}$
Existential	in P	NP-c	NP-c	NP-c	Σ_2^p-c	Π_2^p-c	in Θ_2^p
Universal	in P	coNP-c	in P	Π_2^p-c	Π_2^p-c	Π_2^p-c	in Θ_2^p
Unrejected	in P	coNP-c	coNP-c	coNP-c	Π_2^p-c	Σ_2^p-c	in Θ_2^p
Uncontested	in P	coNP-c	D^p-c	D^p-c	D_2^p-c	Π_2^p-c	in Θ_2^p
Plurality	in P	PP-c	PP-c	in $\mathrm{PP}^{\mathrm{NP}}$	in $\mathrm{PP}^{\mathrm{NP}}$	Π_2^p-c	in Θ_2^p
Majority	in P	PP-c	PP-c	in $\mathrm{PP}^{\mathrm{NP}}$	in $\mathrm{PP}^{\mathrm{NP}}$	Π_2^p-c	in Θ_2^p
Supermajority	in P	PP-c	PP-c	in $\mathrm{PP}^{\mathrm{NP}}$	in $\mathrm{PP}^{\mathrm{NP}}$	Π_2^p-c	in Θ_2^p

Table 1: Complexity of Aim Verification problem for selected strategic aims and semantics [90]. For a complexity class $\mathcal{C}$, $\mathcal{C}$-c denotes that the problem is complete for $\mathcal{C}$.

a satisfies the Existential aim. a satisfies the Unrejected aim iff a has justification status $\{\texttt{in}\}$, $\{\texttt{undec}\}$ $\{\texttt{in}, \texttt{undec}\}$, or $\emptyset$. a satisfies the Uncontested aim iff a has justification status $\{\texttt{in}\}$ or $\{\texttt{in}, \texttt{undec}\}$. Also, a satisfies the Uncontested aim iff a satisfies the Existential and Unrejected aims.

Furthermore, when a semantics consists of a single extension (in particular, the grounded semantics) credulous and skeptical acceptance are identical, there are only three possible justification statuses for an argument ($\{\texttt{in}\}$, $\{\texttt{undec}\}$, and $\{\texttt{out}\}$), and all but the Unrejected aim, of those listed, are identical. In summary, a unitary semantics greatly simplifies analysis of player aims.

Thus, as we consider a wider range of semantics we must also address a wider range of player aims.

6.3 Computational Problems

We can break down the play of a game into two computational problems: recognising whether (or not) an argumentation framework satisfies a given aim, which is called the Aim Verification problem, and determining what arguments to play in order to leave the game in a state where the given aim is satisfied, the decision form of which is called the Desired Outcome problem. These problems will be different for the different players, because they have different aims.

The problem of verifying that an aim is satisfied by some state of strategic argumentation is a fundamental part of each move in a game.

The Aim Verification Problem

Instance A split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$, an argumentation semantics, a critical argument $a \in \mathcal{A}_{Com}$, and an aim.

Question Is the aim concerning the critical argument satisfied under the given semantics by the argumentation framework $(\mathcal{A}_{Com}, \gg)$?

The complexity of this problem, for a selection of semantics and aims, is presented in Table 1. Given Pr's aim, the complexity of verifying Op's aim is the complement of the complexity of Pr's aim.

These results are derived from existing work on the complexity of credulous and skeptical acceptance in abstract argumentation frameworks for the various semantics (see, for example, [43; 141]), and relations between the different aims (Proposition 3 of [90]). For example, the Uncontested aim is the conjunction of Existential and Unrejected, where the latter is the dual of the negation of Existential. Under the (say) preferred semantics, credulous acceptance is NP-complete. Thus the complexity of Uncontested is a conjunction of NP and coNP, which gives us D^p. Completeness is a straightforward reduction.

For the counting aims, clearly the complexity is in PP^V, where V is the complexity of verifying that a set of arguments forms an extension of the appropriate type[13]. The lower bound for the stable semantics is obtained by reduction from the MAJSAT problem, and the complete semantics is treated by reduction from the stable semantics.

Table 1 only addresses a selected set of strategic aims. When a player has such an aim, their opponent will usually have a quite different aim, one not mentioned in the table. Since we are considering only games where the opponent's aim is the complement of the proponent's aim, the complexity of the Aim Verification problem for Op is the complement of the complexity of the Aim Verification problem for Pr. Thus, for example, under the complete semantics, if Pr has the Existential aim then aim verification for Pr is NP-complete, and aim verification for Op is coNP-complete. In general, though, when the opponent's aim is not the complement of the proponent's, the complexity of the two problems is not so directly related.

The Desired Outcome problem [91] is the problem that a player must solve at each step of a strategic abstract argumentation game. It involves identifying that the player has a legal move, leaving the state of the game in a desired state.

The Desired Outcome Problem for Pr

Instance A split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ an argumenta-

[13] There has been some work done on counting extensions, both on the complexity of counting and identifying tractable cases [14; 53]. These works focus on absolute counting, rather than comparing counts (as in the counting aims), so the results are presented in terms of #P rather than PP. Nevertheless, the complexity results are comparable to those for the counting aims in the Aim Verification problem.

	$\mathcal{GR}$	$\mathcal{ST}$	$\mathcal{CO}$	$\mathcal{PR}$	$\mathcal{SST}$	$\mathcal{EA}$	$\mathcal{ID}$
Existential	NP-c	NP-c	NP-c	NP-c	Σ_2^p-c	Σ_3^p-c	Σ_2^p-c
Universal	NP-c	Σ_2^p-c	NP-c	Σ_3^p-c	Σ_3^p-c	Σ_3^p-c	Σ_2^p-c
Unrejected	NP-c	Σ_2^p-c	Σ_2^p-c	Σ_2^p-c	Σ_3^p-c	Σ_2^p-c	Σ_2^p-c
Uncontested	NP-c	Σ_2^p-c	Σ_2^p-c	Σ_2^p-c	Σ_3^p-c	Σ_3^p-c	Σ_2^p-c
Plurality	NP-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	Σ_3^p-c	Σ_2^p-c
Majority	NP-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	Σ_3^p-c	Σ_2^p-c
Supermajority	NP-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	$\mathrm{NP}^{\mathrm{PP}}$-c	Σ_3^p-c	Σ_2^p-c

Table 2: Complexity of the Desired Outcome problem for Pr, for selected aims and semantics [91; 90; 89]. For a complexity class $\mathcal{C}$, $\mathcal{C}$-c denotes that the problem is complete for $\mathcal{C}$.

tion semantics, a critical argument $a \in \mathcal{A}_{Com}$, and an aim for Pr.

Question Is there a set $I \subseteq \mathcal{A}_{\mathsf{Pr}}$ such that Pr's aim with respect to the critical argument is achieved in the argumentation framework $(\mathcal{A}_{Com} \cup I, \gg)$?

This problem is a generalization of the strategic argumentation problem, as defined in Section 2, which is restricted to accepting the critical argument under the grounded semantics.

It is not difficult to see that the Desired Outcome problem can be solved by a non-deterministic algorithm with an oracle for the Aim Verification problem with Pr's aim. The complexity of this problem, for a selection of semantics and aims, is presented in Table 2.

The complement of this problem decides when Pr does not have a next move. The complexity of this complement problem is clearly the complement of the complexity of the Desired Outcome problem.

We can define the Desired Outcome problem for Op similarly, based on Op's aim. The complexities of the Desired Outcome problems for Pr and Op are not as directly related as is the case for aim verification.

Showing the presence of the Desired Outcome problem in the appropriate complexity class is comparatively straightforward, but showing it is complete in the class requires the construction of argumentation frameworks that extend those used for credulous or skeptical acceptance. An example construction for the Desired Outcome problem with the Universal aim under the stable semantics is shown in Figure 2. In this case the problem is Σ_2^p-complete, so we reduce the satisfiability of $\exists X \forall Y \; \psi$ (where ψ is in DNF) formulas to this problem. The diagram has three parts: on the

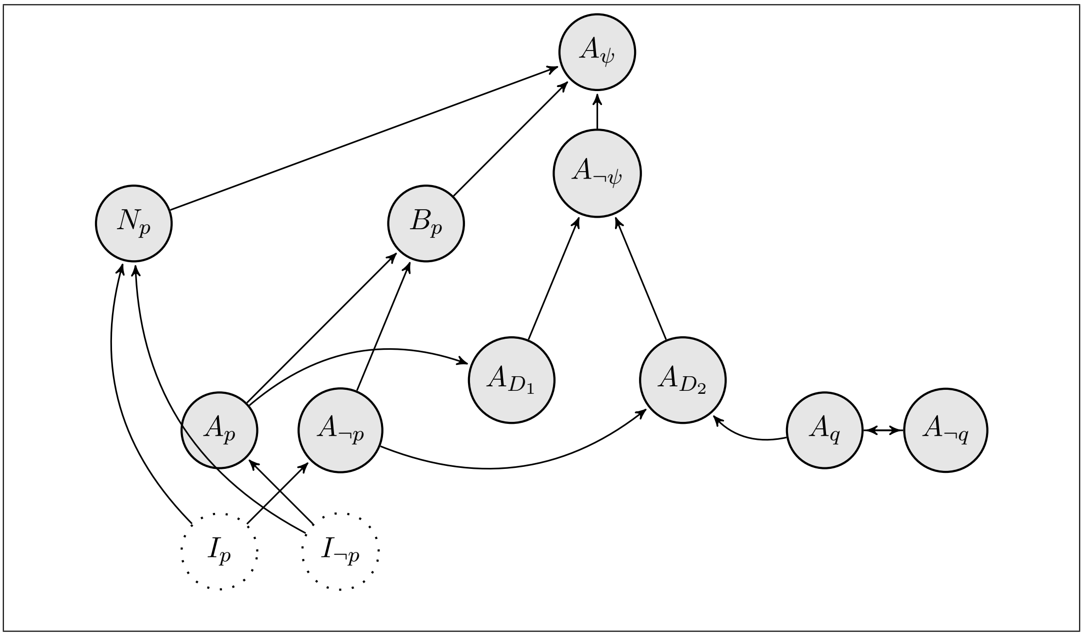


Figure 2: Example construction for the Desired Outcome problem with the Universal aim under the stable semantics

left is the representation of a variable p in X, in the middle is the representation of ψ, and on the right is the representation of a variable q in Y.

In the diagram, the grey nodes are arguments in $\mathcal{A}_{Com}$, and the white nodes (I_p and $I_{\neg p}$) are arguments in $\mathcal{A}_{\mathsf{Pr}}$. $\gg$ is described by the directed edges. ($\mathcal{A}_{\mathsf{Op}}$ is irrelevant to this problem.) Intuitively, an argument A_s (where s is a literal) accepted in a stable extension corresponds to the literal s being true. The critical argument is A_ψ, and Pr must move so that A_ψ is accepted in all stable extensions. The construction ensures that if Pr plays either both I_p and $I_{\neg p}$ or neither I_p nor $I_{\neg p}$ then either B_p or N_p is accepted and A_ψ is rejected in all stable extensions. Thus, Pr must play only one argument for each p, and this ensures only one of A_p and $A_{\neg p}$ can be accepted. This part of the construction is common to all reductions.

In the diagram, the formula is $\exists p \forall q \; \neg p \vee (p \wedge \neg q)$. It is represented in a slightly roundabout way. The treatment of variables q in Y ensures that both stable extensions containing A_q (i.e. q is true) and stable extensions containing $A_{\neg q}$ (i.e. q is false) are generated. A more formal description of this construction is in the proof of Theorem 7 of [91].

Given a specific game, we write AV_{Pr} (AV_{Op}) for the Aim Verification problem for Pr's (respectively, Op's) aim. Similarly, DO_{Pr} (DO_{Op}) denotes the Desired Outcome problem for Pr (respectively, Op).

Play begins by Pr playing a set of arguments, including the critical argument, and proceeds by Op and Pr alternately solving their Desired Outcome problem and playing the corresponding set of arguments. Play can extend for, at most, $\min(|\mathcal{A}_{\mathsf{Pr}}|, |\mathcal{A}_{\mathsf{Op}}|)$ rounds before play terminates, when one player does not make a move. Thus, play for Pr, over the entire game, has a computational cost in $\mathrm{P}^{DO_{\mathsf{Pr}}}$ while the cost of play for Op is in $\mathrm{P}^{DO_{\mathsf{Op}}}$ [90].

The Aim Verification problem is of little interest for the concrete forms of strategic argumentation discussed in Section 5. In those cases the inference problem is polynomial [83; 23]. Consequently, verifying any of the aims or justification statuses is also polynomial. The Desired Outcome problem corresponds to the SSA, SSSA and AsSA problems in Section 5: they represent the computational cost of making a move, in their respective games. In the case of structured arguments, conceptually the argumentation theory gives rise to an argumentation framework, which can then be interpreted in a chosen semantics. However, this does not mean that the NP-completeness for grounded semantics in Table 2 can be used to prove Theorem 5.8. The difficulty is that there might be greater than polynomially many arguments generated from the argument theory.

7 Corruption in Argumentation

When a game such as strategic argumentation is a model of a real-world situation, we must acknowledge the extra forces and influences that operate upon a player, beyond those of the specific role they have in the game. Often a player is assumed to have no motivations beyond performing their role and conforming to the rules of the game, but this is a rather simplistic view. While games do have rules, we need to consider the possibility that a player breaks the rules, or "cheats".

The context of the game is important in this regard. Organizations have many mechanisms to discourage the risk of corruption of their processes by the individuals performing these processes: managerial oversight, transparency through audit trails, the presence of co-workers, random inspections, etc. Society, as a whole, provides an entire justice system to enforce the rules the society considers important, and to detect and punish violations. When these mechanisms are not available, or are limited, how can we discourage rule-breaking?

[16] proposed an answer to this question in the case of vote manipulation: if the computational difficulty of determining what an individual must do to alter the result of an election is too great, a potential vote manipulator may be discouraged from the manipulation, even though he has the opportunity to do it. They called this

concept *computational resistance to strategic manipulation.* This insight has spawned a whole subfield of computational social choice [29]. In this section we describe the application of these ideas to strategic argumentation.

Throughout this section, we consider that players are engaged by a client to play the game. A player is expected to adhere to the rules of the game and, in particular, play the game to win for her client. However, while the client is invested in winning the game, the player has various competing incentives. These are the source of the corruption we consider. A player might cheat on behalf of her client, or might sacrifice her client's chances for other incentives. This issue is known in management theory as the *principal-agent problem* or the *agency problem* [47].

7.1 Corruption and Resistance

Strategic argumentation has relatively few rules, though some are implicit rather than explicitly stated. The players must take turns, but violations of this rule are obvious and, anyway, offer no advantage to the players. A player must make a move if one is available to her. This rule is implicit in the assumption that the player will play her role properly. Such a rule is difficult to enforce without knowledge of the player's arguments. The player's arguments are assumed to be private, but this is also difficult to enforce. We will focus on violations of this privacy[14].

We consider two forms of corruption. The first, *collusion*, arises when one player induces the other to let her win. Such behaviour on its own is straightforward, though illicit, and does not, as such, appear in the game. But it is complicated by the desire of the guilty parties not to be detected. Thus, colluding players must not only ensure the "right" player wins, they must also make sure that an external observer cannot distinguish the collusive play from normal play. If the work to ensure this is computationally more difficult than simply playing the game honestly, then we consider the game to be *resistant to collusion.*

The following example is an instance of collusion.

Example 7.1. *Consider the strategic argument game depicted in Figure 3, where vertices are arguments (grey if they can be played by* Pr*, white for* Op*) and edges are attacks of one argument on another. For concreteness, we assume that we employ the grounded semantics and* Pr*'s strategic aim is that argument* A *is accepted. Normal play would proceed as follows:* Pr *plays* A*,* Op *plays* B_1 *(thus defeating* A*),* Pr *plays* C *(restoring* A *by defeating* B_1*), and* Op *plays* D *(defeating* C*, and allowing* B_1 *to defeat* A*). Thus, normally,* Pr *loses.*

[14] Earlier works that consider privacy include [32] and [105], which have a focus on minimizing the exposure of a player's arguments during play, rather than the loss of privacy by corruption.

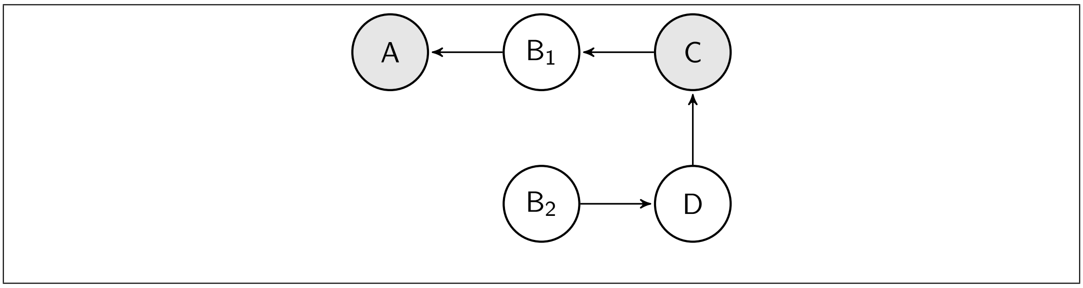


Figure 3: A strategic argumentation game. An argument is grey if it can be played by Pr and white if it can be played by Op.

However, Pr *and* Op *might collude to ensure* Pr *wins by playing as follows:* Pr *plays* A*,* Op *plays* B_1 and B_2*, and* Pr *plays* C *(restoring* A*).* Pr *now wins because* Op *has no effective move: to play* D *would have no effect because it is defeated by* B_1*.*

This example also serves to show the difference between collusion and an omniscient argumentation framework $(\mathcal{A}_{Com} \cup \mathcal{A}_{\mathsf{Pr}} \cup \mathcal{A}_{\mathsf{Op}}, \gg)$*. Under any completist semantics,* A *is accepted in the omniscient argumentation framework, but if* Pr *and* Op *collude to ensure* Op *wins they can do so by following the normal play above.*

The second form of corruption, *espionage*, occurs when, through some means, one player gains knowledge of the other player's arguments. Again, this act is not apparent in the game, but it requires work to develop a strategy, based on that knowledge, to defeat the other player. If this is computationally more difficult than playing the game honestly, then we consider the game to be *resistant to espionage.*

In Example 7.1, the corrupt sequence of moves might also occur if Op committed espionage on Pr in order to ensure Pr wins.

For both forms of resistance, we need to clarify what "computationally more difficult" means. Computational difficulty will be measured in terms of a hierarchy of complexity classes where, although one class might be contained in another, it is often not known that the two classes are distinct. However, if the two classes were equal then part of the (say) polynomial complexity hierarchy would collapse, and this is commonly believed by complexity theorists not to happen. Thus "computationally more difficult" is subject to this commonly-believed assumption. For counting aims we are dealing with the counting polynomial hierarchy, and the corresponding assumption is messier. The topic is less investigated and there are some collapses known within the counting hierarchy. However, those collapses do not affect the containments

$$\mathrm{P}^{\mathrm{PP}} \subseteq \mathrm{NP}^{\mathrm{PP}} \subseteq \mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}} \subseteq \mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}} \subseteq \cdots \subseteq \mathrm{PSPACE}$$

The assumption that these containments are strict is the basis of resistance for counting aims.

Inherent in the resistance approach to corruption is the assumption that players will be effectively penalised if their corruption is detected. This assumption relies on issues of governance, lasting identification of the players, and enforcement and scale of penalties, among others. But these issues depend on the context of the game and are beyond the scope of this chapter.

7.2 Computational Problems

We now consider the computational problems that must be solved by players in order to exploit corruption.

Colluders need to to construct an alternating sequence of moves that ends with Pr winning, that is, with Op unable to make a move. This is formalized as follows.

The Winning Sequence Problem for Pr

Instance A split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ and a desired outcome for Pr.

Question Is there a sequence of moves such that Pr wins?

A similar problem arises when the colluders wish to ensure that Op wins.

The problem for Pr can be solved by nondeterministically generating a sequence of moves, verifying that each move achieves the aim for its player, and verifying that Op has no further move. That is, it can be solved in NP with oracles for AV_{Pr}, AV_{Op} and (the complement of) DO_{Op}. $AV_{\mathsf{Op}} = coAV_{\mathsf{Pr}}$, since we assume Pr and Op have complementary aims, so the larger of $\mathrm{NP}^{AV_{\mathsf{Pr}}}$ and $\mathrm{NP}^{DO_{\mathsf{Op}}}$ is an upper bound for this problem.

In the case of espionage, one player, say Pr, knows her opponent's arguments $\mathcal{A}_{\mathsf{Op}}$ and desires a strategy that will ensure Pr wins, no matter what moves Op makes. A *strategy* for Pr in a split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ is a function from a set of common arguments to the set of arguments to be played in the next move. A sequence of moves $S_1, T_1, S_2, T_2, \ldots$ resulting in common arguments $\mathcal{A}^{\mathsf{Pr},1}_{Com}, \mathcal{A}^{\mathsf{Op},1}_{Com}, \mathcal{A}^{\mathsf{Pr},2}_{Com}, \mathcal{A}^{\mathsf{Op},2}_{Com}, \ldots$ is *consistent with* a strategy s for Pr if, for every j, $S_{j+1} = s(\mathcal{A}^{\mathsf{Op},j}_{Com}, \mathcal{A}_{\mathsf{Pr}})$. A strategy for Pr is *winning* if every valid sequence of moves consistent with the strategy is won by Pr.

The Winning Strategy Problem for Pr

Instance A split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$ and a desired outcome for Pr.

	$\mathcal{GR}$	$\mathcal{ST}$	$\mathcal{CO}$	$\mathcal{PR}$	$\mathcal{SST}$	$\mathcal{EA}$	$\mathcal{ID}$
Existential	Σ_2^p-c	Σ_3^p-c	Σ_3^p-c	Σ_3^p-c	Σ_4^p-c	Σ_3^p-c	Σ_3^p-c
Universal	Σ_2^p-c	Σ_2^p-c	Σ_2^p-c	Σ_3^p-c	Σ_3^p-c	Σ_3^p-c	Σ_3^p-c
Unrejected	Σ_2^p-c	Σ_2^p-c	Σ_2^p-c	Σ_2^p-c	Σ_3^p-c	Σ_4^p-c	Σ_3^p-c
Uncontested	Σ_2^p-c	Σ_2^p-c	Σ_3^p-c	Σ_3^p-c	Σ_4^p-c	Σ_3^p-c	Σ_3^p-c
Plurality	Σ_2^p-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	Σ_3^p-c	Σ_3^p-c
Majority	Σ_2^p-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	Σ_3^p-c	Σ_3^p-c
Supermajority	Σ_2^p-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	Σ_3^p-c	Σ_3^p-c

Table 3: Complexity of the Winning Sequence problem for Pr for selected aims and semantics [90].

Question Is there a winning strategy for Pr that satisfies the standards?

There is also, of course, the corresponding problem for Op which arises when Op conducts the espionage.

The following result shows that the Winning Strategy problem is PSPACE-complete for all completist semantics and all the aims discussed in this chapter. This is not surprising since, as a result of the espionage, Pr is essentially playing a complete knowledge game and such games are known to be PSPACE-hard, in general.

Theorem 7.2. *[90] Consider any completist semantics for abstract argumentation, and any of the above aims for* Pr*.*

The Winning Strategy problem is PSPACE-complete.

This theorem applies both to espionage by Pr and espionage by Op. The constructed argumentation framework for this proof is well-founded. Consequently the construction serves for all completist semantics.

7.3 Audit: Standards and Compliance

To investigate collusion, we need to understand what "normal play" looks like and how to recognise it. [92] proposes that we view this as a matter of audit, with an external body setting standards for play and testing for compliance. In this view there can be multiple standards. We have already seen one standard: that a player must make a move, if she has one (we will call this the *compulsory move* standard). Consequently, colluding players must arrange their play to ensure that the designated loser has no possible moves at the end of the game. Earlier work [91; 90; 89] implicitly operated under this standard.

However, this standard fails to address obvious collusion, like that in Example 7.1. Thus, additional standards are required. However, a standard can only be justified if it does not interfere with honest play. That is, a player should never face a choice between following the standard and improving her chances of winning. Otherwise, any violation of the standard can be explained away as an attempt to improve those chances.

It is clear that the problem in Example 7.1 stems from Op playing B2. But it is not clear what is an appropriate standard that would prevent this move. Several possibilities suggest themselves:

(1) A player should not play an argument that attacks one of her own (unplayed) arguments, thus causing a self-inflicted injury.

(2) A player should play the smallest number of arguments to achieve her aim[15].

(3) A player should play a subset-minimal set of arguments that achieve her aim.

(1) is clearly too strong to be a standard. If, in Example 7.1 (Figure 3), B_1 also attacked B_2 then following this standard would cause Op to lose immediately. However, when the omniscient argumentation framework is known to a player, [128] prove that this standard (which they call the overcautious selection function) is dominant. Unfortunately, a player cannot be expected to know the omniscient argumentation framework.

(2) is more plausible, but consider the following example from [92].

Example 7.3. *Consider the strategic argumentation game in Figure 4, and play that proceeds as follows:* Pr *plays* A*,* Op *plays* B_1 *and* B_2*, and* Pr *plays* C_1 *and* C_2*. At this stage O must attack both* C_1 *and* C_2*, and she has two alternatives: (1) play* E*, which attacks both* C_1 *and* C_2*, or (2) play both* D_1 *and* D_2*, each attacking one of the* C *arguments. Clearly (1) is the minimum cardinality move. However,* Pr *then responds with* F*, and wins. In (2), the play of* F *is insufficient for* Pr*, since* B_2 *remains undefeated. Hence* Op *wins.*

Thus minimum cardinality is not suitable as a standard, because it can prevent a winning move.

However, [92] showed that (3) is compatible with normal play: every non-minimal move is dominated by a minimal move[16]. Thus the requirement to play only subset-minimal moves is a suitable standard. It remains open whether there are other standards that could be applied.

[15] This is similar to the heuristic of [105], though the details of the game are different.

[16] Previous work addressing redundancy or relevance in argumentation includes [54; 98].

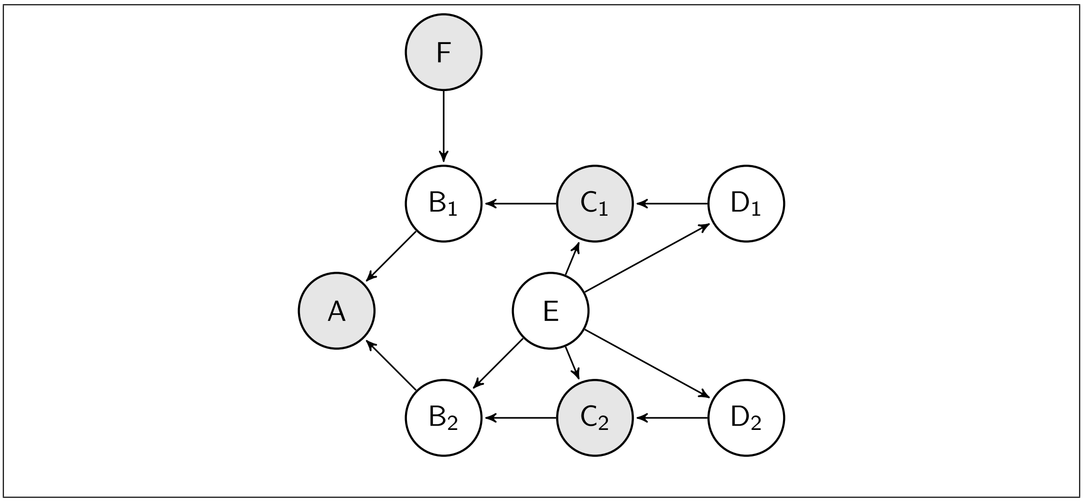


Figure 4: Split argumentation framework demonstrating non-dominance of minimum cardinality moves.

In addition, we need to consider how play can be verified as compliant with a standard. This involves issues of which data need to be accessed by the auditor, as well as the computational difficulty of verifying compliance

In terms of accessibility, all that an auditor needs for subset-minimality is an ability to inspect the initial $\mathcal{A}_{Com}$, the sequence of moves, and $\gg$ restricted to the current $\mathcal{A}_{Com}$, all of which can be considered public information. On the other hand, to verify the compulsory move standard requires knowledge of the player's arguments, which is private. Thus an auditor verifying both standards needs access to all aspects of a split argumentation framework. (However, each client might be in a position to audit the compulsory move standard, which would allow the player's arguments to be kept private from the auditor.)

For the auditor, the cost of verifying compliance with the subset-minimality standard involves polynomial many solutions of the Minimal Move problem (see next subsection) for Pr, and the same for Op. In comparison, the compulsory move standard requires a $coDO_{\mathsf{L}}$ check, where L is the loser of the game, to verify that there is no move for L left to play.

For the players, compliance with the subset-minimality standard increases the difficulty of making a move. Not only must they find a move, they must also verify that it is minimal. It also increases the cost to players exploiting collusion: they must arrange the game so that their designated player wins, but also ensure that each move is minimal. Furthermore, one easy avenue for exploiting collusion has

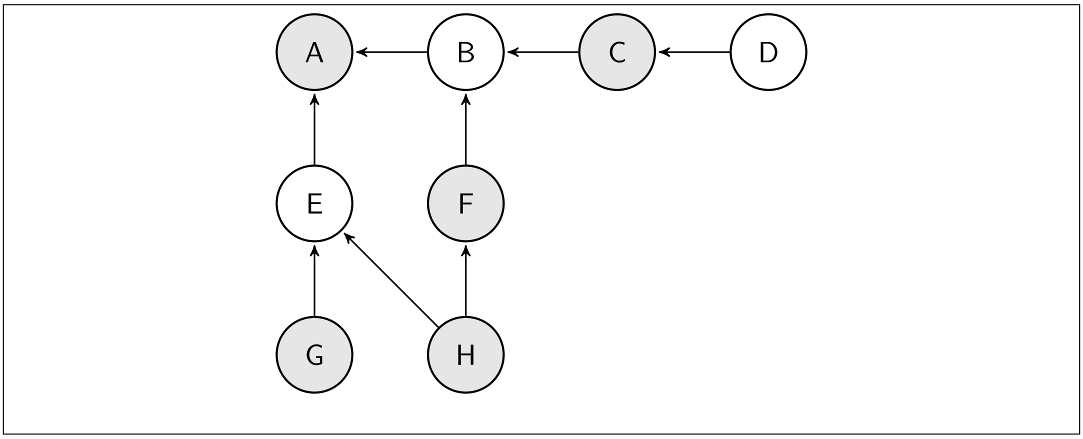


Figure 5: A strategic argumentation game demonstrating weakness of the compulsory move and subset-minimality standards.

been eliminated. Consequently, there are games (like Example 7.1) where compliance with both standards ensures that exploitation of collusion cannot be disguised as normal play.

This leads to some questions. Are these two standards sufficient to prevent the disguise of collusion? If not, can we add standards to achieve this goal? Unfortunately, the answer to the first question is no, as the following example shows.

Example 7.4. *Consider the strategic argument game depicted in Figure 5, where arguments in* $\mathcal{A}_{\mathsf{Pr}}$ *are grey and arguments in* $\mathcal{A}_{\mathsf{Op}}$ *are white, and* A *is the critical argument. If* Pr *refrains from playing* H *then* Pr *will win, since the two arguments attacking* A *(*B *and* E*) can be attacked by* Pr*'s arguments* F *and* G*, which cannot be attacked by* Op*. For example, the sequence of moves:* $\mathsf{A}, \mathsf{B}, \mathsf{F}, \mathsf{E}, \mathsf{G}$ *results in* Pr *winning.*

On the other hand, the sequence of moves: $\mathsf{A}, \mathsf{E}, \mathsf{H}, \mathsf{B}, \mathsf{C}, \mathsf{D}$ *results in* Op *winning. Thus,* Pr *and* Op *can collude to ensure* Op *wins.*

This example suggests that a variation of (1) above might be needed to detect collusion more thoroughly. Which leads us to the second question: is it possible to impose enough justified standards that no collusion can be disguised as compliant play? Again the answer is no.

Consider the argumentation game in Figure 6 under the grounded semantics, where A is the critical argument. After Pr plays A, Op has the choice of playing B or C. Depending on this choice, either Pr or Op will win. If Pr and Op collude they can determine the outcome, but any real restriction imposed by a standard will restrict

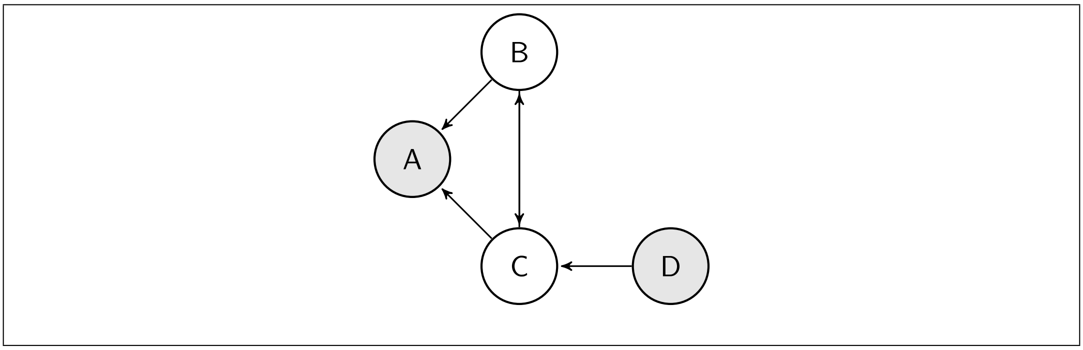


Figure 6: A strategic argumentation game demonstrating that no accumulation of justifiable standards can make all collusion detectable.

to one possible outcome, so it cannot be a justified standard. Thus any collusion in this game cannot be detected by imposing justified standards.

Hence, we see that collusion cannot be prevented simply by imposing more and more standards. We must continue to rely on computational difficulty to discourage corruption.

We now take a stab at formalizing these considerations. A *standard* is a restriction on moves a player may make. More precisely, a standard is a function from a player's aim, her private set of arguments ($\mathcal{A}_{\mathsf{Pr}}$ or $\mathcal{A}_{\mathsf{Op}}$), a proposed move (a subset of her private arguments), and the set of arguments $\mathcal{A}_{Com}$, that are common knowledge, to the set $\{permitted, not\_permitted\}$. The standard is *complied with* by a player in the play of a game if each move by the player is permitted by the standard.

A set of standards is *justified* if, for every argumentation game, if for every unpermitted move that achieves the player's aim there is a better (or equal) permitted move that achieves the player's aim. A move m by a player is considered better or equal to another move m' if, for every behaviour of the opposing player, the player can achieve a better or equal outcome of the game by playing m, rather than playing m'. Note that a set of standards might be unjustified even though each standard, individually, is justified. However the combination of the compulsory move and subset-minimality standards is justified.

We say that a strategic argumentation game played under a given finite set of standards has *detectable collusion* if any occurrence of collusion that affects the outcome of the game violates a standard. The set of standards must be finite because an infinite set of standards creates difficulties for compliance verification, both for the players and the auditor. The best that could be done is checks on a random subset of standards. On the face of it, this might be sufficient for the auditor, but if the player has no way to verify her move is compliant with all standards then the auditor cannot reliably infer collusion or incompetence from her failure to comply.

We say that a strategic argumentation game played under a given set of standards is *determined* if all compliant plays of the game lead to the same winner. It appears that collusion is detectable iff the game is determined.

These considerations are similar to the issues in game-theoretic *mechanism design* (see, for example, [57]) where the aim of the design is to achieve some social good, such as fairness, honesty, ..., despite the self-interest of the parties involved. Thus there is a strong focus is on a strategy-proof mechanism, where there is no advantage to players in deviating from socially good behaviour. A classic example of mechanism design is two-person cake-cutting, where the mechanism specifies that one player cuts the cake in two, and the other chooses a piece. This mechanism encourages fairness in the division of the cake.

In an argumentation setting, [116] addresses a version of strategic abstract argumentation (with multiple players) where all players simultaneously play a selection of their arguments, aiming for their focal argument to be accepted. The social good desired is that the arguments accepted after all moves are those that would be accepted if all arguments were available (the omniscient view of the split argumentation framework). That is, roughly, the social good is that arguments are not hidden[17]. Other work, such as [126; 122], also considers hiding of arguments as unfair or dishonest.

This is a different attitude than in strategic argumentation, which treats argument hiding as an inherent feature of adversarial argumentation. [116] characterize when their game is strategy-proof, that is, when there is no advantage to players from hiding arguments. It is only in very restrictive circumstances that honesty is the best policy. Their focus is on the game itself. In particular, the self-interest players have derives from their goals within the game. This is in common with most work on mechanism design. In contrast, the work in this section aims at aligning the self-interest of players with their clients, where that self-interest extends *beyond the game itself.* The introduction of standards is an instance of mechanism design, but we have seen that there is no mechanism that allows strategic play and prevents all collusion. Consequently, computational resistance serves as a back-stop, to discourage collusion.

[17] An argument a is *hidden* if it is not played, even though a player has it available to play. Sometimes, more specifically, it refers to an argument a that defeats an argument b, but is not played when b is played.

	$\mathcal{GR}$	$\mathcal{ST}$	$\mathcal{CO}$	$\mathcal{PR}$	$\mathcal{SST}$	$\mathcal{EA}$	$\mathcal{ID}$
Existential	*Res*	*Res*	*Res*	*Res*	*Res*		*Res*
Universal	*Res*		*Res*				*Res*
Unrejected	*Res*					*Res*	*Res*
Uncontested	*Res*		*Res*	*Res*	*Res*		*Res*
Plurality	*Res*	*Res*	*Res*	*Res*	*Res*		*Res*
Majority	*Res*	*Res*	*Res*	*Res*	*Res*		*Res*
Supermajority	*Res*	*Res*	*Res*	*Res*	*Res*		*Res*

Table 4: Resistance to collusion to ensure Pr wins, for several aims and semantics [90]. *Res* denotes that the combination of aim and semantics is computationally resistant to collusion, while a blank denotes that it is not resistant.

7.4 Resistance to Corruption

Recall that resistance to collusion is based upon the relative computational difficulty of exploiting the corruption, while disguising it, versus the difficulty of playing the game honestly. In other words, we compare the complexity of the Winning Sequence problem with the complexity of normal play as described at the end of subsection 6.3. This comparison is presented in Table 4. While not all combinations of aim and semantics show computational resistance, many do. However, it is notable that three of the aims under the stable semantics do not have resistance to collusion.

This comparison, however, deals only with the initial standard: that a player must play if she has a move. We need to recalculate both the computational cost of normal, honest play and the complexity of the Winning Sequence problem under both standards, in order to determine resistance to collusion when both standards apply. Hence, we need to consider the computational cost of verifying compliance with the subset-minimality standard. The Minimal Move Problem is to verify that a given move is a subset-minimal move.

The Minimal Move Problem for Pr

Instance A split argumentation framework $(\mathcal{A}_{Com}, \mathcal{A}_{\mathsf{Pr}}, \mathcal{A}_{\mathsf{Op}}, \gg)$, an argumentation semantics, an aim for Pr, and a move $M \subseteq \mathcal{A}_{\mathsf{Pr}}$ that achieves the aim for Pr.

Question Is M a minimal set that achieves the aim under the given semantics? That is, is there no subset $N \subset M$ such that Pr's desired outcome is achieved in the argumentation framework $(\mathcal{A}_{Com} \cup N, \gg)$?

It is clear that the complement of this problem can be solved by a non-deterministic algorithm that guesses N and uses an oracle for the Aim Verification problem. Thus the Minimal Move Problem is in coNP^{AV}, where AV is the complexity of the Aim Verification problem. The complexity of the Minimal Move problem for Pr and Op (denoted by MM_{Pr} and MM_{Op}) for selected aims (of Pr) and the grounded and stable semantics is given in Table 5. This is also the work that an auditor must do to verify compliance with the subset-minimality standard. All aims for the grounded semantics lead to the same complexity, so these results have been condensed to a single row.

Honest (i.e. non-corrupt) play under both standards consists of a polynomial number of moves, each involving the search for an effective move, incorporating a verification that the aim is satisfied and the move is minimal. The cost of a single move for Pr is DOM_{Pr}, which is in NP$^{\{AV_{\mathsf{Pr}}, MM_{\mathsf{Pr}}\}}$ and the total cost of honest play is P$^{DOM_{\mathsf{Pr}}}$, and similarly for Op. The total cost of honest play for each player, under the two standards, is shown in Table 5. In some cases the complexity of play has increased as a result of the additional standard, but in other cases it remains the same.

Finally, we must recalculate the cost for collusive play (assuming the players want Pr to win), denoted by WSM. This is the cost of solving the Winning Sequence problem when each player is constrained by the standard to play only subset-minimal moves. The players must search for a sequence of effective minimal moves, and ensure Op has no effective move remaining. Thus WSM is in NP$^{\{AV_{\mathsf{Pr}}, MM_{\mathsf{Pr}}, AV_{\mathsf{Op}}, MM_{\mathsf{Op}}, coDO_{\mathsf{Op}}\}}$. The complexity of WSM is also given in Table 5. In most cases the additional standard does not change the complexity of solving the Winning Sequence problem.

We can see from the table that, once the subset-minimality standard is incorporated, all aims under the stable semantics are resistant to collusion, an improvement (compare with Table 4).

While the additional standard may increase the cost of playing a strategic argumentation game, it is still not comparable to the cost of solving the Winning Strategy problem. Hence all the completist semantics and all the aims remain resistant to espionage.

Of all the semantics that have been investigated, the naive semantics has an interesting property – it is *corruption-proof*, at least for the non-counting aims [89]. Under this semantics the extensions are the maximal conflict-free sets. It is corruption-proof because the outcome is determined by the arguments the players have, if they comply with the compulsory move standard. In this sense, the game is

	MM_{Pr}	MM_{Op}	Hon^{Min}_{Pr}	Hon^{Min}_{Op}	WSM	
Grounded semantics	coNP-c	coNP-c	Δ^p_2-c	Δ^p_2-c	Σ^p_2-c	*Res*
Stable semantics						
Existential	coNP-c	Π^p_2-c	Δ^p_2-c	Δ^p_3-c	Σ^p_3-c	*Res*
Universal	Π^p_2-c	coNP-c	Δ^p_3-c	Δ^p_2-c	Σ^p_3-c	*Res*
Unrejected	Π^p_2-c	coNP-c	Δ^p_3-c	Δ^p_2-c	Σ^p_3-c	*Res*
Uncontested	Π^p_2-c	coNP-c	Δ^p_3-c	Δ^p_2-c	Σ^p_3-c	*Res*
Plurality/Majority	$\mathrm{coNP}^{\mathrm{PP}}$-c	$\mathrm{coNP}^{\mathrm{PP}}$-c	$\mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	*Res*
Supermajority	$\mathrm{coNP}^{\mathrm{PP}}$-c	$\mathrm{coNP}^{\mathrm{PP}}$-c	$\mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{P}^{\mathrm{NP}^{\mathrm{PP}}}$-c	$\mathrm{NP}^{\mathrm{NP}^{\mathrm{PP}}}$-c	*Res*

Table 5: Complexity of Minimality problems and normal play with the minimality standard (for Pr and Op), Winning Sequence problems (for Pr), and resistance to collusion (to ensure Pr wins), under the grounded and stable semantics, for selected aims (of Pr) [92].

strategy-proof. Consequently, if the game has an outcome different from the expected one, we detect corruption/incompetence. But, since every game is determined, this is not a suitable semantics in which to do strategic argumentation.

7.5 Concrete Argumentation Systems

As we saw in Section 5, the SSA, SSSA, and AsSA problems for DL(∂) and DL(δ) are NP-complete, as are the problems for the ASPIC-like language under the grounded semantics. These correspond to the Desired Outcome problem. It was shown in [88] that the Winning Strategy problem is PSPACE-complete and the Winning Sequence problem is Σ^p_2-complete for DL(∂); hence, argumentation in DL(∂) is resistant to corruption. These results relied on careful constructions and proofs reliant on the specific logic.

There are many concrete languages, beyond those discussed in Section 5, that can be used to express arguments. There is a wide variety of defeasible logics [23; 96; 95; 22; 94], languages incorporating inheritance in logic programming [78; 28], other logic programming-based languages [139; 140; 76; 123], languages inspired by argumentation [38; 135], as well as primitive systems like non-monotonic inheritance networks [129]. Unlike systems such as ASPIC [2; 112; 143] and assumption-based argumentation [26], these languages are designed independently from – and sometimes prior to – abstract argumentation. Thus the results of this section do not apply directly to such languages, and following the approach of [88] to establish resistance to corruption would be time-consuming.

However, it was shown in [93] that many of these concrete languages can encode abstract argumentation frameworks under appropriate semantics. Most of the languages employ the grounded semantics, while DefLog [135], ASPDA [140] and a version of NDL [95] employ the stable semantics. Similarly, defeasible logics defined in the framework of [5] for a range of logic programming semantics can encode corresponding abstract argumentation frameworks under the corresponding (in the sense of [31]) completist semantics. As a result, the hardness complexity results for these semantics are carried over to the concrete languages. Consequently, many of these languages are shown to be resistant to corruption. See [91] for details.

8 Related Work

Dialogue games for argumentation describe systems where two opponents argue about the tenability of one or more claims (and thus are in the class of persuasion dialogues [138]). Persuasion dialogues are typically substantive: the participants provide substantive reasons for their claims [81]. As a consequence, the information available during the game evolves, each participant discovering new pieces of information each time the opponent makes new claims.

A structural difference between strategic argumentation and many persuasion dialogues lies within the nature of the reply/counter-argument a player may present: in our setting a participant never asks a *why?* question to a previous opponent's claim. In fact, the answer to the *why?* question is already provided at the very moment a claim is made: every and each claim is justified/supported by the argument proving it (all the rules in the proof of which the claim is the conclusion). Dialogue systems have been classified based on their structural properties, that is whether a player can make a single or multiple moves in one turn, and whether she is allowed to reply only once or multiple times to the other player's moves. In our game, the turn shifts immediately after a player's move, but this is nonetheless a relaxed constraint given that, during such a move, the player may advance a set of arguments, and not just a single one. Moreover, the player is not obliged to respond to the opponent's last move but she may attack any argument proposed so far (possibly her own if this can prove her claim). It is nonetheless true that our framework is a a sort of *unique/move* protocol (a hybrid version): a player can respond only once before the turn passes to the other player even if, as we have shown, such a response is not limited to a single argument.

We do not allow argument retractions (also known as withdrawals): once an argument is played, it will remain as part of the common rules/knowledge base till the end of the game. But it is clear that such a constraint does not prevent a player attacking one of her previously played arguments. We force a replying move to be structurally *relevant*, that is it must be capable of changing the dialogical status of the critical literal/argument (except for the surrendering move which, instead, gives the victory to the adversary). Even allowing retraction in our framework, the computational complexity does not change: a retraction operation would choose a set of rules/arguments to be discarded; thus there is still a choice to be made. However, retraction would change the nature of the game: in the game of Figure 6, Op would not lose. Furthermore, retraction requires restrictions to ensure games terminate.

On the other hand, within our framework a player is not committed to the arguments she plays. Commitments typically require that moves do not contradict or challenge previous commitments/statements; in our framework, players have commitments only towards the claim at dispute as they may, at any time, advance arguments contradicting their own previous statements.

Our turn-taking is in line with the notion of [109; 82] where “when a player is to move, s/he keeps moving until s/he has changed the status of the initial move his or her way”. The sole difference is that we consider the playing of more arguments as a single move, but the essential idea is that even in our framework the player must change the status of the initial claim (the critical literal/argument).

The structure of the arguments defined by our framework is in line with [109]. The idea of an argumentation theory is that of containing all the arguments that are constructible on the basis of a certain theory or knowledge base.

Our framework is *sound* and *fair* according to definitions given in [109]. It is sound because if the proponent wins the game, then the current theory actually proves the critical literal. (Symmetrically, if the opponent wins, the theory either fails to prove the critical literal, disproves it, or proves the opposite, depending on the game variant.) The framework also satisfies fairness given that if, at a given turn, the theory proves the critical literal, then proponent is winning the game. (Again, depending on the type of game, we have that if the theory either fails to prove the critical literal, disproves it, or proves the opposite at a given turn, then the opponent is winning the game.)

The conceptual basis of our formalisation that an argument moved at some earlier stage might be a legal counterargument against some later arguments is not a novelty in the literature of the field, and has been adopted in many frameworks [108; 109].

Our dialogues are coherent (in the sense proposed by [109, Section 7.1]) since we do not allow players to retract their claims. A participant can play a set of arguments conflicting with some of the moves she has put forward in previous steps, if this helps her in taking advantage of information disclosed by the adversary.

[36] describe a rigorous persuasion dialogue game $\mathrm{RPD}_{\mathrm{GD}}$ obtained by adapting the game RPD_0 of [138], replacing propositional logic as the underlying information carrier with abstract argumentation. It has some features in common with strategic argumentation, including private arguments, alternating moves and strategic play. On the other hand, each move is a single locution, which may be a statement, challenge, or question; the only semantics considered is the grounded semantics; and the roles of Pr and Op are quite different from each other, in comparison to strategic argumentation. [36] analyse strategies for their game but it is unclear whether they could be adapted to strategic argumentation.

In game-theoretic terms, a player in a strategic argumentation game has *perfect information* of the structure of the game, the history of the game, and the effects of each move. On the other hand, the players have *incomplete information* of the arguments – and, hence, the possible moves – of adversaries. Most games in the argumentation literature are games of perfect information, while many assume complete information of the adversary, or don't care. For dialogues that are collaborative, seeking to find a joint truth[18], privacy/incomplete information would seem not to matter; for those designed to provide an operational characterization (or proof theory) for specific semantics[19], again it would seem that privacy does not matter. Many works seeking to apply game-theoretic solution concepts, such as Nash equilibria, to argumentation games [120; 115; 97; 50] assume players have complete information about an adversary's possible moves, since that is an underlying assumption of Nash equilibria. On the other hand, many argumentation games in the literature are incomplete information games, for example [121; 107; 116; 125; 27; 69].

One way of analysing argumentation games of incomplete information is to frame them as *Bayesian extensive games with observable actions* [106, chap. 12]: this is possible because every player observes the move of the other player and uncertainty only derives from an initial move of Chance that distributes private information (rules or arguments) among the players. Hence, Chance selects types for the players by assigning to them possibly different theories from the set of all possible theories constructible from a given language. If this hypothesis is correct, notice that Bayesian extensive games with observable actions allow to simply extend the argumentation models proposed, for example, in [120; 69]. Despite this fact, however, complexity results for Bayesian games are far from encouraging (see [61] for games of strategy). Indeed, it seems that considerations similar to those presented by [34] can be applied to argument games: the calculation of the perfect Bayesian equilibrium solution can

[18] Such dialogues are known as *inquiry* dialogues [138].

[19] Examples of such work are [136; 3; 100].

be tremendously complex due to both the size of the strategy space (as a function of the size of the game tree, and it can be computationally hard to compute it [40]), and the dependence between variables representing strategies and players' beliefs.

Many works, for example [119; 70] (and see [127; 24] for more discussion), have addressed the development of a model of the adversary, which can help in developing heuristics for choosing a particular move. Such work does not change the worst-case complexity of making a move, which is NP-hard or worse (see Table 2). Furthermore, even with full knowledge of the adversary, the problem of developing a strategy to beat the adversary is PSPACE-complete (Theorem 7.2).

As mentioned earlier, some work [116; 122] considers hiding arguments (that is, playing an argument a_1 that you know is defeated by a_2, but keeping a_2 private) to be dishonest or even lying. However, in a game of incomplete knowledge a player does not know which arguments hold in the omniscient argumentation framework, so this attitude seems harsh. In any case, our focus is on strategic arguing, where hiding arguments is acceptable. Those works also address "bullshitting" [56] (the introduction of arguments that the player does not know), which is not acceptable in strategic argumentation. We assign to the adjudicator the responsibility for rejecting such arguments. [116] shows that, for their single simultaneous move game, honesty is the best policy only in very restrictive circumstances. [122] identifies some cases in which a player can detect a dishonest adversary, while [107] show that, as the players play more games the probability of a lie being caught by the adversary approaches 1. Apart from these works, which might be considered as addressing corruption isolated to a single player, there seems no discussion of corruption in formal argumentation prior to [88]. [126] address "argumentational integrity", but this refers to fairness in the performance of general argumentation; they do, however, agree that "pretence of truth" is unfair, and would also consider hidden arguments as "insincere contributions".

A majority of the (persuasion) dialogue and argumentation literature takes the perspective of Dung, which sees arguments as monadic elements. There, arguments are typically abstract: the players know such arguments, can propose one (or a set) of them during a turn of the game, but the players do not know their internal structure. Although for many applications this perspective is admissible and gives good benefits in simplifying the problem, in some cases it results in an oversimplification. Anyway, restricting to abstract argumentation does not reduce the complexity of the problems, in general. We have seen in Section 7.5 that hardness results at the abstract level can be extended to the concrete level. Thus, it seems that the complexity of the problems largely comes from the problems themselves (including semantics and strategic aims) and not from the level of detail of the arguments.

Strategic argumentation can be considered a specific form of collective argumentation [25] (and judgement aggregation), where different argumentation frameworks contribute to a combined judgement on the arguments. This topic is usually considered in the context of collaboration, but some work considers self-interested agents [27; 77]. Strategic argumentation is clearly a framework-wise approach, in the classification of [25], where argument frameworks are combined, and arguments then evaluated in the result. See Chapter 4 [18] of this handbook for additional discussion of this topic from a computational social choice perspective.

An approach to argumentation of interest for strategic argumentation is *probabilistic argumentation*. We refer the readers to Chapter 7 of this volume [73] for an in-depth discussion of this topic. Under the constellations approach to probabilistic argumentation, the key idea is that the existence (or, perhaps, validity) of arguments and attacks is unknown, but there is a probability distribution function describing the likelihood of different possibilities. Such an approach could be a useful refinement for strategic argumentation, allowing the replacement of a complete unknown (the adversary's arguments) with a more detailed model of the adversary. This might provide the basis for a player to choose among different moves.

Within the framework proposed in [80], probabilities are used to represent the likelihood that arguments and attacks exist. This defines a probability distribution over all possible worlds, where each possible world is an abstract argumentation framework consisting of some subset of the arguments and attacks. Extensions arise, as usual, for a possible world, by applying any of various semantics. In [80; 52], the authors tackle the probabilistic counterpart of the problem $\mathrm{VER}^{\sigma}(S)$, that is, the problem $\mathrm{PROB}^{\sigma}_{\mathcal{F}}(S)$ of computing the probability $Prs^{\sigma}_{\mathcal{F}}(S)$ that a set S of arguments is an extension according to a given semantics σ, given a probabilistic argumentation framework $\mathcal{F}$. [80] suggested that computing the exact value of probability $Prs^{\sigma}_{\mathcal{F}}(S)$ requires exponential time, and employed a Monte-Carlo simulation approach to approximate $\mathrm{PROB}^{\sigma}_{\mathcal{F}}(S)$. However, as far as the admissible and stable semantics are concerned, [52]'s results show that the exact value of $Prs^{\sigma}_{\mathcal{F}}(S)$ can be determined in polynomial time, without enumerating the possible worlds. Nevertheless, in general the number of extensions is potentially exponential and, for other semantics, the problem is intractable. Consequently, it seems likely that many of the problems arising in strategic probabilistic argumentation will also be difficult.

Finally, there are some works that might appear to be addressing strategic argumentation, but have only weak relevance to the topic. *Strategic manoeuvring* was introduced in [133] to bridge the gap between dialectical and rhetorical approaches to the study of argumentation [134]. It refers to "the efforts arguers make in argumentative discourse to reconcile aiming for rhetorical effectiveness with maintaining dialectical standards of reasonableness" [134]. It was introduced in the context of the pragma-dialectical theory of argumentation [132;

131], which focuses on analysis and evaluation of lingual argumentation. This theory is a much broader view of argumentation than we address here. Nevertheless, there might be links between strategic manoeuvring and strategic argumentation applied to value-based or audience-based argumentation frameworks [19].

We have already mentioned [126], which addresses ethics of lingual argumentative communication. It proposes standards for lingual argumentation, under the title *argumentational integrity*, and develops a taxonomy of these standards. The standards address rhetoric rather than the relation between arguments, and the notion of integrity does not include corruption (except to the extent already discussed in Section 7.1).

Despite the title, [46] analyzes a very different scenario than we do here. In that work, a decision-maker consults an expert, who possibly has an ulterior motive, about deciding between two alternatives. For example, a customer consulting a camera salesman about which camera to buy. The expert has all the arguments (which are informal) for both alternatives, and the decision-maker has none. The game is modelled probabilistically, and the paper performs an equilibrium analysis. Apart from the words "strategic argumentation" and the possibility of a self-interested player, there is no relationship between this work and the work on strategic argumentation presented here.

9 Future Directions

There are multiple avenues for further research in this area.

- The NP-completeness results in Section 5.1 apply to a wide variety of logics whose inference problem can be solved in polynomial time. Other logics, such as those in [21], that have a harder inference problem might result in complexities higher in the polynomial hierarchy. An analysis of such cases could extend the existing results.

- Structured argumentation theories can generate a large number of arguments, possibly infinitely many. This prevents applying the results of Section 7 to structured argumentation directly. For example, we used a different method to prove Theorem 5.8. What is needed is to find a polynomially-sized argumentation framework that is equivalent to the generated argumentation framework for the semantics of interest.

- In this chapter we have focused on a competitive situation, where the two players' aims are inconsistent. However, the basics of strategic argumentation also apply when the player's aims are consistent. In this case, strategic argumentation represents a crude adversarial negotiation. It is worth exploring how concepts from strategic argumentation can be used to analyse such negotiations, both in strategic argumentation games and in other negotiation games.

- Work has focused on two-player games of strategic argumentation. However, there are often more than two stakeholders in an adjudication, and so it would be interesting to see how strategic argumentation can be extended to more players. Among the many issues that would need to be addressed are: the protocol for turn-taking, the criterion for terminating the game, and the possibility of some players cooperating to construct an argument that none of them could construct individually. There is discussion of multi-party dialogues in [37; 99; 127]. In general, game play would appear to be more complex because of the potential for shifting alliances between players, and because players might not be compelled to make a move at each opportunity. Corruption might also be more complicated.

- In current work, the players' aims are implicitly assumed to be known and fixed. In some scenarios this might be realistic. However, there are scenarios where the motivations of a player are unclear, and/or may change over the course of argumentation. For example, a defence lawyer might begin with a "not guilty" aim but, if the trial is going badly, change tack to instead aim at a mis-trial. Thus, the extension of strategic argumentation to consider aims as possibly private and flexible/changeable is an interesting one.

- In the treatment of strategic abstract argumentation, the most prominent semantics for Dung's framework have been addressed, but there remain many semantics in the literature for which resistance to corruption is unknown. In addition, the treatment of the subset-minimality standard remains to be done for most semantics.

- The treatment of espionage assumes that full knowledge of an adversary's arguments is obtained. Perhaps the illicit gain of only some knowledge is more realistic. How can this framework be extended to cases where only partial knowledge is obtained? The work of [39] could be a first step in this direction. That paper represents partial knowledge and determines whether a player has the ability to force a desired outcome. However, it will need much expansion, as it only addresses Existential and Universal outcomes, and only for the stable semantics; assumes that the player's control arguments cannot be attacked by partially-known attacks; and does not consider multiple moves.

- Although standards are insufficient to make corruption visible, they can also be useful in guiding heuristic approaches to playing strategic argumentation games. For example, the subset-minimality standard prevents a player needlessly creating an opening for the adversary. ([105] employ this as a heuristic in a different dialogue game than the one we have presented.) Thus, it would be helpful to identify more standards, especially those that can be incorporated in heuristics or used to improve a heuristic move.

- The brief discussion of argument retraction in Section 8 deserves expansion. Strategic argumentation with retraction would seem to produce an outcome that is less arbitrary than without retraction, but perhaps the strategic element would be much diminished. Argument retraction would need to be restricted in some way, or an explicit termination rule introduced, otherwise a losing player might be able to prevent termination by repeatedly retracting arguments and then replaying them. Treating such retraction as a disavowal of some or all of the backtracked arguments (i.e. a commitment not to use those arguments in the remainder of the game) might temper the power of retraction and lead to a richer game.

- The notion of resistance to corruption we discussed is based on worst-case complexity, but this is sometimes not reflective of the difficulty of problems that arise in practice. An empirical comparison of the difficulty of solving the problems in practice and a study of approximation algorithms for these problems are needed.

- As observed in subsection 6.1, it can be worthwhile to consider an adjudicator as part of a strategic argumentation game. In this case we might consider whether the adjudicator can be subject to corruption. If the role of the adjudicator is simply to enforce the consequences arrived at by the players then there is nothing *in the game* that allows us to detect corruption.

  However, if we assume that the adjudicator chooses the semantics under which the game will be adjudicated, we have an action by the adjudicator that can be subject to analysis. This leads to quite different games, especially if the adjudicator changes the semantics *during* the playing of the game. While this appears to be rather Kafkaesque, it might be somewhat reflective of some situations where the judiciary can be influenced by other arms of government. The adjudicator then has both the choice of semantics to impose, and the choice of timing of this move. More realistically, [110] presents a game where the adjudicator plays an active role, based on a detailed model of legal procedure.

Perhaps that model is a base on which corruption of adjudication can be investigated.

10 Conclusions

Strategic argumentation is a primarily adversarial approach to dialogue games with incomplete information. It reflects aspects of legal argument. The idea can be applied at a concrete level, as we have demonstrated using defeasible logic rules as the basis for arguments, and at an abstract level, which was demonstrated using Dung's argumentation system.

The key element of strategic argumentation games is each player re-establishing their aim at the end of their turn. The details of the argument framework are not needed at this level of abstraction, only that they can be used to define a notion of acceptance/aim achievement. Consequently, we have a formulation of strategic argumentation that applies to Dung's notion of argumentation framework [41], but also to bipolar argumentation frameworks [33], abstract argumentation frameworks with sets of attacking arguments [101; 55][20], and preference-based argumentation frameworks [3; 17]. If, in the dialogue game $(\mathcal{A}, \mathcal{R})$, we extend $\mathcal{R}$ beyond simply relations on $\mathcal{A}$ then we can have strategic argumentation on constrained argumentation frameworks [35], weighted argument systems [42], abstract dialectical frameworks [30], and probabilistic argumentation frameworks [80], and the ideas might well be applicable to other forms of argumentation framework. Similarly, the ideas of strategic argumentation apply to semantics other than Dung-style semantics.

We have also demonstrated how the strategic argumentation framework can be used to address issues of corruption, even when the corrupt behaviour is motivated by rewards extrinsic to the game. We have not much addressed the strategies that a player might employ when playing a strategic argumentation game, although the study of standards in Section 7 provides some guidelines. More information on that topic can be found in Section 5.2 of Chapter 9 [24] in this handbook.

Acknowledgments

We thank the reviewers for their comments, which helped to improve this chapter. Michael Maher has an adjunct position at Griffith University and an honorary position at UNSW.

[20] We have already addressed argumentation with sets of attacking arguments in a non-abstract setting, in Section 5.

References

[1] Leila Amgoud. A replication study of semantics in argumentation. In Sarit Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019*, pages 6260–6266. ijcai.org, 2019.

[2] Leila Amgoud, Lianne Bodenstaff, Martin Caminada, Peter McBurney, Simon Parsons, Henry Prakken, Jelle van Veenen, and Gerard Vreeswijk. Final review and report on formal argumentation system. Technical report, 2006.

[3] Leila Amgoud and Claudette Cayrol. A reasoning model based on the production of acceptable arguments. *Ann. Math. Artif. Intell.*, 34(1-3):197–215, 2002.

[4] Grigoris Antoniou. A Discussion of Some Intuitions of Defeasible Reasoning. In George Vouros and Themistoklis Panayiotopoulos, editors, *Methods and Applications of Artificial Intelligence*, volume 3025 of *Lecture Notes in Computer Science*, pages 311–320. Springer Berlin / Heidelberg, 2004.

[5] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. A flexible framework for defeasible logics. In *AAAI/IAAI*, pages 405–410, 2000.

[6] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. Representation results for defeasible logic. *ACM Trans. Comput. Log.*, 2(2):255–287, 2001.

[7] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. Representation results for defeasible logic. *ACM Transactions on Computational Logic*, 2(2):255–286, 2001.

[8] Grigoris Antoniou, David Billington, Guido Governatori, Michael J. Maher, and Andrew Rock. A family of defeasible reasoning logics and its implementation. In *Proc. ECAI-2000*, pages 459–463, 2000.

[9] Grigoris Antoniou, David Billington, and Michael J. Maher. On the analysis of regulations using defeasible rules. In *32nd Annual Hawaii International Conference on System Sciences (HICSS-32)*, 1999.

[10] Grigoris Antoniou, Thomas Skylogiannis, Antonis Bikakis, Martin Doerr, and Nick Bassiliades. Dr-brokering: A semantic brokering system. *Knowledge-Based Systems*, 20(1):61–72, 2007.

[11] Joseph M. Barbato. Scotland's bastard verdict: Intermediacy and the unique three-verdict system. *Indiana International & Comparative Law Review*, 15:543–582, 2005.

[12] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *Knowledge Eng. Review*, 26(4):365–410, 2011.

[13] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In Pietro Baroni; Dov Gabbay; Massimiliano Giacomin; Leendert van der Torre, editor, *Handbook on Formal Argumentation*, volume 1, pages 688–767. College Publications, February 2018.

[14] Pietro Baroni, Paul E. Dunne, and Massimiliano Giacomin. On extension counting problems in argumentation frameworks. In Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Guillermo Ricardo Simari, editors, *Computational Models*

of Argument: Proceedings of COMMA 2010, volume 216 of *Frontiers in Artificial Intelligence and Applications*, pages 63–74. IOS Press, 2010.

[15] Pietro Baroni, Antonio Rago, and Francesca Toni. How many properties do we need for gradual argumentation? In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1736–1743, 2018.

[16] John J. Bartholdi, Craig A. Tovey, and Michael A. Trick. The computational difficulty of manipulating an election. *Social Choice and Welfare*, 6(3):227–241, 1989.

[17] Dorothea Baumeister, Daniel Neugebauer, and Jörg Rothe. Verification in attack-incomplete argumentation frameworks. In Toby Walsh, editor, *Algorithmic Decision Theory*, pages 341–358. Springer International Publishing, 2015.

[18] Dorothea Baumeister, Daniel Neugebauer, and Jörg Rothe. Collective acceptability in abstract argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 4. College Publications, 2021.

[19] Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.

[20] Salem Benferhat, Didier Dubois, and Henri Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In David Heckerman and E. H. Mamdani, editors, *UAI '93: Proceedings of the Ninth Annual Conference on Uncertainty in Artificial Intelligence*, pages 411–419. Morgan Kaufmann, 1993.

[21] David Billington. A defeasible logic for clauses. In *AI 2011: Advances in Artificial Intelligence - 24th Australasian Joint Conference, Proceedings*, pages 472–480, 2011.

[22] David Billington. *Factual and Plausible Reasoning*, volume 81 of *Studies in Logic*. College Publications, 2019.

[23] David Billington, Grigoris Antoniou, Guido Governatori, and Michael J. Maher. An inclusion theorem for defeasible logics. *ACM Trans. Comput. Log.*, 12(1):6, 2010.

[24] Elizabeth Black, Nicolas Maudet, and Simon Parsons. Argumentation-based dialogue. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 9. College Publications, 2021.

[25] Gustavo Adrian Bodanza, Fernando Tohmé, and Marcelo Auday. Collective argumentation: A survey of aggregation issues around argumentation frameworks. *Argument & Computation*, 8(1):1–34, 2017.

[26] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.

[27] Elise Bonzon and Nicolas Maudet. On the outcomes of multiparty persuasion. In *Argumentation in Multi-Agent Systems - 8th International Workshop, ArgMAS 2011*,

volume 7543 of *Lecture Notes in Computer Science*, pages 86–101. Springer, 2012.

[28] Annalisa Bossi, Michele Bugliesi, Maurizio Gabbrielli, Giorgio Levi, and Maria Chiara Meo. Differential logic programming. In *Conference Record of the Twentieth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 359–370, 1993.

[29] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. *Handbook of Computational Social Choice.* Cambridge University Press, 2016.

[30] Gerhard Brewka and Stefan Woltran. Abstract dialectical frameworks. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010*, 2010.

[31] Martin Caminada, Samy Sá, João Alcântara, and Wolfgang Dvořák. On the equivalence between logic programming semantics and argumentation semantics. *Int. J. Approx. Reasoning*, 58:87–111, 2015.

[32] Claudette Cayrol, Sylvie Doutre, Marie-Christine Lagasquie-Schiex, and Jérôme Mengin. “Minimal defence”: a refinement of the preferred semantics for argumentation frameworks. In *9th International Workshop on Non-Monotonic Reasoning (NMR 2002), Proceedings*, pages 408–415, 2002.

[33] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 8th European Conference, ECSQARU 2005, Proceedings*, pages 378–389, 2005.

[34] Georgios Chalkiadakis and Craig Boutilier. Coalitional bargaining with agent type uncertainty. In Manuela M. Veloso, editor, *IJCAI*, pages 1227–1232, 2007.

[35] Sylvie Coste-Marquis, Caroline Devred, and Pierre Marquis. Constrained argumentation frameworks. In *Proceedings, Tenth International Conference on Principles of Knowledge Representation and Reasoning*, pages 112–122, 2006.

[36] Joseph Devereux and Chris Reed. Strategic argumentation in rigorous persuasion dialogue. In *Argumentation in Multi-Agent Systems, 6th International Workshop, ArgMAS 2009*, pages 94–113, 2009.

[37] Frank Dignum and Gerard Vreeswijk. Towards a testbed for multi-party dialogues. In Frank Dignum, editor, *Advances in Agent Communication, International Workshop on Agent Communication Languages, ACL 2003, Melbourne, Australia, July 14, 2003*, volume 2922 of *Lecture Notes in Computer Science*, pages 212–230. Springer, 2003.

[38] Yannis Dimopoulos and Antonis C. Kakas. Logic programming without negation as failure. In *Proceedings of the 1995 International Symposium on Logic Programming*, pages 369–384, 1995.

[39] Yannis Dimopoulos, Jean-Guy Mailly, and Pavlos Moraitis. Control argumentation frameworks. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18)*, pages 4678–4685. AAAI Press, 2018.

[40] Yannis Dimopoulos, Bernhard Nebel, and Francesca Toni. On the computational

complexity of assumption-based argumentation for default reasoning. *Artif. Intell.*, 141(1/2):57–78, 2002.

[41] P.M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

[42] Paul E. Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael J. Wooldridge. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artif. Intell.*, 175(2):457–486, 2011.

[43] Paul E. Dunne and Michael Wooldridge. Complexity of abstract argumentation. In I. Rahwan and G.R. Simari, editors, *Argumentation in Artificial Intelligence*, pages 85–104. Springer, 2009.

[44] Wolfgang Dvořák. On the complexity of computing the justification status of an argument. In Sanjay Modgil, Nir Oren, and Francesca Toni, editors, *Theorie and Applications of Formal Argumentation - First International Workshop, TAFA*, volume 7132 of *Lecture Notes in Computer Science*, pages 32–49. Springer, 2011.

[45] Wolfgang Dvořák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. *IfCoLog Journal of Logics and their Applications*, 4(8), 2017.

[46] Wioletta Dziuda. Strategic argumentation. *J. Econ. Theory*, 146(4):1362–1397, 2011.

[47] Kathleen M. Eisenhardt. Agency theory: An assessment and review. *The Academy of Management Review*, 14(1):57–74, 1989.

[48] Jenny Eriksson Lundström, Guido Governatori, Subhasis Thakur, and Vineet Padmanabhan. An asymmetric protocol for argumentation games in defeasible logic. In Aditya Ghose and Guido Governatori, editors, *10 Pacific Rim International Workshop on Multi-Agents*, volume 5044 of *LNAI*, pages 219–231, Heidelberg, 2008. Springer.

[49] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Common Knowledge Revisited.* MIT Press, 2003.

[50] Xiuyi Fan and Francesca Toni. On the interplay between games, argumentation and dialogues. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, Singapore, May 9-13, 2016*, pages 260–268, 2016.

[51] Arthur M. Farley and Kathleen Freeman. Burden of proof in legal argumentation. In *Proceedings of the Fifth International Conference on Artificial Intelligence and Law, ICAIL '95*, pages 156–164, 1995.

[52] Bettina Fazzinga, Sergio Flesca, and Francesco Parisi. On the complexity of probabilistic abstract argumentation. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, 2013.

[53] Johannes Klaus Fichte, Markus Hecher, and Arne Meier. Counting complexity for reasoning in abstract argumentation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI*, pages 2827–2834. AAAI Press, 2019.

[54] Johannes C. Flieger. Relevance and minimality in systems of defeasible argumentation. Technical report, Imperial College, London, 2002.

[55] Giorgos Flouris and Antonis Bikakis. A comprehensive study of argumentation frame-

works with sets of attacking arguments. *Int. J. Approx. Reason.*, 109:55–86, 2019.

[56] Harry G. Frankfurt. *On Bullshit.* Princeton University Press, 2005.

[57] Dinesh Garg, Yadati Narahari, and Sujit Gujar. Foundations of mechanism design: A tutorial. *Sadhana*, 33(2), 2008.

[58] Thomas F. Gordon. The pleadings game: An exercise in computational dialectics. *Artif. Intell. Law*, 2:239–292, December 1994.

[59] Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artif. Intell.*, 171(10-15):875–896, 2007.

[60] Thomas F. Gordon and Douglas Walton. Proof burdens and standards. In I. Rahwan and G.R. Simari, editors, *Argumentation in Artificial Intelligence*, pages 239–260. Springer, 2009.

[61] Georg Gottlob, Gianluigi Greco, and Toni Mancini. Complexity of pure equilibria in bayesian games. In Manuela M. Veloso, editor, *IJCAI*, pages 1294–1299, 2007.

[62] Guido Governatori. On the relationship between Carneades and Defeasible Logic. In *The 13th International Conference on Artificial Intelligence and Law, Proceedings of the Conference*, pages 31–40, 2011.

[63] Guido Governatori, Marlon Dumas, Arthur H.M. ter Hofstede, and Phillipa Oaks. A formal approach to protocols and strategies for (legal) negotiation. In Henry Prakken, editor, *Procedings of the 8th International Conference on Artificial Intelligence and Law*, pages 168–177. IAAIL, ACM Press, 2001.

[64] Guido Governatori and Michael J. Maher. Annotated defeasible logic. *Theory Pract. Log. Program.*, 17(5-6):819–836, 2017.

[65] Guido Governatori, Michael J. Maher, Grigoris Antoniou, and David Billington. Argumentation semantics for defeasible logics. *Journal of Logic and Computation*, 14(5):675–702, 2004.

[66] Guido Governatori, Francesco Olivieri, Antonino Rotolo, and Simone Scannapieco. Computing strong and weak permissions in defeasible logic. *Journal of Philosophical Logic*, 42(6):799–829, 2013.

[67] Guido Governatori, Francesco Olivieri, Simone Scannapieco, Antonino Rotolo, and Matteo Cristani. Strategic argumentation is NP-complete. In *Proc. ECAI 2014*. IOS Press, 2014.

[68] Guido Governatori and Antonino Rotolo. Changing legal systems: legal abrogations and annulments in defeasible logic. *Logic Journal of IGPL*, 18(1):157–194, 2010.

[69] Davide Grossi and Wiebe van der Hoek. Audience-based uncertainty in abstract argument games. In *IJCAI'13*, pages 143–149. AAAI Press, 2013.

[70] Christos Hadjinikolis, Yiannis Siantos, Sanjay Modgil, Elizabeth Black, and Peter McBurney. Opponent modelling in persuasion dialogues. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2013.

[71] Emmanuel Hadoux, Aurélie Beynier, Nicolas Maudet, Paul Weng, and Anthony Hunter. Optimization of probabilistic argumentation with markov decision models. In Qiang Yang and Michael J. Wooldridge, editors, *Proceedings of the Twenty-Fourth Interna-*

tional Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015, pages 2004–2010. AAAI Press, 2015.

[72] Mustafa Hashmi, Guido Governatori, Ho-Pun Lam, and Moe Thandar Wynn. Are we done with business process compliance: state of the art and challenges ahead. *Knowl. Inf. Syst.*, 57(1):79–133, 2018.

[73] Anthony Hunter, Sylwia Polberg, Nico Potyka, Tjitze Rienstra, and Matthias Thimm. Probabilistic argumentation: A survey. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 7. College Publications, 2021.

[74] Mohammad Badiul Islam and Guido Governatori. RuleRS: a rule-based architecture for decision support systems. *Artif. Intell. Law*, 26(4):315–344, 2018.

[75] David S. Johnson. A catalog of complexity classes. In *Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity*, pages 67–161. Elsevier, 1990.

[76] Antonis C. Kakas, Pavlos Moraitis, and Nikolaos I. Spanoudakis. *GORGIAS*: Applying argumentation. *Argument & Computation*, 10(1):55–81, 2019.

[77] Dionysios Kontarinis, Elise Bonzon, Nicolas Maudet, and Pavlos Moraitis. On the use of target sets for move selection in multi-agent debates. In *ECAI 2014 - 21st European Conference on Artificial Intelligence*, pages 1047–1048, 2014.

[78] Els Laenens and Dirk Vermeir. A fixpoint semantics for ordered logic. *J. Log. Comput.*, 1(2):159–185, 1990.

[79] Ho-Pun Lam, Guido Governatori, and Régis Riveret. On aspic^{+} and defeasible logic. In *Computational Models of Argument - Proceedings of COMMA 2016*, pages 359–370, 2016.

[80] Hengfei Li, Nir Oren, and Timothy J. Norman. Probabilistic argumentation frameworks. In Sanjay Modgil, Nir Oren, and Francesca Toni, editors, *Theorie and Applications of Formal Argumentation - First International Workshop, TAFA 2011. Revised Selected Papers*, volume 7132 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2012.

[81] Paul Lorenzen and Kuno Lorenz. *Dialogische Logik.* Darmstadt, 1978.

[82] Ronald Prescott Loui. Process and policy: Resource-bounded nondemonstrative reasoning. *Computational Intelligence*, 14(1):1–38, 1998.

[83] Michael J. Maher. Propositional defeasible logic has linear complexity. *Theory and Practice of Logic Programming*, 1(6):691–711, 2001.

[84] Michael J. Maher. Relative expressiveness of defeasible logics. *Theory and Practice of Logic Programming*, 12(4-5):793–810, 2012.

[85] Michael J. Maher. Relative expressiveness of defeasible logics II. *Theory and Practice of Logic Programming*, 13:579–592, 2013.

[86] Michael J. Maher. Relative Expressiveness of Well-Founded Defeasible Logics. In Stephen Cranefield and Abhaya Nayak, editors, *AI 2013: Advances in Artificial Intelligence*, volume 8272 of *Lecture Notes in Computer Science*, pages 338–349. Springer International Publishing, 2013.

[87] Michael J. Maher. Comparing defeasible logics. In Torsten Schaub, Gerhard Friedrich,

and Barry O'Sullivan, editors, *ECAI 2014 - 21st European Conference on Artificial Intelligence*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 585–590. IOS Press, 2014.

[88] Michael J. Maher. Complexity of exploiting privacy violations in strategic argumentation. In Duc Nghia Pham and Seong-Bae Park, editors, *PRICAI 2014: Trends in Artificial Intelligence - 13th Pacific Rim International Conference on Artificial Intelligence*, volume 8862 of *Lecture Notes in Computer Science*, pages 523–535. Springer, 2014.

[89] Michael J. Maher. Corrupt strategic argumentation: The ideal and the naive. In *AI 2016: Advances in Artificial Intelligence*, pages 17–28, 2016.

[90] Michael J. Maher. Resistance to corruption of general strategic argumentation. In *Proc. Int. Conf. Principles and Practice of Multi-Agent Systems*, pages 61–75, 2016.

[91] Michael J. Maher. Resistance to corruption of strategic argumentation. In *AAAI Conference on Artificial Intelligence*, pages 1030–1036, 2016.

[92] Michael J. Maher. Corruption and audit in strategic argumentation. Technical report, Reasoning Research Institute, 2017.

[93] Michael J. Maher. Relating concrete defeasible reasoning formalisms and abstract argumentation. *Fundam. Inform.*, 155(3):233–260, 2017.

[94] Michael J. Maher, Ilias Tachmazidis, Grigoris Antoniou, Stephen Wade, and Long Cheng. Rethinking defeasible reasoning: A scalable approach. *Theory and Practice of Logic Programming*, 20(4):552–586, 2020.

[95] Frederick Maier. Interdefinability of defeasible logic and logic programming under the well-founded semantics. *Theory and Practice of Logic Programming*, 13:107–142, 2013.

[96] Frederick Maier and Donald Nute. Well-founded semantics for defeasible logic. *Synthese*, 176(2):243–274, 2010.

[97] P. Matt and F. Toni. A game-theoretic measure of argument strength for abstract argumentation. In *JELIA 2008*, volume 5293 of *LNCS*, pages 285–297. Springer, 2008.

[98] Mohamed Mbarki, Jamal Bentahar, and Bernard Moulin. Specification and complexity of strategic-based reasoning using argumentation. In *Argumentation in Multi-Agent Systems, Third International Workshop, ArgMAS 2006, Hakodate, Japan, May 8, 2006, Revised Selected and Invited Papers*, pages 142–160, 2006.

[99] Peter McBurney and Simon Parsons. Dialogue games for agent argumentation. In Guillermo Ricardo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 261–280. Springer, 2009.

[100] Sanjay Modgil and Martin Caminada. Proof theories and algorithms for abstract argumentation frameworks. In *Argumentation in Artificial Intelligence*, pages 105–129. Springer, 2009.

[101] Søren Holbech Nielsen and Simon Parsons. A generalization of Dung's abstract framework for argumentation: Arguing with sets of attacking arguments. In *Argumentation in Multi-Agent Systems, Third International Workshop, ArgMAS 2006*, pages 54–73, 2006.

[102] Donald Nute. Defeasible reasoning: A philosophical analysis in Prolog. In James H. Fetzer, editor, *Aspects of Artificial Intelligence*, volume 1 of *Studies in Cognitive Systems*. Springer, 1988.

[103] Donald Nute. Defeasible logic. In Dov M. Gabbay, Chris J. Hogger, and J. Allen Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3. Oxford University Press, 1994.

[104] Donald Nute. Defeasible logic: Theory, Implementation and Applications. In *Proceedings of the 14th International Conference on Applications of Prolog (INAP 2001)*, pages 151–169. Springer, Berlin, 2001.

[105] Nir Oren, Timothy J. Norman, and Alun D. Preece. Loose lips sink ships: a heuristic for argumentation. In *Proceedings of the Third International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2006)*, pages 121–134, 2006.

[106] M. J. Osborne and A. Rubinstein. *A Course in Game Theory.* MIT Press, 1999.

[107] Simon Parsons and Elizabeth Sklar. How agents alter their beliefs after an argumentation-based dialogue. In *Argumentation in Multi-Agent Systems, Second International Workshop, ArgMAS 2005*, pages 297–312, 2005.

[108] Henry Prakken. Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese*, 127:2001, 2000.

[109] Henry Prakken. Coherence and flexibility in dialogue games for argumentation. *J. Log. Comput.*, 15(6):1009–1040, 2005.

[110] Henry Prakken. A formal model of adjudication dialogues. *Artif. Intell. Law*, 16(3):305–328, 2008.

[111] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1:93–124, 2010.

[112] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.

[113] Henry Prakken and Giovanni Sartor. Modelling reasoning with precedents in a formal dialogue game. *Artif. Intell. Law*, 6(2-4):231–287, 1998.

[114] Henry Prakken and Giovanni Sartor. Law and logic: A review from an argumentation perspective. *Artif. Intell.*, 227:214–245, 2015.

[115] Ariel D. Procaccia and Jeffrey S. Rosenschein. Extensive-form argumentation games. In Marie-Pierre Gleizes, Gal A. Kaminka, Ann Nowé, Sascha Ossowski, Karl Tuyls, and Katja Verbeeck, editors, *EUMAS 2005 - Proceedings of the Third European Workshop on Multi-Agent Systems*, pages 312–322. Koninklijke Vlaamse Academie van Belie voor Wetenschappen en Kunsten, 2005.

[116] Iyad Rahwan, Kate Larson, and Fernando A. Tohmé. A characterisation of strategy-proofness for grounded argumentation semantics. In *IJCAI*, pages 251–256, 2009.

[117] Iyad Rahwan, Sarvapali D. Ramchurn, Nicholas R. Jennings, Peter Mcburney, Simon Parsons, and Liz Sonenberg. Argumentation-based negotiation. *The Knowledge Engineering Review*, 18(4):343–375, 2003.

[118] Daniel M. Reeves, Michael P. Wellman, and Benjamin N. Grosof. Automated negotiation

from declarative contract descriptions. *Computational Intelligence*, 18(4):482–500, 2002.

[119] Tjitze Rienstra, Matthias Thimm, and Nir Oren. Opponent models with uncertainty for strategic argumentation. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2013.

[120] Régis Riveret, Henry Prakken, Antonino Rotolo, and Giovanni Sartor. Heuristics in argumentation: A game theory investigation. In *COMMA 2008*, pages 324–335. IOS Press, 2008.

[121] Bram Roth, Régis Riveret, Antonino Rotolo, and Guido Governatori. Strategic argumentation: a game theoretical investigation. In *The Eleventh International Conference on Artificial Intelligence and Law, Proceedings of the Conference*, pages 81–90, 2007.

[122] Chiaki Sakama. Dishonest arguments in debate games. In *Computational Models of Argument - Proceedings of COMMA 2012, Vienna, Austria, September 10-12, 2012*, pages 177–184, 2012.

[123] Chiaki Sakama. Debate games in logic programming. In *Declarative Programming and Knowledge Management - Declarative Programming Days, KDPD 2013*, pages 185–201, 2013.

[124] K. Satoh and K. Takahashi. A semantics of argumentation under incomplete information. In *Proceedings of Jurisn 2011*, 2011.

[125] Ken Satoh and Kazuko Takahashi. A semantics of argumentation under incomplete information. In *JURISIN*, pages 86–97, 2011.

[126] Margrit Schreier, Norbert Groeben, and Ursula Christmann. "That's Not Fair!" argumentational integrity as an ethics of argumentative communication. *Argumentation*, 9:267–289, 1995.

[127] Matthias Thimm. Strategic argumentation in multi-agent systems. *Künstliche Intell.*, 28(3):159–168, 2014.

[128] Matthias Thimm and Alejandro Javier García. On strategic argument selection in structured argumentation systems. In *Argumentation in Multi-Agent Systems - 7th International Workshop, ArgMAS 2010*, pages 286–305, 2010.

[129] David S. Touretzky. *The Mathematics of Inheritance Systems.* Morgan Kaufmann, 1986.

[130] David S. Touretzky, John F. Horty, and Richmond H. Thomason. A Clash of Intuitions: The Current State of Nonmonotonic Multiple Inheritance Systems. In *Proceedings of the 10th international joint conference on Artificial intelligence (IJCAI'87)*, pages 476–482, San Francisco, CA, USA, 1987. Morgan Kaufmann Publishers Inc.

[131] Frans H. van Eemeren, editor. *Reasonableness and Effectiveness in Argumentative Discourse, Fifty Contributions to the Development of Pragma-Dialectics*, volume 27 of *Argumentation Library*. Springer, 2015.

[132] Frans H. van Eemeren and R. Grootendorst, editors. *A Systematic Theory of Argumentation. The Pragma-Dialectical Approach.* Cambridge University Press, 2004.

[133] Frans H. van Eemeren and Peter Houtlosser. Strategic maneuvering: Maintaining a

delicate balance. In Frans H. van Eemeren and Peter Houtlosser, editors, *Dialectic and Rhetoric: The Warp and Woof of Argumentation Analysis*. Kluwer Academic Publishers, 2002.

[134] Frans H. van Eemeren and Peter Houtlosser. Strategic maneuvering: A synthetic recapitulation. *Argumentation*, 20:381–392, 2006.

[135] Bart Verheij. DefLog: on the logical interpretation of prima facie justified assumptions. *J. Log. Comput.*, 13(3):319–346, 2003.

[136] Gerard Vreeswijk and Henry Prakken. Credulous and sceptical argument games for preferred semantics. In *Logics in Artificial Intelligence, European Workshop, JELIA 2000*, volume 1919 of *Lecture Notes in Computer Science*, pages 239–253. Springer, 2000.

[137] Klaus W. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Inf.*, 23(3):325–356, 1986.

[138] Douglas Walton and Erik C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, 1995.

[139] Hui Wan, Benjamin N. Grosof, Michael Kifer, Paul Fodor, and Senlin Liang. Logic programming with defaults and argumentation theories. In *ICLP*, pages 432–448, 2009.

[140] Hui Wan, Michael Kifer, and Benjamin N. Grosof. Defeasibility in answer set programs with defaults and argumentation rules. *Semantic Web*, 6(1):81–98, 2015.

[141] Stefan Woltran. Abstract argumentation – all problems solved? presentation at ECAI-14, 2014. `https://www.dbai.tuwien.ac.at/staff/woltran/ecai2014.pdf`.

[142] Yining Wu and Martin Caminada. A labelling-based justification status of arguments. *Studies in Logic*, 3(4):12–29, 2010.

[143] Yining Wu and Mikolaj Podlaszewski. Implementing crash-resistance and non-interference in logic-based argumentation. *J. Log. Comput.*, 25(2):303–333, 2015.

Received May 2021

On the Incremental Computation of Semantics in Dynamic Argumentation

Gianvincenzo Alfano, Sergio Greco, Francesco Parisi
Department of Informatics, Modeling, Electronics and System Engineering, University of Calabria, Italy
`{g.alfano,greco,fparisi}@dimes.unical.it`

Gerardo I. Simari, Guillermo R. Simari
Department of Computer Science and Engineering, Universidad Nacional del Sur (UNS) & Institute for Computer Science and Engineering (ICIC UNS–CONICET), Bahia Blanca, Argentina
`gis@cs.uns.edu.ar, grsimari@gmail.com`

Abstract

Argumentation frameworks often model dynamic situations where arguments and their relationships (*e.g.,* attacks) frequently change over time. As a consequence, the sets of conclusions (*e.g.,* extensions of abstract argumentation frameworks, or warranted literals for structured argumentation frameworks) often need to be computed again after performing an update. However, as most of the argumentation semantics proposed so far suffer from high computational complexity, computing the set of conclusions from scratch is costly in general. In this work, we address the problems of efficiently recomputing extensions of dynamic abstract argumentation frameworks and warranted literals in dynamic defeasible knowledge bases. In particular, we first present an incremental algorithmic solution whose main idea is that of using an initial extension and the update to identify a (potentially small) portion of an abstract argumentation framework, which is sufficient to compute an extension of the updated framework.

1 Introduction

Computational Argumentation is an established research field in the area of Knowledge Representation and Reasoning (KR) [29; 91; 21; 84; 61], which is central in Artificial Intelligence (AI). An (abstract) argumentation framework [55] is a simple, yet powerful formalism for modeling disputes between agents. The formal meaning of an argumentation framework is given in terms of argumentation semantics, which intuitively tell us the sets of arguments (referred to as *extensions*) that can be accepted together to support a point of view in a discussion. For an abstract argumentation framework, an argument is an abstract entity whose role is entirely determined by its relationships with other arguments. In contrast, DeLP [69] is a well-known argumentation formalism where arguments have an explicit structure as they derive from a knowledge base (DeLP program) consisting of facts and strict and defeasible rules. By considering the structure of arguments, *i.e.,* their inner construction, it becomes possible to analyze reasons for and against a conclusion closely, and the *warrant status* of a claim in the context of a knowledge base represents the main output of a dialectical process.

Although the ideas underlying abstract and structured argumentation frameworks are intuitive and straightforward, most of the argumentation semantics proposed so far suffer from high computational complexity [58; 57; 60; 77; 50]. Most research in the domain of formal argumentation (both in the abstract and structured settings) have focused on *static* frameworks (*i.e.,* frameworks whose structure does not change over time), whereas argumentation frameworks are frequently used for modeling dynamic systems [25; 62; 81; 24; 51; 36; 37]; since, as a matter of fact, the argumentation process is inherently dynamic, this is not surprising. For instance, consider how many times we change our minds after learning something new about a situation that is the focus of our reasoning. There is evidence of that in social network threads [76], where users frequently post new arguments against or supporting other posts, often made by the same users that change their minds. Surprisingly, the definition of evaluation algorithms and the analysis of the computational complexity taking into account such dynamic aspects have been mostly neglected, whereas, in these situations, incremental computation techniques could significantly improve performance. In many cases, especially when few updates at a time are performed, the changes made to a framework can result in small changes to the set of its conclusions—extensions of abstract argumentation frameworks; warranted literals for structured argumentation—and recomputing the whole semantics from scratch can be avoided.

The following is a summary of the contributions of this work:

- By focusing on the most popular argumentation semantics for abstract frameworks, *i.e., complete*, *preferred*, *stable*, *ideal*, and *grounded*, we present a general approach for incrementally solving the following computational task: given an argumentation framework AF, an extension for AF under semantics σ, and an update u, obtain an extension of the updated argumentation framework $u(AF)$

under σ. In other words, we explore the possibility of incrementally solving the task σ-SE of the International Competition on Computational Models of Argumentation (ICCMA) [93]: given an argumentation framework, obtain some σ-extension. The technique consists of the following main steps: (*i*) identification of the *influenced set*, which intuitively consists of the set of arguments whose acceptance status may change after performing an update; (*ii*) identification of a (possibly) smaller argumentation framework, called *reduced* argumentation framework, based on the influenced set and additional information provided by the initial extension; (*iii*) using *any* non-incremental algorithm to compute an extension of the reduced argumentation framework; and (*iv*) obtaining the final extension by merging a portion of the initial extension with the one computed for the reduced argumentation framework.

- We show that the main idea behind the above-described incremental approach can be adapted to *extended* abstract argumentation frameworks, *i.e.,* bipolar argumentation frameworks allowing the presence of attacks and supports, as well as argumentation frameworks with second-order interactions (*e.g.,* attacks towards attacks). This is achieved by leveraging meta-argumentation approaches, which provide ways to transform a more general abstract framework into a Dung framework.

- Intending to minimize wasted effort in the computation of the warrant status of literals of a DeLP program after performing an update, we summarize the necessary elements to develop the updating techniques in DeLP's structured argumentation. Particularly, we focus only on literals that are potentially affected by a given update (namely, *influenced* and *core* literals), and avoids the computation of the status of *inferable* and *preserved* literals.

Organization. As a prelude, we first briefly recall basic notions of abstract argumentation frameworks [55] and then introduce updates in Section 2. The incremental technique for recomputing an extension of an updated abstract argumentation framework under different semantics is presented in Section 3. The main idea behind the above-described incremental approach is then adapted to cope with extended argumentation frameworks in Section 4. Next, in Section 5, we discuss the critical aspects of the technique dealing with structured argumentation in an easy-to-read manner. Related work is discussed in Section 6, and conclusions and directions for future work are drawn in Section 7.

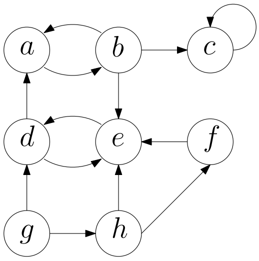


Figure 1: AF_0 of Example 2.1.

2 Abstract Argumentation Frameworks and Updates

We assume the existence of a set $\mathcal{A}rgs$ of *arguments*. An *(abstract) argumentation framework* [55] is a pair $\langle Ar, att\rangle$, where $Ar \subseteq \mathcal{A}rgs$ is a finite set of *arguments*, and $att \subseteq Ar \times Ar$ is a binary relation over Ar whose elements are called *attacks*. Thus, an argumentation framework can be viewed as a directed graph where nodes correspond to arguments and edges correspond to attacks.

Example 2.1 (Running example for abstract argumentation). *Let $AF_0 = \langle Ar_0, att_0\rangle$ be an argumentation framework, where $Ar_0 = \{a, b, c, d, e, f, g, h\}$ and $att_0 = \{(a, b), (b, a), (b, c), (c, c), (d, a), (d, e), (e, d), (b, e), (f, e), (g, d), (g, h), (h, e), (h, f)\}$. The argumentation framework AF_0 is shown in Figure 1.*

Given an argumentation framework $\langle Ar, att\rangle$ and arguments $a, b \in Ar$, we say that a *attacks* b iff $(a, b) \in att$, and that a set $S \subseteq Ar$ attacks b iff there is $a \in S$ attacking b. We use $S^+ = \{b \mid \exists a \in S : (a, b) \in att\}$ to denote the set of all arguments that are attacked by S.

Moreover, we say that $S \subseteq Ar$ *defends* a iff $\forall b \in Ar$ such that b *attacks* a, there is $c \in S$ such that c *attacks* b. A set $S \subseteq Ar$ of arguments is said to be:

- *conflict-free* if there are no $a, b \in S$ such that a *attacks* b;
- *admissible* if it is conflict-free and it defends all its arguments.

An argumentation semantics specifies the criteria for identifying a set of arguments, called *extension*, that can be considered “reasonable” together. A *complete extension* ($\mathcal{CO}$) is an admissible set that contains all the arguments that it defends. A complete extension S is said to be:

- *preferred ($\mathcal{PR}$)* iff it is maximal (w.r.t. $\subseteq$);
- *stable ($\mathcal{ST}$)* iff it attacks every argument in $A \setminus S$;

σ	$\mathcal{E}_\sigma(AF_0)$	$\mathcal{E}_\sigma(AF)$
$\mathcal{CO}$	$\{\{f,g\},\{a,f,g\},\{b,f,g\}\}$	$\{\{g\},\{a,g\},\{b,f,g\}\}$
$\mathcal{PR}$	$\{\{a,f,g\},\{b,f,g\}\}$	$\{\{a,g\},\{b,f,g\}\}$
$\mathcal{ST}$	$\{\{b,f,g\}\}$	$\{\{b,f,g\}\ \}$
$\mathcal{ID}$	$\{\{f,g\}\}$	$\{\{g\}\}$
$\mathcal{GR}$	$\{\{f,g\}\}$	$\{\{g\}\}$

Table 1: Sets of extensions for AF_0 and $AF = +(c,f)(AF_0)$.

- *grounded ($\mathcal{GR}$)* iff it is minimal (w.r.t. $\subseteq$).
- *ideal ($\mathcal{ID}$)* iff it is contained in every preferred extension and it is maximal (w.r.t. $\subseteq$).

Given an argumentation framework AF and a semantics $\sigma \in \{\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}$, $\mathcal{ID}\}$, we use $\mathcal{E}_\sigma(AF)$ to denote the set of σ-extensions for AF, *i.e.,* the set of extensions for AF according to the given semantics σ.

All the above-mentioned semantics except the stable admit at least one extension, and the grounded and ideal admits exactly one extension [55; 56; 41]. Grounded and ideal semantics are called *deterministic* or *unique status* as $|\mathcal{E}_{\mathcal{GR}}(AF)| = |\mathcal{E}_{\mathcal{ID}}(AF)| = 1$, whereas the other above recalled semantics are called *nondeterministic* or *multiple status*. For any AF AF, it holds that $\mathcal{E}_{\mathcal{ST}}(AF) \subseteq \mathcal{E}_{\mathcal{PR}}(AF) \subseteq \mathcal{E}_{\mathcal{CO}}(AF)$, $\mathcal{E}_{\mathcal{GR}}(AF) \subseteq \mathcal{E}_{\mathcal{CO}}(AF)$, and $\mathcal{E}_{\mathcal{ID}}(AF) \subseteq \mathcal{E}_{\mathcal{CO}}(AF)$.

Example 2.2. *The set of admissible sets for the argumentation framework AF_0 shown in Figure 1 is $\{\emptyset, \{b\}$, $\{g\}$, $\{a,g\}$, $\{b,g\}$, $\{f,g\}$, $\{a,g,f\}$, $\{b,g,f\}\}$, and the set $\mathcal{E}_\sigma(AF_0)$ of extensions, with $\sigma \in \{\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{ID}$, $\mathcal{GR}\}$ is as reported in the second column of Table 1.*

2.1 Labelling and Status of Arguments

The argumentation semantics can be also defined in terms of *labelling* [21]. A labelling for an argumentation framework $\langle Ar, att\rangle$ is a total function $\mathcal{L}ab : Ar \rightarrow \{\texttt{in}, \texttt{out}, \texttt{undec}\}$ assigning to each argument a label. $\mathcal{L}(a) = \texttt{in}$ means that argument a is accepted, $\mathcal{L}(a) = \texttt{out}$ means that a is rejected, while $\mathcal{L}(a) = \texttt{undec}$ means that a is undecided.

Let $in(\mathcal{L}) = \{a \mid a \in Ar \wedge \mathcal{L}(a) = \texttt{in}\}$, $out(\mathcal{L}) = \{a \mid a \in Ar \wedge \mathcal{L}(a) = \texttt{out}\}$, and $un(\mathcal{L}) = \{a \mid a \in Ar \wedge \mathcal{L}(a) = \texttt{undec}\}$. In the following, we also use the triple $\langle in(\mathcal{L}), out(\mathcal{L}), un(\mathcal{L})\rangle$ to represent the labelling $\mathcal{L}$.

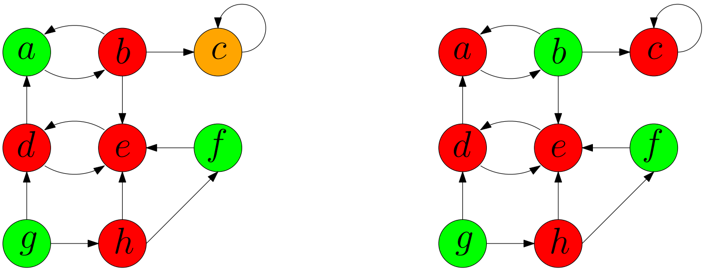


Figure 2: Labelling $\mathcal{L}$ corresponding to the preferred extensions $E_{\mathcal{PR}} \in \mathcal{E}_{AF_0}(\mathcal{PR}) = \langle\{a, f, g\}, \{b, d, e, h\}, \{c\}\rangle$ (left-hand side) and $E^{'}_{\mathcal{PR}} \in \mathcal{E}_{AF_0}(\mathcal{PR}) = \langle\{b, f, g\}, \{a, d, e, h\}, \{c\}\rangle$ (right-hand side). A green (resp., red, orange) node x is such that $\mathcal{L}(x) = \texttt{in}$ (resp., $\texttt{out}$, $\texttt{undec}$).

Given an argumentation framework $AF = \langle Ar, att\rangle$, a labelling $\mathcal{L}$ for AF is said to be *admissible (or legal)* if $\forall a \in in(\mathcal{L}) \cup out(\mathcal{L})$ it holds that

(i) $\mathcal{L}(a) = \texttt{out}$ iff $\exists\, b \in Ar$ such that $(b, a) \in att$ and $\mathcal{L}(b) = \texttt{in}$; and

(ii) $\mathcal{L}(a) = \texttt{in}$ iff $\mathcal{L}(b) = \texttt{out}$ for all $b \in Ar$ such that $(b, a) \in att$.

Moreover, $\mathcal{L}$ is a *complete* labelling iff conditions (i) and (ii) hold for all $a \in Ar$.

Between complete extensions and complete labellings there is a bijective mapping defined as follows: for each extension E there is a unique labelling $\mathcal{L} = \langle E, E^+, Ar \setminus (E \cup E^+)\rangle$ and for each labelling $\mathcal{L}$ there is a unique extension $in(\mathcal{L})$. We say that $\mathcal{L}$ is the labelling *corresponding* to E.

Example 2.3. *Continuing from Example 2.2, $\langle\{a, f, g\}, \{b, d, e, h\}, \{c\}\rangle$ is the labelling corresponding to the preferred extension $E_{\mathcal{PR}} \in \mathcal{E}_{\mathcal{PR}}(AF_0) = \{a, f, g\}$, as shown in Figure 2.*

In the following, we say that the *status of an argument* a w.r.t. a labelling $\mathcal{L}$ (or its corresponding extension $in(\mathcal{L})$) is $\texttt{in}$ (resp., $\texttt{out}$, $\texttt{undec}$) iff $\mathcal{L}(a) = \texttt{in}$ (resp., $\mathcal{L}(a) = \texttt{out}$, $\mathcal{L}(a) = \texttt{undec}$). We will avoid to mention explicitly the labelling (or the extension) whenever it is understood.

2.2 Updating a Dung Argumentation Framework

An argumentation framework typically models a temporary situation, and new arguments and attacks can be added or retracted to take into account new available knowledge.

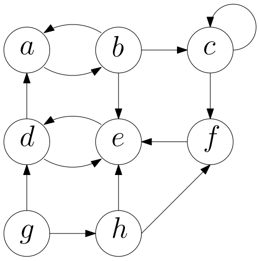


Figure 3: $AF = +(c, f)(AF_0)$

Performing an update on an argumentation framework AF_0 means modifying it into an argumentation framework AF by adding or removing arguments or attacks. We use $+(a, b)$, with $a, b \in Ar_0$ and $(a, b) \notin att_0$, (resp. $-(a, b)$, with $(a, b) \in att_0$) to denote the addition (resp. deletion) of an attack (a, b), and $u(AF_0)$ to denote the application of update $u = \pm(a, b)$ to AF_0 (where ± means either + or −). Applying an update u to an argumentation framework implies that its semantics (set of extensions or labellings). Table 1 reports the sets of extensions for the argumentation framework AF_0 of Figure 1 and for $AF = +(c, f)(AF_0)$ of Figure 3 which is obtained from AF_0 by performing the update $+(c, f)$.

Concerning the addition (resp. deletion) of a set of isolated arguments, it is easy to see that if AF is obtained from AF_0 through the addition (resp. deletion) of a set S of isolated argument, then, let E_0 be an extension for AF_0, $E = E_0 \cup S$ (resp. $E = E_0 \setminus S$) is an extension for AF that can be trivially computed. Of course, if arguments in S are not isolated, for addition we can first add isolated arguments and then add attacks involving these arguments, while for deletion we can first delete all attacks involving arguments in S. Thus we do not consider these kinds of update in the following.

3 Incremental Computation of Extensions in Dynamic Argumentation Frameworks

We tackle the problem of incrementally computing extensions of dynamic argumentation frameworks: given an initial extension and an update (or a set of updates), we devise a technique for computing an extension of the updated argumentation framework under five well-known semantics (*i.e., complete*, *preferred*, *stable*, *grounded*, and *ideal*).

The idea, initially proposed in [74; 75] and then developed in [4], is that of identifying a reduced (updated) argumentation framework sufficient to compute an

update $+(a,b)$		$\mathcal{L}_0(b)$		
		`in`	`undec`	`out`
$\mathcal{L}_0(a)$	`in`			$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}$
	`undec`		$\mathcal{CO}$, $\mathcal{GR}$	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{GR}$
	`out`	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$	$\mathcal{CO}$, $\mathcal{GR}$	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}$

Table 2: Cases for which $E_0 \in \mathcal{E}_\sigma(u(AF_0))$ for $u = +(a,b)$.

extension of the whole argumentation framework and use state-of-the-art algorithms to recompute an extension of the reduced argumentation framework only.

For the sake of presentation, we first present the technique for semantics $\sigma \in \{\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}\}$, and then show how to deal with the ideal semantics in Section 3.3, since the definition of the reduced argumentation framework for the ideal semantics is different from that for the other semantics.

We first give some sufficient conditions ensuring that a given σ-extension for an argumentation framework AF continues to be a σ-extension for the updated argumentation framework $u(AF)$. Then, we introduce the *influenced set* that intuitively consists of the set of arguments whose status may change after performing an update.

Updates Preserving a Given Initial Extension

Given an update $\pm(a,b)$ and an initial extension E_0 corresponding to $\mathcal{L}_0$, for each pair of initial statuses $\mathcal{L}_0(a)$ and $\mathcal{L}_0(b)$ of the arguments involved in the update, Tables 2 and 3 tell us the semantics for which E_0 is still an extension after the update, as stated in the following proposition.

Proposition 3.1 (Irrelevant Updates [5]). *Let AF_0 be an argumentation framework, σ a semantics, $E_0 \in \mathcal{E}_\sigma(AF_0)$ an extension of AF_0 under semantics σ, $\mathcal{L}_0$ the labelling corresponding to E_0, and u an update. If σ is in the cell $\langle \mathcal{L}_0(a), \mathcal{L}_0(b) \rangle$ of Table 2 and $u = +(a,b)$ (resp., of Table 3 and $u = -(a,b)$), then $E_0 \in \mathcal{E}_\sigma(u(AF_0))$.*

The results in Tables 2 and 3 concerning the grounded semantics follow from those in [39; 40], where the principles according to which the grounded extension does not change when attacks are added or removed have been studied.

In the following, given an argumentation framework AF_0 and a σ-extension E_0 for it, we say that an update u is *irrelevant* w.r.t. E_0 and σ iff the conditions of Proposition 3.1 hold. Otherwise, u is said to be *relevant*.

update $-(a,b)$		$\mathcal{L}_0(b)$		
		`in`	`undec`	`out`
$\mathcal{L}_0(a)$	`in`	NA	NA	
	`undec`	NA		$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{GR}$
	`out`	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}$	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{GR}$	$\mathcal{CO}$, $\mathcal{PR}$, $\mathcal{ST}$, $\mathcal{GR}$

Table 3: Cases for which $E_0 \in \mathcal{E}_\sigma(u(AF_0))$ for $u = -(a,b)$.

Example 3.2. *Consider AF_0 of Figure 1 and its sets of extensions listed in the second column of Table 1. $E_0 = \{b,f,g\}$ is an extension according to semantics $\sigma \in \{\mathcal{CO},\mathcal{PR},\mathcal{ST}\}$. Thus, $\mathcal{L}_0(c) =$ `out` and $\mathcal{L}_0(f) =$ `in`, and using Proposition 3.1 it follows that for update $u = +(c,f)$ E_0 is still an extension of $u(AF_0)$ (see the last column of Table 1). Thus $+(c,f)$ is irrelevant w.r.t. E_0 and σ.*

In contrast, $+(c,f)$ is relevant w.r.t. $E_0 = \{a,f,g\}$ and any semantics (in this case $\mathcal{L}_0(c) =$ `undec` and $\mathcal{L}_0(f) =$ `in`, and no semantics is listed in the cell $\langle$`undec`, `in`$\rangle$ of Table 2).

It is important to note that Tables 2 and 3 are not meant to be exhaustive, as more conditions can be found for which a σ-extension is preserved after an update. For instance, for the grounded semantics, the initial extension is preserved also if $\mathcal{L}_0(a) =$ `out` and $\mathcal{L}_0(b) =$ `in` and argument a of updated $+(a,b)$ is not reachable from b. Here we provided a simple set of conditions that can be easily checked by just looking at the initial labelling $\mathcal{L}_0$. The technique for the incremental computation can be trivially extended by considering a more general set of such conditions.

Influenced Set

Given an argumentation framework, an update, and an initial σ-extension of the considered framework, the influenced set consists of the arguments whose acceptance status (according to the semantics σ) may change after performing the update. For irrelevant updates, the influenced set will be empty, as in this case, the initial extension can be immediately returned as an extension of the updated argumentation framework. If none of the conditions of Proposition 3.1 hold (*i.e.,* the update is relevant), then the influenced set may turn out to be not empty. In such case, the influenced set will be used to delineate a portion of the argumentation framework, called *reduced* argumentation framework, that we will use to recompute (a portion of) an extension for the updated argumentation framework.

Given an argumentation framework $AF = \langle Ar, att\rangle$ and an argument $b \in Ar$, we use $Reach_{AF}(b)$ to denote the set of arguments that are reachable from b in the graph AF.

Definition 3.3 (Influenced Set [5])**.** Let $AF = \langle Ar, att\rangle$ be an argumentation framework, $u = \pm(a, b)$, E an extension of AF under semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{ID}, \mathcal{GR}\}$, and let

- $INF_0(u, AF, E) = \begin{cases} \emptyset \;\; \textit{if } u \textit{ is irrelevant w.r.t. } E \textit{ and } \sigma \textit{ or } \exists(z, b) \in att \\ \qquad \textit{s.t. } z \neq a \wedge z \in E \;\wedge\; z \notin Reach_{AF}(b); \\ \{b\} \; \textit{otherwise}; \end{cases}$
- $INF_{i+1}(u, AF, E) = INF_i(u, AF, E) \;\cup\; \{y \mid \exists(x, y) \in att \textit{ s.t. } x \in INF_i(u, AF, E) \wedge \nexists(z, y) \in att \textit{ s.t. } z \in E \wedge z \notin Reach_{AF}(b)\}$.

The *influenced set* of u w.r.t. AF and E is $INF(u, AF, E) = INF_n(u, AF, E)$ such that $INF_n(u, AF, E) = INF_{n+1}(u, AF, E)$.

Thus, the set of arguments that are influenced by an update of b's status are those that can be reached from b without using any intermediate argument y whose status is known to be `out` because it is determined by an argument $z \in E$ that is not reachable from (and thus not influenced by) b.

Example 3.4. *Consider the argumentation framework $AF_0 = \langle Ar_0, att_0\rangle$ of Figure 1 and the update $u = +(c, f)$. We have that $Reach_{AF_0}(f) {=} Ar_0 \setminus \{g, h\}$. The influenced set depends on the initial extension chosen. For the (preferred) extension $\{b, f, g\}$ of Example 3.2, we have that the influenced set is empty as u is irrelevant. In contrast, for the (preferred) extension $E_0 = \{a, f, g\}$, the influenced set is $INF(u, AF_0, E_0) = \{f, e\}$. Indeed, $d \notin INF(u, AF_0, E_0)$ since it is attacked by $g \in E_0$ which is not reachable from f. Thus the arguments that can be reached from d do not belong to $INF(u, AF_0, E_0)$. If we consider the initial grounded extension $\{f, g\}$, then $\{f, e\}$ turns out once again to be the influenced set.*

Reduced Argumentation Framework

Given the influenced set, we define a subgraph, called *reduced* argumentation framework, that will be used to compute the status of the influenced arguments, thus providing an extension that will be combined with that of initial argumentation framework to obtain an extension of the updated argumentation framework, for every semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$.

For any argumentation framework $AF = \langle Ar, att\rangle$ and set $S \subseteq Ar$ of arguments, we denote with $AF{\downarrow_S} = \langle S, att \cap (S \times S)\rangle$ the subgraph of AF induced by arguments in S. Moreover, given two argumentation frameworks $AF_1 = \langle Ar_1, att_1\rangle$ and $AF_2 =$

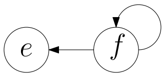


Figure 4: $RAF(+(c,f), AF_0, \{a,f,g\})$

$\langle Ar_2, att_2\rangle$, we denote as $AF_1 \sqcup AF_2 = \langle Ar_1 \cup Ar_2, att_1 \cup att_2\rangle$ the *union* of the two argumentation frameworks.

Definition 3.5 (Reduced Argumentation Framework [5]). *Let $AF_0 = \langle Ar_0, att_0\rangle$ be an argumentation framework, $E_0 \in \mathcal{E}_\sigma(AF_0)$ an extension for AF_0 under a semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$, and $u = \pm(a,b)$ an update. Let $AF = \langle Ar, att\rangle$ be the argumentation framework updated using u. The reduced argumentation framework for AF_0 w.r.t. E_0 and u (denoted as $RAF(u, AF_0, E_0)$) is as follows.*

- *$RAF(u, AF_0, E_0)$ is empty if $INF(u, AF_0, E_0)$ is empty.*
- *$RAF(u, AF_0, E_0) = AF{\downarrow_{INF(u,AF_0,E_0)}} \sqcup AF_1 \sqcup AF_2$ where:*
  - *(i) AF_1 is the union of the frameworks $\langle\{a,b\}, \{(a,b)\}\rangle$ s.t. $(a,b) \in att$, $a \notin INF(u, AF_0, E_0)$, $a \in E_0$, and $b \in INF(u, AF_0, E_0)$;*
  - *(ii) AF_2 is the union of the frameworks $\langle\{c\}, \{(c,c)\}\rangle$ s.t. there is $(e,c) \in att$, $e \notin INF(u, AF_0, E_0)$, $e \notin (E_0 \cup E_0^+)$, and $c \in INF(u, AF_0, E_0)$.*

Hence, the argumentation framework $RAF(u, AF_0, E_0)$ contains, in addition to the subgraph of $u(AF_0)$ induced by $INF(u, AF_0, E_0)$, additional nodes and edges containing needed information on the "external context", *i.e.,* information about the status of arguments which are attacking some argument in $INF(u, AF_0, E_0)$. Specifically, if there is in $u(AF_0)$ an edge from an uninfluenced node a whose status in `in` to an influenced node b, then we add the edge (a,b) so that, as a does not have incoming edges in $RAF(u, AF_0, E_0)$, its status is confirmed to be `in`. Moreover, if there is in $u(AF_0)$ an edge from an uninfluenced node e to an influenced node c such that e is `undec`, we add edge (c,c) to $RAF(u, AF_0, E_0)$ so that the status of c cannot be `in`. Using fake arguments/attacks to represent external contexts has been exploited in [20] where decomposability properties of argumentation semantics are investigated.

Example 3.6. *For our running example, if $E_0 = \{a,f,g\}$ and $u = +(c,f)$, the reduced argumentation framework $RAF(+(c,f), AF_0, E_0)$ consists of the subgraph induced by $INF(u, AF_0, E_0) = \{f,e\}$ plus the edge (f,f) as there is the attack (c,f) in the updated argumentation framework from a non influenced argument c labelled as* `undec` *toward the influenced argument f. Hence, $RAF(+(c,f), AF_0, E_0)$ $= \langle\{e,f\}, \{(f,f),(f,e)\}\rangle$ as shown in Figure 4.*

The following theorem states that, for every semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$, an extension for the updated argumentation framework can be obtained by the union of an extension of the reduced argumentation framework and the projection of the initial extension on the uninfluenced part.

Theorem 3.7 ([5]). *Let AF_0 be an argumentation framework, $AF = u(AF_0)$ be the argumentation framework resulting from performing update $u = \pm(a, b)$ on AF_0, and $E_0 \in \mathcal{E}_\sigma(AF_0)$ be an extension for AF_0 under a semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$. If $\mathcal{E}_\sigma(RAF(u, AF_0, E_0))$ is not empty, then there is an extension $E \in \mathcal{E}_\sigma(AF)$ for the updated argumentation framework AF such that $E = (E_0 \setminus INF(u, AF_0, E_0)) \cup E_d$, where E_d is a σ-extension for reduced argumentation framework $RAF(u, AF_0, E_0)$.*

Example 3.8. *Continuing with our example, for the preferred semantics, let $E_0 = \{a, f, g\}$ and $u = +(c, f)$, we have that $INF(u, AF_0, E_0) = \{f, e\}$, and $RAF(+(c, f), AF_0, E_0) = \langle\{e, f\}, \{(f, f), (f, e)\}\rangle$. Thus, using the theorem, there is an extension E of the updated argumentation framework such that $E = (\{a, f, g\} \setminus \{f, e\}) \cup E_d$ where $E_d = \emptyset$ is a preferred extension of the reduced argumentation framework. In fact, $E = \{a, g\} \in \mathcal{E}_{\mathcal{PR}}(u(AF_0))$.*

It is worth noting that the set of extensions of an argumentation framework can be empty only for the stable semantics. Thus, in the case that this happens for the reduced argumentation framework (*i.e.*, $\mathcal{E}_\sigma(RAF(u, AF_0, E_0)) = \emptyset$), the theorem does not give a method to determine an extension of the updated argumentation framework, as shown in the following example.

Example 3.9. *Consider the two stable extensions $\{a, c\}$ and $\{a, d, e\}$ for AF_0 and the update $u = +(d, d)$. Depending on the initial extension, the influenced set is either $INF(u, AF, \{a, c\}) = \emptyset$ (as u is irrelevant w.r.t. $\{a, c\}$ and $\mathcal{ST}$) or $INF(u, AF, \{a, d, e\})$*
$= \{d\}$. Thus, starting from the extension $\{a, c\}$ we directly know $\{a, c\}$ is a stable extension of the updated argumentation framework. However, starting from $\{a, d, e\}$, the reduced argumentation framework will be $RAF(u, AF_0, \{a, d, e\}) = \langle\{d\}, \{(d, d)\}\rangle$, which has no stable extension. In this case, the theorem does not provide a stable extension of the updated argumentation framework, thought a stable extension exists: that obtained by starting from the initial extension $\{a, c\}$.

Note that, if we consider the preferred semantics, for which the starting extensions are again $\{a, c\}$ and $\{a, d, e\}$, a preferred extension of the updated argumentation framework can be obtained no matter what starting extension is chosen. In particular, as the preferred extension for reduced argumentation framework $\langle\{d\}, \{(d, d)\}\rangle$ is the empty set, it follows that $(\{a, d, e\} \setminus \{d\}) \cup \emptyset = \{a, e\}$ is a preferred extension of the updated argumentation framework.

Algorithm 1 Incr-Alg($AF_0, u, \sigma, E_0, \mathsf{Solver}_\sigma$) [5]

```
Input: $AF_0 = \langle Ar_0, att_0 \rangle$,
    update $u = \pm(a, b)$,
    semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$,
    extension $E_0 \in \mathcal{E}_\sigma(AF_0)$,
    function $\mathsf{Solver}_\sigma(AF)$ returning a $\sigma$-extension of $AF$ if it exists, $\perp$ otherwise;
Output: A $\sigma$-extension $E \in \mathcal{E}_\sigma(u(AF_0))$ if it exists, $\perp$ otherwise;
 1: $S = \mathit{INF}(u, AF_0, E_0)$;
 2: if $(S = \emptyset)$ then
 3:     return $E_0$;
 4: end if
 5: $AF_d = \mathit{RAF}(u, AF_0, E_0)$;
 6: Let $E_d = \mathsf{Solver}_\sigma(AF_d)$;
 7: if $(E_d \neq \perp)$ then
 8:     return $E = (E_0 \setminus S) \cup E_d$;
 9: else
10:     return $\mathsf{Solver}_\sigma(u(AF_0))$;
11: end if
```

3.1 Incremental Algorithm

Algorithm 1 computes an extension of an updated argumentation framework [5]. Besides taking as input an initial argumentation framework AF_0, an update u, a semantics $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$, and an extension $E_0 \in \mathcal{E}_\sigma(AF_0)$, it also takes as input a function that computes a σ-extension for an argumentation framework, if any. In particular, function $\mathsf{Solver}_\sigma(AF)$ will be used to compute an extension of the reduced argumentation framework, which will be then combined with the portion of the initial extension that does not change in order to obtain an extension for the updated argumentation framework (as stated in Theorem 3.7).

More in detail, Algorithm 1 works as follows. First, the influenced set of AF_0 w.r.t. update u and the given initial extension E_0 is computed (Line 1). If it is empty, then E_0 will be still an extension of the updated argumentation framework under the given semantics σ, and thus it is returned (Line 3). Otherwise, the reduced argumentation framework AF_d is computed at Line 5, and function Solver_σ is invoked to compute a σ-extension of AF_d, if any. If $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{GR}\}$, then AF_d will have an extension E_d, which is combined with $E_0 \setminus S$ at Line 8 to get an extension for the updated argumentation framework. For the stable semantics, if $\mathcal{E}_{\mathcal{ST}}(\mathit{RAF}(u, AF_0, E_0))$ is not empty, then the algorithm proceeds as for the other

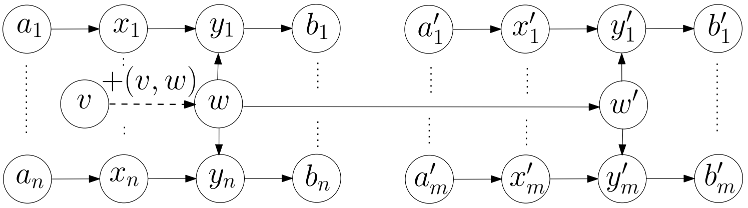


Figure 5: Simulating multiple updates by a single one.

semantics (Line 8). Otherwise, function Solver_σ is invoked to compute a stable extension of the whole updated argumentation framework $u(AF_0)$, if any.

The soundness and completeness of the algorithm follows from the result of Theorem 3.7 and the soundness and completeness of function $\mathsf{Solver}_\mathcal{S}$ used.

Theorem 3.10 (Soundness and Completeness [5]). *Let AF_0 be an argumentation framework, $u = \pm(a,b)$, and $E_0 \in \mathcal{E}_\sigma(AF_0)$ an extension for AF_0 under $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$. If Solver_σ is sound and complete then Algorithm 1 computes $E \in \mathcal{E}_\sigma(u(AF_0))$ if $\mathcal{E}_\sigma(u(AF_0))$ is not empty, otherwise it returns $\bot$.*

3.2 Applying Multiple Updates Simultaneously

The approach described in the previous section extends to the case of multiple updates, *i.e.,* set of updates performed simultaneously. In fact, performing a set of updates $U = \{+(a_1,b_1),\dots,+(a_n,b_n),\ -(a'_1,b'_1),\ \dots,-(a'_m,b'_m)\}$ on AF_0 can be reduced to performing a single update $+(v,w)$ on an argumentation framework $AF^U_{E_0}$ whose definition depends on both the set of updates U and the initial σ-extension E_0, as explained in what follows.

Given a set U of updates for an argumentation framework AF_0, and a σ-extension E_0 for AF_0, we use U^* to denote the subset of U consisting of the relevant updates (that is, the updates in U for which the conditions of Proposition 3.1 do not hold).

The argumentation framework $AF^U_{E_0}$ for applying a set U^* of relevant updates is obtained from AF_0 by (i) adding arguments x_i, y_i and the chain of attacks between a_i and b_i as shown in Figure 5, for each update $+(a_i,b_i) \in U^*$; (ii) replacing each attack (a'_j,b'_j) in AF_0 with the chain of attacks between a'_j and b'_j as shown in Figure 5, for each update $-(a_j,b_j) \in U^*$; and (iii) adding the new arguments v, w, w' and the attacks involving them as shown in Figure 5. The following definition considers a general set of updates which includes also irrelevant updates.

Definition 3.11 (AF for applying a set of updates [75]).
Let $AF_0 = \langle Ar_0, att_0\rangle$ be an argumentation framework, and E_0 a σ-extension for AF_0. Let

- $att^+ = \{(a_1, b_1), \ldots, (a_n, b_n)\} \subseteq (Ar_0 \times Ar_0) \setminus att_0$, *and*
- $att^- = \{(a'_1, b'_1), \ldots, (a'_m, b'_m)\} \subseteq att_0$

such that $att^+ \cap att^- = \emptyset$ *be two sets of attacks.*

Let $U = \{+(a_i, b_i) \mid (a_i, b_i) \in att^+\} \cup \{-(a_j, b_j) \mid (a_j, b_j) \in att^-\}$ *be a set of updates, and* $U^* \subseteq U$ *be the set of relevant updates w.r.t.* E_0 *and* σ*. Then,* $AF^U_{E_0} = \langle Ar^U, att^U \rangle$ *denotes the argumentation framework obtained from* AF_0 *as follows:*

- $Ar^U = Ar_0 \cup \{x_i, y_i \mid +(a_i, b_i) \in U^*\} \cup \{x'_j, y'_j \mid -(a_j, b_j) \in U^*\} \cup \{v, w, w'\}$, *where all* x_i, y_i, x'_j, y'_j, w, w', *and* v *are new arguments not occurring in* Ar_0, *and*
- $$\begin{aligned} att^U = {} & (att_0 \setminus att^-) \cup \{(a_i, b_i) \mid +(a_i, b_i) \in (U \setminus U^*)\} \cup \\ & \{(a_i, x_i), (x_i, y_i), (y_i, b_i) \mid +(a_i, b_i) \in U^*\} \cup \\ & \{(a_j, x'_j), (x'_j, y'_j), (y'_j, b_j) \mid -(a_j, b_j) \in U^*\} \cup \\ & \{(w, y_i) \mid +(a_i, b_i) \in U^*\} \cup \\ & \{(w', y'_j) \mid -(a_j, b_j) \in U^*\} \cup \{(w, w')\}. \end{aligned}$$

It is worth noting that, in the definition above, each argument x_i, y_i, x'_i, and y'_i is assumed to be unique and non-identical for every attack (a_i, b_i).

The following theorem states that every extension of the argumentation framework AF obtained by performing on AF_0 all the updates in U corresponds to an extension of $+(v, w)(AF^U_{E_0})$, where $+(v, w)$ is a single attack update.

Theorem 3.12 ([75; 5])**.** *Let* $AF_0 = \langle Ar_0, att_0 \rangle$ *be an argumentation framework,* E_0 *a* σ*-extension for* AF_0*, and* U *a set of updates. Let* AF *be the argumentation framework obtained from* AF_0 *by performing all updates in* U *on it. Then, for any semantics* $\sigma \in \{\mathcal{CO}, \mathcal{PR}, \mathcal{ST}, \mathcal{GR}\}$ $E \in \mathcal{E}_\sigma(AF)$ *iff there is* $E^U \in \mathcal{E}_\sigma(+(v, w)(AF^U_{E_0}))$ *such that* $E^U \cap Ar_0 = E$.

3.3 Dealing with the Ideal Semantics

Algorithm 1 can be extended to deal with the ideal semantics. The only difference is that we need a new definition of reduced argumentation framework since, as illustrated in the following example, that of Definition 3.5 does not work for the ideal semantics.

Example 3.13. *Consider the argumentation framework* $AF_0 = \langle \{a, b, c, d\}, \{(a, b), (b, a), (c, d), (d, c), (a, c), (b, c)\} \rangle$ *and the update* $u = -(b, c)$*. The ideal extension of* AF_0 *is* $E_0 = \{d\}$ *(i.e., arguments* a *and* b *are both labeled as* `undec`*). The influenced set is* $INF(u, AF_0, E) = \{c, d\}$*. However, the RAF obtained by applying*

Definition 3.5 is $\langle\{c, d\}, \{(c, c), (c, d), (d, c)\}\rangle$, its ideal extension is $\{d\}$, and applying the result of Theorem 3.7 we would obtain that $\{d\}$ is still the ideal extension for $u(AF_0)$. But this is not correct, as the ideal extension for $u(AF_0)$ is the empty set.

Before defining the reduced framework for the ideal semantics, we define the paths providing the information on the "context" outside the influenced set $\mathit{INF}(u, AF, E)$ that needs to be added to determine the new status of the arguments influenced by update u w.r.t. the ideal extension E.

Given an argumentation framework $AF = \langle Ar, att\rangle$ with ideal extension E and a set $S \subseteq Ar$, we use $\mathit{Node}(AF, S, E)$ (resp. $\mathit{Edge}(AF, S, E)$) to denote a set of arguments $x_1, \dots, x_n$ (resp. attacks $(x_1, x_2), \dots, (x_{n-1}, x_n)$) in AF such that there is a path $x_1 \dots x_n$ in AF with $x_n \in S$, $x_1, \dots, x_{n\text{-}1} \notin S$ and $x_1, \dots, x_{n\text{-}1} \notin E \cup E^+$ (*i.e.,* $x_1, \dots, x_{n\text{-}1}$ are `undec`). Essentially, if S is the influenced set of an update, to determine the status of nodes in S we must also consider all nodes and attacks occurring in paths (of any length) ending in S whose nodes outside S are all labeled as `undec`. The motivation to also consider the paths ending in S is that some of the undecided arguments occurring in these paths could be labelled `in` or `out` in some preferred labelling and, therefore, together they could determine a change in the status of nodes in S.

Definition 3.14. (Reduced Argumentation Framework for Ideal Semantics [75; 8])
Let $AF_0 = \langle Ar_0, att_0\rangle$ be an argumentation framework, E_0 be the ideal extension for AF_0, and $u = \pm(a, b)$ an update. Let $AF = \langle Ar, att\rangle$ be the argumentation framework updated by using u. The reduced argumentation framework for AF_0 w.r.t. E_0 and u (denoted as $RAF_{\mathcal{ID}}(u, AF_0, E_0)$) is as follows.

- *$RAF_{\mathcal{ID}}(u, AF_0, E_0)$ is empty if $\mathit{INF}(u, AF_0, E_0)$ is empty.*
- *$RAF_{\mathcal{ID}}(u, AF_0, E_0) = AF{\downarrow}_{\mathit{INF}(u,AF_0,E)} \sqcup AF_1 \sqcup AF_2$ where:*
  - *(i) AF_1 is the union of the frameworks $\langle\{a, b\}, \{(a, b)\}\rangle$ such that $(a, b) \in att$ and $a \notin \mathit{INF}(u, AF_0, E_0)$, $a \in E_0$, and $b \in \mathit{INF}(u, AF_0, E_0)$;*
  - *(ii) AF_2 is the union of the frameworks $\langle \mathit{Node}(AF, \mathit{INF}(u, AF_0, E_0), E_0)$ and $\mathit{Edge}(AF, \mathit{INF}(u, AF_0, E_0), E_0)\rangle$.*

Example 3.15. *For the argumentation framework AF_0 of running example (see Figures 1 and 3), where the initial ideal extension is $E_0 = \{f, g\}$ and $u = +(c, f)$, the reduced argumentation framework $RAF_{\mathcal{ID}}(+(c, f), AF_0, E_0)$ consists of the subgraph induced by $\mathit{INF}(u, AF_0, E_0) = \{f, e\}$ plus the sub-graph consisting of the paths of undecided arguments ending in the influenced set, that is, $AF_2 = \langle\{a, b, c\}, \{(a, b), (b, a),$*

$(b, c), (c, c),$
$(c, f)\}\rangle$. *Hence,* $RAF_{\mathcal{ID}}(+(c, f), AF_0, E_0) = \langle\{a, b, c, e, f\}, \{(a, b), (b, a), (b, c), (c, c), (c, f), (f, e)\}\rangle$. *The ideal extension of the reduced framework is the empty set.*

It can be shown that the result of Theorem 3.7 also holds for the case of the ideal semantics [8]. By applying that result, we obtain that the (updated) ideal extension for the updated argumentation framework of Example 3.15 is $(\{f, g\} \setminus \{f, e\}) \cup \emptyset = \{g\}$ (see Table 1).

Example 3.16. *Consider again the argumentation framework* AF_0 *and the update* u *of Example 3.13, where the ideal extension of* AF_0 *is* $E_0 = \{d\}$ *and* $INF(u, AF_0, E) = \{c, d\}$.

Thus, $RAF_{\mathcal{ID}}(u, AF_0, E_0) = AF{\downarrow}_{INF(u,AF_0,E_0)} \sqcup AF_1 \sqcup AF_2$ *where:*

- $AF{\downarrow}_{INF(u,AF_0,E)} = \langle\{c, d\}, \{(c, d), (d, c)\}\rangle$,
- $AF_1 = \langle\emptyset, \emptyset\rangle$ *and*
- $AF_2 = \langle\{a, b, c\}, \{(a, b), (b, a), (a, c)\}\rangle$.

That is, $RAF_{\mathcal{ID}}(u, AF_0, E_0) = \langle\{a, b, c, d\}, \{(a, b), (b, a), (c, d), (d, c), (a, c)\}\rangle$, *and its ideal extension is* $\emptyset$. *Thus, using the result of Theorem 3.7, we obtain that the ideal extension for the updated argumentation framework* $u(AF_0)$ *is the empty set.*

Finally, Algorithm 1 can be used to compute the updated ideal extension of a given argumentation framework by using $AF_d = RAF_{\mathcal{ID}}(u, AF_0, E_0)$ at Line 5 and an external solver that computes the ideal extension of the reduced argumentation framework.

In the next two sections, we will deal with other possible ways to apply the incremental algorithm in other approaches to formal (computational) argumentation. First, Section 4 deals with bipolarity and extended argumentation frameworks, while Section 5 centers on Defeasible Logic Programming (DeLP) as a structured argumentation formalism.

4 Bipolarity and Second-Order Attacks

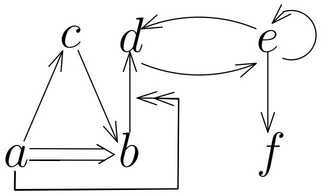
Dung's framework has been extended along several dimensions; for instance, see [22; 83; 96]. The proposed incremental approach can be applied to different kinds of abstract argumentation frameworks that extend Dung's model. The main idea is that of using meta-argumentation approaches, which provide ways to transform a more general abstract framework into a Dung framework, and apply the incremental technique on the meta argumentation framework [4; 6; 7].

Bipolarity in argumentation is discussed in [17], where a survey of the use of bipolarity is given, as well as a formal definition of bipolar argumentation frameworks, which extend Dung's concept of argumentation framework by also including the relation of support between arguments. The notion of support has been found to be useful in many application domains, including decision making [16]. Several interpretations of the notion of support have been proposed in the literature [17; 46; 47; 48; 38; 96] (see [53] for a comprehensive survey). In this work, we focus on *deductive* support [38; 96] which is intended to capture the following intuition: if argument a supports argument b then the acceptance of a implies the acceptance of b, and thus the non-acceptance of b implies the non-acceptance of a. However, the approach presented in this section can be adapted to work also with *necessary* support [89; 88; 23] due to the duality between these two kinds of interpretations of the support relation [53]. The acceptability of arguments in the presence of a support relation was first investigated in [46]. Later on, to handle bipolarity in argumentation, [47; 48] proposed an approach based on using the concept of *coalition* of arguments, where sets of arguments are considered as a group that plays the role of an argument and defeats then occur between coalitions. However, when considering a deductive interpretation of support [38; 96], coalitions may lead to counter-intuitive results [53]; nevertheless, they are useful in contexts where support is interpreted differently.

Furthermore, other abstract argumentation frameworks have been considered, such as Extended Argumentation Frameworks, which extend bipolar argumentation frameworks by modelling (apart from attacks/supports between arguments) also attacks towards an attack or a support (called second-order attacks). Thanks to a meta argumentation approach, an extended argumentation framework can be converted into an abstract argumentation framework by using additional meta-arguments as well as attacks between them to model supports and second-order attacks.

In the following, we discuss how to extend the incremental technique to deal with extended argumentation frameworks. An *Extended Argumentation Framework* [38] is a quadruple $\langle Ar, att, sup, s\text{-}att\rangle$, where where (i) $Ar \subseteq \mathcal{A}rgs$, (ii) $att \subseteq Ar \times Ar$, (iii) $sup \subseteq Ar \times Ar$ is a binary relation over Ar whose elements are called *supports*, (iv) $att \cap sup = \emptyset$,and (v) $s\text{-}att$ is a binary relation over $Ar \times \ (att \cup sup)$ whose elements are called *second-order attacks*.

In the following, a second-order attack from an argument a to an attack (b, c) will be denoted as $(a \twoheadrightarrow (b \rightarrow c))$, while an attack from an argument a to a support (b, c) will be denoted as $(a \twoheadrightarrow (b \Rightarrow c))$. Thus, a Dung argumentation framework is an extended argumentation framework of the form $\langle Ar, att, \emptyset, \emptyset\rangle$, while a bipolar argumentation framework is extended argumentation framework of the form $\langle Ar, att, sup, \emptyset\rangle$.



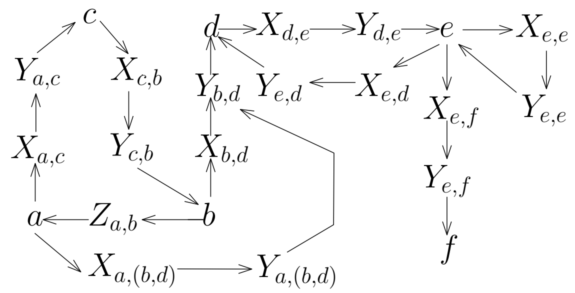


Figure 7: Meta framework for EF_0 of Example 4.1.

Example 4.1. *Consider the extended argumentation framework* $EF_0 = \langle Ar_0, att_0, sup_0, s\text{-}att_0 \rangle$ *where:*

- $Ar_0 = \{a, b, c, d, e, f\}$ *is the set of arguments;*
- $att_0 = \{(a, c),\ (c, b),\ (b, d),\ (d, e), (e, d),\ (e, e),\ (e, f)\}$ *is the set of attacks;*
- $sup_0 = \{(a, b)\}$ *is the set of supports; and*
- $s\text{-}att_0 = \{(a, (b, d))\}$ *is the set of second-order attacks.*

The corresponding graph is shown in Figure 6, where second-order attacks are drawn using double-headed arrows.

The semantics of an extended argumentation framework can be given by means of the following meta argumentation framework.

Definition 4.2 (Meta Argumentation Framework [38])**.** *The meta argumentation framework for* $EF = \langle Ar, att, sup, s\text{-}att \rangle$ *is* $MF = \langle Ar^m, att^m \rangle$ *where:*

- $Ar^m \;=\; Ar \cup \{X_{a,b}, Y_{a,b} \mid (a, b) \in att\} \cup \{Z_{a,b} \mid (a, b) \in sup\} \cup$
  $\{X_{a,(b,c)}, Y_{a,(b,c)} \mid (a, (b, c)) \in s\text{-}att, (b, c) \in att\}$
- $att^m {=} \{(a, X_{a,b}), (X_{a,b}, Y_{a,b}), (Y_{a,b}, b) | (a, b) \in att\} \cup$
  $\{(b, Z_{a,b}), (Z_{a,b}, a) \mid (a, b) \in sup\} \cup$
  $\{(a, X_{a,(b,c)}),\ (X_{a,(b,c)}, Y_{a,(b,c)}),\ (Y_{a,(b,c)},\ Y_{b,c}) \mid$
  $(a, (b, c)) \in s\text{-}att, (b, c) \in att\} \cup$
  $\{(a, X_{a,(b,c)}),\ (X_{a,(b,c)},\ Y_{a,(b,c)}),\ (Y_{a,(b,c)},\ Z_{b,c}) \mid$
  $(a, (b, c)) \in s\text{-}att, (b, c) \in sup\}$

The meaning of meta-arguments $X_{a,b}$, $Y_{a,b}$ and $Z_{a,b}$ is as follows. $X_{a,b}$ represents the fact that the corresponding attack (a, b) is "negligible" in the extended argumentation framework—it belongs to an extension of the meta argumentation framework iff a does not belong to an extension of the extended argumentation framework. On the other hand, $Y_{a,b}$ represents the fact that (a, b) is "significant" in the extended argumentation framework, and it belongs to an extension of the meta argumentation framework iff argument b does not. Finally, meta-argument $Z_{a,b}$ represents a support relation between a and b: it does not belong to an extension for the meta argumentation framework iff the supported argument b is accepted in the deductive model of support.

Moreover, a second order attack of the form $(a \twoheadrightarrow (b \rightarrow c))$ is encoded as an attack towards the meta-argument $Y_{b,c}$ (that represents the fact that (b, c) is "significant"), while an attack of the form $(a \twoheadrightarrow (b \Rightarrow c))$ is encoded as an attack toward the meta-argument $Z_{b,c}$. The meta argumentation framework for the extended argumentation framework of Example 4.1 is shown in Figure 7.

Extensions for an extended argumentation framework EF are obtained from extensions for its meta argumentation framework MF: E is an σ-extension for EF iff $E^m \in \mathcal{E}_\sigma(MF)$ and $E = E^m \cap Ar$, where Ar is the set of arguments of EF. Using this relationship, the notion of labelling can be extended to extended argumentation frameworks as well. As done in [38], in the following we will focus on the preferred and stable semantics. However, the technique can be also applied to grounded, ideal, and complete semantics by means of meta argumentation approach.

Example 4.3. *For the meta argumentation framework MF of Figure 7, we have the following preferred extensions (which are also stable extensions): (i)* $\{a, b, d, f, Y_{a,c}, X_{c,b}, Y_{d,e}, Y_{a,(b,d)}, X_{e,e}, X_{e,d}, X_{e,f}, \}$, *which corresponds to the extension* $\{a, b, d, f\}$ *of the extended argumentation framework of Example 4.1, and (ii)* $\{c, d, f, X_{a,c}, Y_{c,b}, Z_{a,b}, X_{b,d}, Y_{d,e}\ X_{e,e}, X_{e,d}, X_{e,f}, X_{a,(b,d)}\}$, *which corresponds to the extension* $\{c, d, f\}$ *of the extended argumentation framework.*

Updates over Extended Argumentation Frameworks For extended argumentation frameworks we also consider updates consisting of additions and deletions of support relations and second-order attacks, in addition to the attack updates considered for Dung's frameworks. Specifically, the addition (resp., deletion) of a support relation from an argument a to an argument b will be denoted as $+(a \Rightarrow b)$ (resp. $-(a \Rightarrow b)$). Analogously, the addition (resp., deletion) of a second-order attack from an argument a to an attack (b, c) will be denoted as $+(a \twoheadrightarrow (b \rightarrow c))$ (resp., $-(a \twoheadrightarrow (b \rightarrow c))$). Finally, if (b, c) is a support, then the update will be denoted as $+(a \twoheadrightarrow (b \Rightarrow c))$ (resp., $-(a \twoheadrightarrow (b \Rightarrow c))$). We use $u(EF_0)$ to denote the extended argumentation framework resulting from the application of update u to an initial extended framework EF_0.

We introduce the compact argumentation framework for performing an update on extended argumentation frameworks—it will be used in a variant of Algorithm 1 for the incremental computation. The definition builds on (the compact version of) that proposed in [38] and considers additional meta-arguments and attacks that will allow us to simulate addition updates to be performed on the extended argumentation framework by means of single updates performed on the corresponding (compact) meta argumentation framework.

Definition 4.4 (Compact Argumentation Framework [7]). *Let $EF = \langle Ar, att, sup, s\text{-}att\rangle$ be an extended argumentation framework, and u an update of one of the following forms:*

$$\bullet\ u = \pm(e \to f) \qquad \bullet\ u = \pm(e \Rightarrow f)$$
$$\bullet\ u = \pm(e \twoheadrightarrow (g \to h)) \quad \bullet\ u = \pm(e \twoheadrightarrow (g \Rightarrow h)).$$

The compact argumentation framework for EF w.r.t. u is $CF(EF, u) = \langle Ar^m, att^m\rangle$ where:

- $$\begin{aligned} Ar^m = A \cup \{&Z_{a,b} \mid (a,b) \in sup\} \cup \\ \{&X_{c,d}, Y_{c,d} \mid (e,(c,d)) \in s\text{-}att, (c,d) \in att\} \cup \\ \{&Z_{e,f} \mid u = +(e \Rightarrow f)\} \cup \\ \{&X_{g,h}, Y_{g,h} \mid u = +(e \twoheadrightarrow (g \to h))\} \end{aligned}$$
- $$\begin{aligned} att^m = att \setminus \{&(g,h) \mid u = +(e \twoheadrightarrow (g \to h))\} \cup \\ \{&(g, X_{g,h}), (X_{g,h}, Y_{g,h}), (Y_{g,h}, h) \mid u = +(e \twoheadrightarrow (g \to h))\} \cup \\ \{&(b, Z_{a,b}), (Z_{a,b}, a) \mid (a,b) \in sup\} \cup \\ \{&(e, Z_{a,b}) \mid (e,(a,b)) \in s\text{-}att, (a,b) \in sup\} \cup \\ \{&(c, X_{c,d}), (X_{c,d}, Y_{c,d}), (Y_{c,d}, d), (e, Y_{c,d}) \mid \\ &(e,(c,d)) \in s\text{-}att, (c,d) \in att\} \cup \\ \{&(f, Z_{e,f}) \mid u = +(e \Rightarrow f)\}. \end{aligned}$$

Besides the meta-arguments $Z_{a,b}$ of Definition 4.2, and the attacks involving those arguments, the above meta argumentation framework contains meta-arguments $X_{c,d}, Y_{c,d}$ for encoding second order attacks in $s\text{-}att$ toward attacks $(c,d) \in att$. In fact, an attack $e \twoheadrightarrow (a \Rightarrow b)$ in $s\text{-}att$ toward a support is encoded as an attack from e toward $Z_{a,b}$ in the meta argumentation framework, while $e \twoheadrightarrow (c \to d)$ in $s\text{-}att$ is encoded as an attack from e toward $Y_{c,d}$ in the meta argumentation framework (which contains also the attacks $(c, X_{c,d}), (X_{c,d}, Y_{c,d}), (Y_{c,d}, d)$). Moreover, meta-arguments $Z_{e,f}$ and $X_{g,h}, Y_{g,h}$, are added to the meta argumentation framework for encoding,

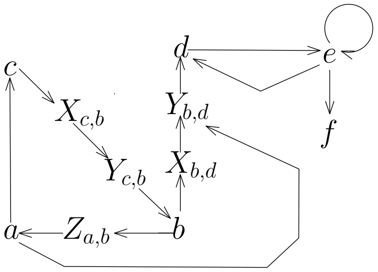


Figure 8: Compact argumentation framework for the extended argumentation framework EF_0 of Figure 6 w.r.t. the update $u = +(d \twoheadrightarrow (c \to b))$.

respectively, the addition of a second order attack toward a support $(e, f) \in sup$ or toward an attack $(g, h) \in att$. In the latter case, meta-arguments $X_{g,h}$ and $Y_{g,h}$ along with the set of attacks $\{(g, X_{g,h}), (X_{g,h}, Y_{g,h}), (Y_{g,h}, h)\}$ are used to simulate the attack $g \to h$ which is attacked by e in the extended argumentation framework. This enables the definition of simple attack updates to simulate second-order attack updates.

Example 4.5. *The compact argumentation framework for the EAF EF_0 of Figure 6 w.r.t. the update $u = +(d \twoheadrightarrow (c \to b))$ is shown in Figure 8. Herein, the attacks involving the meta-arguments $X_{b,d}$ and $Y_{b,d}$ allow us to simulate the second order attack $a \twoheadrightarrow (b \to d)$. Moreover, the attacks involving the meta-arguments $X_{c,b}$ and $Y_{c,b}$ are added to enable the simulation of the second-order update u by a single attack update on the meta argumentation framework.*

We now define updates on the meta argumentation framework.

Definition 4.6 (Updates for the meta argumentation framework [7]). *Let $EF = \langle Ar, att, sup, s\text{-}att \rangle$ be an extended argumentation framework, and u an update for EF. The corresponding update u^m for the compact argumentation framework $CF(EF, u)$ is as follows:*

$u^m =$	$u^m =$
$+(Z_{e,f} \to e)$ *if* $u = +(e \Rightarrow f)$	$-(Z_{e,f} \to e)$ *if* $u = -(e \Rightarrow f))$
$+(c \to d)$ *if* $u = +(c \to d)$	$-(c \to d))$ *if* $u = -(c \to d))$
$+(e \to Y_{g,h})$ *if* $u = +(e \twoheadrightarrow (g \to h))$	$-(e \to Y_{g,h})$ *if* $u = -(e \twoheadrightarrow (g \to h))$
$+(e \to Z_{a,b})$ *if* $u = +(e \twoheadrightarrow (a \Rightarrow b))$	$-(e \to Z_{a,b})$ *if* $u = -(e \twoheadrightarrow (a \Rightarrow b))$

For instance, continuing with Example 4.5, given the extended argumentation framework EF_0 of Figure 6 and the update $u = +(d \twoheadrightarrow (c \to b))$, we have that update u^m for the compact argumentation framework $CF(EF_0, u)$ shown in Figure 8 is $u^m = +(d \to Y_{c,b})$.

Finally, given an initial extension for an extended argumentation framework and an update, we define the initial labelling for the corresponding compact argumentation framework as follows.

Definition 4.7 (Corresponding initial labelling [7])**.** *Given an extended argumentation framework $EF_0 = \langle Ar, att, sup,\ s\text{-}att\rangle$ and a initial labelling $\mathcal{L}_0$, the corresponding initial labelling $\mathcal{L}_0^m$ for the compact argumentation framework $CF(EF_0, u) = \langle Ar^m, att^m\rangle$ is as follows:*

$\forall a \in Ar \cap Ar^m$:	$\mathcal{L}_0^m(a) = \mathcal{L}_0(a)$;	
$\forall\ X_{a,b} \in Ar^m$:	$\mathcal{L}_0^m(X_{a,b}) = \texttt{in}$	*if* $\mathcal{L}_0(a) = \texttt{out}$
	$\mathcal{L}_0^m(X_{a,b}) = \texttt{out}$	*if* $\mathcal{L}_0(a) = \texttt{in}$
	$\mathcal{L}_0^m(X_{a,b}) = \texttt{undec}$	*if* $\mathcal{L}_0(a) = \texttt{undec}$
$\forall\ Y_{a,b} \in Ar^m$:	$\mathcal{L}_0^m(Y_{a,b}) = \texttt{in}$	*if (i)* $\mathcal{L}_0^m(X_{a,b}) = \texttt{out}$ *and* *(ii)* $\forall c \in Ar$ *s.t.* $(c, (a, b)) \in s\text{-}att$, $\mathcal{L}_0(c) = \texttt{out}$
	$\mathcal{L}_0^m(Y_{a,b}) = \texttt{out}$	*if (i)* $\mathcal{L}_0^m(X_{a,b}) = \texttt{in}$ *or* *(ii)* $\exists\ c \in Ar \mid (c, (a, b)) \in s\text{-}att$ *and* $\mathcal{L}_0(c) = \texttt{in}$
	$\mathcal{L}_0^m(Y_{a,b}) = \texttt{undec}$,	*otherwise.*

$\forall Z_{a,b} \in Ar^m$:	$\mathcal{L}_0^m(Z_{a,b}) = \texttt{in}$	*if (i)* $\mathcal{L}_0(b) = \texttt{out}$ *and* *(ii)* $\forall\ c \in Ar$ *s.t.* $(c, (a, b)) \in s\text{-}att$, $\mathcal{L}_0(c) = \texttt{out}$
	$\mathcal{L}_0^m(Z_{a,b}) = \texttt{out}$	*if (i)* $\mathcal{L}_0(b) = \texttt{in}$ *or* *(ii)* $\exists\ c \in Ar \mid (c, (a, b)) \in s\text{-}att$ *and* $\mathcal{L}_0(c) = \texttt{in}$
	$\mathcal{L}_0^m(Z_{a,b}) = \texttt{undec}$,	*otherwise.*

For instance, given the initial preferred extension $E_0 = \{a, b, d, f\}$ of the extended argumentation framework EF_0 of Example 4.1, the initial labelling for the compact argumentation framework $CF(EF_0,\ +(d \to Y_{c,b}))$ of Figure 8 is such that $\mathcal{L}_0^m(a) = \mathcal{L}_0(a) = \texttt{in}$, $\mathcal{L}_0^m(c) = \mathcal{L}_0(c) = \texttt{out}$, $\mathcal{L}_0^m(X_{c,b}) = \texttt{in}$, and $\mathcal{L}_0^m(Y_{c,b}) = \texttt{out}$. Also, we have that $\mathcal{L}_0^m(b) = \mathcal{L}_0(b) = \texttt{in}$, $\mathcal{L}_0^m(X_{b,d}) = \texttt{out}$, $\mathcal{L}_0^m(Y_{b,d}) = \texttt{out}$ since $\mathcal{L}_0^m(a) = L_0(a) = \texttt{in}$, and $\mathcal{L}_0^m(d) = \mathcal{L}_0(d) = \texttt{in}$.

Algorithm 2 Incr-EAF(EF_0, u, E_0, σ, Solver_σ)

```
Input: Extended argumentation framework $EF_0 = \langle Ar_0, att_0, sup_0, satt_0 \rangle$,
    update $u$ over $EF_0$,
    an initial $\sigma$-extension $E_0$ for $EF_0$,
    semantics $\sigma \in \{\mathcal{PR}, \mathcal{ST}\}$,
    function $\mathsf{Solver}_\sigma(AF)$ that returns an $\sigma$-extension of $AF$ if it exists, and $\bot$
    otherwise;
Output: An $\sigma$-extension $E$ for $u(EF_0)$ if it exists, $\bot$ otherwise;
 1: if $checkProp(EF_0, u, E_0, \sigma)$ then
 2:         return $E_0$;
 3: end if
 4: Let $CF_0 = CF(EF_0, u)$ be the compact argumentation framework for $EF_0$ w.r.t.
    $u$ (cf. Definition 4.4);
 5: Let $u^m$ be the update for $CF_0$ corresponding to $u$ (cf. Definition 4.6);
 6: Let $E_0^m$ be the initial $\sigma$-extension for $CF_0$ corresponding to $E_0$;
 7: Let $E^m = \mathsf{Incr\text{-}Alg}(CF_0, u^m, \sigma, E_0^m, \mathsf{Solver}_\sigma)$;
 8: if $(E^m \neq \bot)$ then
 9:         return $E = (E^m \cap Ar_0)$;
10: else
11:         return $\bot$;
12: end if
```

Incremental Algorithm for Extended Argumentation Frameworks We are now ready to present the algorithm for extending the incremental technique to the case of extended argumentation frameworks. Given an extended argumentation framework EF_0, a semantics $\sigma \in \{\mathcal{PR}, \mathcal{ST}\}$, an extension $E_0 \in \mathcal{E}_\sigma(EF_0)$, and an update u of the form $u = \pm(a \Rightarrow b)$, $u = \pm(a \rightarrow b)$, $u = \pm(e \twoheadrightarrow (c \Rightarrow d))$, or $u = \pm(e \twoheadrightarrow (c \rightarrow d))$, Algorithm 2 computes an extension E of the updated extended argumentation framework $u(EF_0)$, if it exists [7]. The algorithm works as follows. It first checks if the initial extension E_0 is still an extension of the updated extended argumentation framework at Line 1, where $checkProp(EF_0, u, E_0, \sigma)$ is a function returning *true* iff the update is irrelevant—the interested reader can find the conditions under which an update for an extended argumentation framework is irrelevant in [7]. If this is the case, it immediately returns the initial extension. Otherwise, it computes the compact argumentation framework CF_0 (Line 4), the update u^m for CF_0 (Line 5), and the initial σ-extension E_0^m for CF_0 (Line 6). Next, it invokes function $\mathsf{Incr\text{-}Alg}$ (*i.e.,* Algorithm 1). $\mathsf{Incr\text{-}Alg}$ takes as input the parameters CF_0, u^m, σ, E_0^m, and Solver_σ, where Solver_σ is an external solver that can compute an σ-extension for the input argumentation framework. Finally, the extension of the updated extended argumentation framework (if any) is obtained by projecting out the extension E^m returned by Incr-Alg over the set of arguments Ar_0 of the initial extended argumentation framework (Line 9).

From a computational point of view, in the worst case (that is, when every argument is influenced, and thus the RAF collapses to be the updated framework), Algorithm 1 and Algorithm 2 have the same computational complexity as the corresponding task in the static setting under the considered AF semantics. It is worth noting that the overhead of computing the influenced set and the RAF is polynomial in the input framework's size.

The use of the incremental techniques discussed in this section and the previous one become significant in practice. In fact, in [5] it is shown that Algorithm 1 outperforms state-of-the-art solvers that compute the extensions from scratch for single updates by two orders of magnitude on average, and it remains faster than the competitors even when recomputing an extension after performing updates simultaneously. Moreover, [7] reports on an experimental analysis showing that Algorithm 2 also outperforms by two orders of magnitude the computation from scratch on EAFs, where solvers from scratch taking as input the (compact) Dung argumentation frameworks resulting from the transformation of the candidate EAF (cf. Definition 4.4) are used. Finally, the experimental results concerning the use of both Algorithm 1 and Algorithm 2 also revealed that the improvements of using incremental techniques become larger as the computation from scratch becomes more challenging.

5 Incremental Computation in Defeasible Logic Programming

In [30], four frameworks that consider the structure of arguments were presented. Two of them—ASPIC$^+$ [85] and ABA [94]—build the set of all possible arguments from the knowledge base and then rely on using one of the possible Dung semantics to decide on the acceptance of arguments. The other two—Logic-Based Deductive Argumentation [32] and DeLP [70]—only build the arguments involved in answering the query. These last two frameworks exhibit several differences [30]—among them is the base logic used as a knowledge representation language: [32] relies on propositional logic, requiring a theorem prover to solve queries; on the other hand, DeLP [70] adopts an extension of logic programming, which is a computational framework per se. To better understand the differences among the frameworks mentioned above, we refer the interested reader to [68], where a variant of DeLP using the grounded semantics is also discussed.

A fundamental distinction between DeLP and the other three frameworks, which significantly affects a query's resolution, rests on how attacks between arguments are described. DeLP considers two forms of defeat: *proper* and *blocking*; the former is akin to Dung's attack [55], whereas the latter presents a different behavior since the two arguments that are part of the blocking defeat relation, attacker and attackee, are defeated (hence the use of the term *blocking defeater*). Of course, this could be modeled in Dung's graphs as a mutual attack, but the DeLP mechanism forbids, in a properly formed dialogue, the use of two blocking defeaters successively because the introduction of another blocking defeater is unnecessary since the first two are already defeated. Moreover, to find the answers required by the query, other considerations of dialogical nature are taken into account, strengthening the reasoning process by forbidding common dialogical fallacies; these characteristics have been reflected in the development of a game-based semantics [95].

In this section, we focus on the incremental computation in the context of structured argumentation. Particularly, we discuss an incremental technique [12; 11] for *Defeasible Logic Programming (DeLP)* [69; 70] which shares the same underlying ideas and the goal of avoiding wasted effort as in the (incremental) technique previously discussed for AFs. Given that our primary focus is on the changes in the structure of the arguments used to answer a query, we have considered the DeLP language; however, the ideas here developed can inspire similar techniques for other structured argumentation frameworks such as ABA and ASPIC$^+$. Next, we will summarize the necessary elements to develop the updating techniques in DeLP's structured argumentation; see [11] for an extended presentation.

5.1 Defeasible Logic Programming and Updates

A DeLP program $\mathcal{P} = (\Pi, \Delta)$ consists of sets Π and Δ of *strict* and *defeasible* rules defined using elements of a set *Lit* of literals, that are ground atoms obtained from a set **At** of atoms. $Lit_{\mathcal{P}}$ denotes the set of literals occurring in a rule of $\mathcal{P}$, and the symbol "$\sim$" represents strong negation; for any literal $\alpha \in Lit$ the formula $\sim\sim\alpha$ is considered equivalent to α and can be used for denoting it. Particularly, given the literals $\alpha_0, \alpha_1, \dots, \alpha_n$, a strict rule $\alpha_0 \leftarrow \alpha_1, \dots, \alpha_n$ (with $n \geq 0$) represents non-defeasible information, while *defeasible* rules $\alpha_0 \multimap \alpha_1, \dots, \alpha_n$ (with $n > 0$) represent tentative information, *i.e.,* information that can be used if nothing can be posed against it. Given a strict or defeasible rule r, we use *head*(r) to denote α_0, and *body*(r) to denote the set of literals $\{\alpha_1, \dots, \alpha_n\}$. Strict rules with empty body will also be called *facts*.[1]

[1] With a little abuse of notation, in the following we will denote a fact ($\alpha \leftarrow$) simply by α.

As an example of DeLP program, let us consider $\mathcal{P}_1 = (\Pi_1, \Delta_1)$, where:

Π_1=$\{x, y,\ z, (w \leftarrow y)\}$ is the set of strict rules (and facts), and

$\Delta_1 = \{(a \prec w), (a \prec z), ({\sim}a \prec z), (b \prec a), (b \prec z), (c \prec b, x),$

$({\sim}c \prec b), (d \prec\ {\sim}c)\}$ is the set of defeasible rules.

Given a DeLP program $\mathcal{P} = (\Pi, \Delta)$ and a literal $\alpha \in Lit_{\mathcal{P}}$, a *(defeasible) derivation* for α w.r.t. $\mathcal{P}$ is a finite sequence $\alpha_1, \alpha_2, \dots, \alpha_n = \alpha$ of literals such that (i) each literal α_i is in the sequence because there exists a (strict or defeasible) rule $r \in \mathcal{P}$ with head α_i and body $\alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_k}$ such that $i_j < i$ for all $j \in [1, k]$, and (ii) there do not exist two literals α_i and α_j such that $\alpha_j = {\sim}\alpha_i$. A derivation is said to be a *strict derivation* if only strict rules are used.

A program $\mathcal{P}$ is *contradictory* if and only if there exist defeasible derivations for at least two complementary literals α and ${\sim}\alpha$ from $\mathcal{P}$. We assume that Π (the strict part of $\mathcal{P}$) is not contradictory. However, complementary literals can be derived from $\mathcal{P}$ when defeasible rules are used in the derivation. Two literals α and β are said to be *contradictory* if (i) neither $\Pi \cup \{\alpha\}$ nor $\Pi \cup \{\beta\}$ strictly derive a pair of complementary literals, whereas (ii) $\Pi \cup \{\alpha, \beta\}$ does. Pairs of complementary literals are clearly contradictory; a set of literals is said to be contradictory if it contains two contradictory literals.

Considering the program $\mathcal{P}_1$, the literal c can be derived using the following sets of rules and facts: $\{(c \prec x, b), (x), (b \prec a),\ (a \prec w), (w \leftarrow y), (y)\}$; the derivation $(y, w, a, b, x,$
$c)$ describes how rules can be applied to derive c. However, the set of rules $\Pi_1 \cup \Delta_1$ is contradictory since also ${\sim}c$ can be derived using the rules: $\{({\sim}c \prec b), (b \prec a),$ $(a \prec w), (w \leftarrow y), (y)\}$. The non-contradictory set of literals that can be derived from Π_1 is $\{x,\ y,\ w,\ z\}$.

DeLP incorporates a defeasible argumentation formalism for the treatment of contradictory knowledge, allowing the identification of conflicting pieces of knowledge, and a *dialectical process* is used for deciding which information prevails as warranted. This process involves the construction and evaluation of arguments that either support or interfere with a user-issued query. An *argument* $\mathcal{A}$ for a literal α is a couple $\langle \mathcal{A}, \alpha \rangle$ where $\mathcal{A}$ is a set of defeasible rules representing a derivation that is (i) supported by facts, (ii) non-contradictory, and (iii) $\subseteq$-minimal (*i.e.,* there is no proper subset of $\mathcal{A}$ satisfying both (i) and (ii)). As an example, $\langle \mathcal{A}_1, c \rangle = \langle \{(c \prec x, b), (b \prec a),\ (a \prec w), (w \leftarrow y)\}, c \rangle$ and $\langle \mathcal{A}_2, {\sim}a \rangle = \langle \{({\sim}a \prec z)\}, {\sim}a \rangle$ are two arguments that can be obtained from the program $\mathcal{P}_1$. An argument $\langle \mathcal{A}, \alpha \rangle$ is said to be a *sub-argument* of $\langle \mathcal{A}', \alpha' \rangle$ if $\mathcal{A} \subseteq \mathcal{A}'$.

The main task of DeLP is establishing *warranted* literals. A literal α is said to be warranted if there exists an undefeated argument $\langle \mathcal{A}, \alpha \rangle$. To determine if an argument $\langle \mathcal{A}, \alpha \rangle$ is undefeated, *defeaters* for $\langle \mathcal{A}, \alpha \rangle$ are considered, and since reinstatement could happen when all of $\mathcal{A}$'s possible defeaters are defeated, the process continues considering defeaters for $\mathcal{A}$'s defeaters, and so on. To define defeaters, to decide when an attack is successful, we require a comparison criterion $\succ$ over arguments, which is irreflexive and asymmetric. As the comparison criterion is a modular part of the argumentation inference engine, we will abstract away from this criterion and simply assume the existence of a comparison criterion $\succ$ between arguments: $\langle \mathcal{A}, \alpha \rangle \succ \langle \mathcal{B}, \beta \rangle$ meaning that argument $\langle \mathcal{A}, \alpha \rangle$ is preferred to $\langle \mathcal{B}, \beta \rangle$. Intuitively, an argument $\langle \mathcal{A}, \alpha \rangle$ *attacks* an argument $\langle \mathcal{B}, \beta \rangle$ when there is a sub-argument $\langle \mathcal{C}, \gamma \rangle$ of $\langle \mathcal{B}, \beta \rangle$, such that α and γ are contradictory. When the attacker satisfies that $\langle \mathcal{C}, \gamma \rangle$ is not preferred to $\langle \mathcal{A}, \alpha \rangle$ (*i.e.,* $\langle \mathcal{C}, \gamma \rangle \not\succ \langle \mathcal{A}, \alpha \rangle$), the attacker is called a *defeater*. A defeater $\langle \mathcal{A}, \alpha \rangle$ for $\langle \mathcal{B}, \beta \rangle$ will be referred to as a *proper defeater* if $\langle \mathcal{A}, \alpha \rangle \succ \langle \mathcal{C}, \gamma \rangle$; otherwise, it will be called a *blocking defeater*.

An other part of the dialectical process is the construction of the so called *dialectical tree*, which is used to decide the warrant status of a literal. A dialectical tree contains all the possible acceptable argumentation lines (namely, sequences of defeating arguments) that can be constructed from the given argument that sits on the root of that tree as paths from the root to the leaves. (see [52] for a discussion). More in detail, a dialectical tree for an argument $\langle \mathcal{A}, \alpha \rangle$ is a tree-like structure where nodes are arguments and the root node is $\langle \mathcal{A}, \alpha \rangle$. Each root-to-leaf path in the tree is an *acceptable argumentation line*, which is a finite sequence of arguments that satisfy the following four constraints: (i) every argument of the sequence defeats its predecessor; (ii) the arguments in odd (resp., even) positions of the sequence does not contradict the strict part of the program; (iii) two blocking defeaters cannot appear one immediately after the other in the sequence; and (iv) arguments cannot appear twice in the sequence (also when appearing as sub-arguments).

Therefore, it is interesting to note that a dialectical tree for an argument represents the exhaustive dialectical analysis for that argument. Each dialectical tree is then marked to obtain the status of the literal α in the argument at its root through a bottom-up marking procedure, consisting in i) marking all leaves of the tree as UNDEFEATED; then, ii) every non-leaf node is marked as DEFEATED if and only if at least one of its children is marked as UNDEFEATED, otherwise it is marked as UNDEFEATED. Thus, if there exists a marked dialectical tree whose root contains an argument for α, which is marked as UNDEFEATED, we will say that α is *warranted*[2].

[2] The system available at the following link allows us to compare the abstract semantics with that of DeLP: `https://hosting.cs.uns.edu.ar/~daqap/client/index.html`; see [78] for a description.

Considering the program $\mathcal{P}_1$, only x, y, z, w, and b are warranted.

Given a DeLP program $\mathcal{P}$, we define a total function $\mathcal{S}_{\mathcal{P}} : \mathit{Lit} \rightarrow \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$ assigning a *status* to each literal w.r.t. $\mathcal{P}$ as follows: $\mathcal{S}_{\mathcal{P}}(\alpha) = \mathtt{in}$ if α is warranted; $\mathcal{S}_{\mathcal{P}}(\alpha) = \mathtt{out}$ if $\mathcal{S}_{\mathcal{P}}(\sim\!\alpha) = \mathtt{in}$; $\mathcal{S}_{\mathcal{P}}(\alpha) = \mathtt{undec}$ if neither $\mathcal{S}_{\mathcal{P}}(\alpha) = \mathtt{in}$ nor $\mathcal{S}_{\mathcal{P}}(\alpha) = \mathtt{out}$. For literals not occurring in the program we also say that their status is unknown.

Updates. An *update* for a DeLP program $\mathcal{P} = \langle \Pi, \Delta \rangle$ modifies $\mathcal{P}$ into a new program $\mathcal{P}' = \langle \Pi', \Delta' \rangle$ by adding or removing a strict or a defeasible rule r. In particular, we allow the removal of any rule r of $\mathcal{P}$ through an update, and consider the addition of a rule r such that $\mathit{body}(r) \subseteq \mathit{Lit}_{\mathcal{P}}$ and $\mathit{head}(r) \subseteq \mathit{Lit}$, thus allowing also the addition of a rule whose head is a literal not belonging to $\mathit{Lit}_{\mathcal{P}}$. Given a DeLP program $\mathcal{P}$ and a strict or defeasible rule r, we use $u = +r$ (resp., $u = -r$) to denote a rule addition (resp., deletion) update to be performed on $\mathcal{P}$, obtaining the DeLP-program $u(\mathcal{P})$ resulting from the application of update u to $\mathcal{P}$. In the following, we assume that any update u is feasible, meaning that *i*) we only remove (resp. add) strict or defeasible rules appearing (resp., not appearing) in the given program $\mathcal{P}$, and *ii*) guaranteeing that the strict part of the updated program $u(\mathcal{P})$ will not be contradictory.

5.2 Incremental Computation of Warranted Literals

We first introduce the concept of labeled directed hypergraph associated with a DeLP program, which is central to our incremental approach.

Given a program $\mathcal{P}$, the corresponding labelled hypergraph $G(\mathcal{P}) = \langle N, H \rangle$ consists of a set N of nodes and a set H of labelled hyper-edges (Src, t, l), where Src is a possibly empty set called the *source set*, t is called the *target node*, and $l \in \{\mathtt{def}, \mathtt{str}, \mathtt{cfl}\}$ is a label associated to the hyper-edge. Literals for which there exists a strict derivation in Π are immediately added to the set N of nodes of $G(\mathcal{P})$. Then, for each (strict or defeasible) rule whose body is in N, the head is added to N, and a (str or def) labelled hyper-edge corresponding to the (strict or defeasible) rule is added to the set H of hyper-edges. Finally, there is a pair of (cfl) labelled hyper-edges for each pair of complementary literals appearing as nodes in the hypergraph.

The hypergraph $G(\mathcal{P}_1)$ for the DeLP program $\mathcal{P}_1$ is shown in Figure 9(a) where $\leftrightarrow$ (resp. $\lhd\!\!-$ and $\blacktriangleleft\!\!-$) denotes hyper-edges labeled as cfl (resp. def and str).

We say that there is a path from a literal β to a literal α, if either (i) there exists a hyper-edge whose source set contains β and whose target is α, or (ii) there exists a literal γ and also there exist paths from β to γ and from γ to α. Moreover, we say

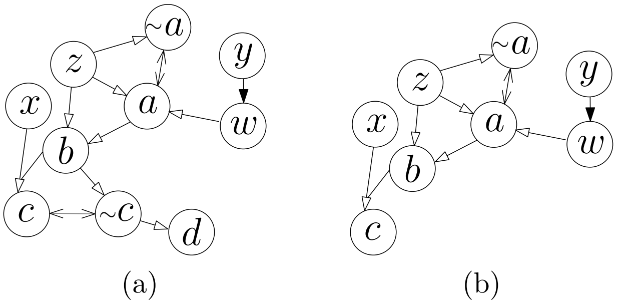


Figure 9: (a) $G(\mathcal{P}_1)$ for program $\mathcal{P}_1$; (b) $G(\mathcal{P}_1')$ for program $\mathcal{P}_1' = -(\sim c \multimap b)(\mathcal{P}_1)$.

that a node y is *reachable* from a set X of nodes if there exists a path from some x in X to y.

Given an update u, we denote with $G(u, \mathcal{P})$ the labeled hypergraph $G(u^+(\mathcal{P}))$ or $G(u^-(\mathcal{P}))$, depending on whether u consists of an insertion or deletion, respectively. The reason of this difference is that, to determine the set of literals whose status may change by deleting a rule r, we need to consider the hypergraph also containing the hyper-edge derived from r.

Given a DeLP-program $\mathcal{P}$ and an update u, our incremental approach for recomputing the status of the literals after performing u consists of the following steps.

- Firstly, it is checked whether the update u is *irrelevant*, that is all literals in *Lit* are preserved. In such a case the initial status $\mathcal{S}_\mathcal{P}$ is returned.
- If u is not irrelevant, we need to:
  - (i) compute the set of literals that are “influenced” by the update;
  - (ii) among the influenced literals determine the subset of literals (called *core literals*) whose status may change after performing the update. The status of uninfluenced literals does not change after the update.
  - (iii) compute the updated status of the core literals; and
  - (iv) determine the updated status of the *inferable* literals, *i.e.,* the literals whose status can be immediately determined from the status of the core literals.

The identification of relevant and irrelevant updates, as well influenced, preserved, core, and inferable literals is discussed below. In [12; 11], it is shown that, in practice, the algorithm resulting from applying the above-mentioned steps turns out to be much more efficient than recomputing everything from scratch.

Irrelevant updates. Sufficient conditions guaranteeing that the status of each literal in the updated program is the same as that of the initial program are investigated in [11]. In these cases we say that the update u is *irrelevant*. One of these conditions holds whenever we add (resp. remove) a defeasible rule whose head's status is `in` (resp. `out`) w.r.t. the initial program. However, this does not hold for updates concerning strict rules. In these cases, we need to makes use of the hypergraph associated with a DeLP program, as well as the status of the literals *related* to an update.

A literal is said to be related to a given update $u = \pm r$ and program $\mathcal{P}$ if it can be reached from $head(r)$ in the labelled hypergraph $G(u, \mathcal{P})$ by navigating forward each rules and backward strict rules only, until no new related literals can be found. We call *deductive closure* of facts and strict rules of a program $\mathcal{P}$ the set of literals that are facts in $\mathcal{P}$ or can be derived from the strict part Π of $\mathcal{P}$. Given this, an update $u = \pm r$ is irrelevant if either (i) $head(r)$ does not belong to $G(u, \mathcal{P})$; or (ii) either $head(r)$ or $\sim head(r)$ appears in the deductive closure of facts and strict rules of both programs $\mathcal{P}$ and $u(\mathcal{P})$; or (iii) at least one literal in the body of r is either `out` or not related to u. Recomputing the status of the updated program's literals can be avoided if an irrelevant update is performed.

Relevant updates and influenced set. We now consider the computation of the status of literals for updates which have not been identified as irrelevant. An update is *relevant* whenever it causes the status of at least one literal to change. That is, even if for relevant updates the status of some literals may not change, and therefore for those literals, their status does not need to be recomputed when the update is performed. To avoid wasted effort, we determine the subset of literals whose status needs to be recomputed after an update. Towards this end, we discuss the concept of *influenced set*, which consists of the set of literals that are related to a given update u and program $\mathcal{P}$ but using only labeled hyper-edges whose corresponding rules are such that (i) the head (or its complement) is not in the deductive closures of both $\mathcal{P}$ and $u(\mathcal{P})$, and (ii) the body does not contain a literal that is not related to u and $\mathcal{P}$ and such that its status is `out`—intuitively, the other hyper-edges can be ignored as they correspond to rules whose head does not change status. For instance, for the program $\mathcal{P}_1$ and update $u = -(\sim c \prec b)$, the influenced set consists only of the literals b, c, $\sim c$, and d.

The notion of influenced set for DeLP programs is conceptually similar to the influenced set of Definition 3.3 for abstract argumentation frameworks (where arguments have no internal structure). Although the aim is analogous, here we deal with incremental computation of the status of *structured* arguments, and consider a notion of influenced set w.r.t. an update for a DeLP program and its status that we then apply to (hyper)graphs representing DeLP programs, from which structured arguments are derived. A significant difference is that here we have both strict and defeasible rules meaning that to determine a portion of the hypergraph that contains nodes corresponding to literals whose status may change, we need to navigate strict edges both forward and backward. As an example, consider the DeLP program $\mathcal{P}_\chi = \langle \Pi_\chi, \Delta_\chi \rangle$ where $\Pi_\chi = \{f_1,\ f_2,\ a \leftarrow b,\ \sim a \leftarrow c\}$ and $\Delta_\chi = \{b \mathrel{-\!\!\prec} f_1\}$, and let $u = +(c \mathrel{-\!\!\prec} f_2)$ be an update yielding the updated DeLP program $\mathcal{P}'_\chi$. The influenced set is $\{c,\ \sim a,\ a, b\}$. Observe that b is included in the influenced set by navigating backward via the (hyper)edge corresponding to the strict rule $a \leftarrow b$, while the other literals are reached by forward reachability. Note that including b is important as its status changes (it is `undec` w.r.t. $\mathcal{P}'_\chi$, it was `in` w.r.t. $\mathcal{P}_\chi$).

Preserved, core, and inferable literals. Using the influenced set we can identify the preserved literals, *i.e.,* the literals whose status does not change after performing a relevant update. This set consists of the literals (*i.e.,* nodes) of the updated hypergraph that are not influenced. The status of a literal for which there is no argument in the (updated) program may depend only on the status of its complementary literal—we call such literals *inferable* and use them to define what we call *core literals.* The core literals for an update u are those in $Lit_{\mathcal{P}'}$ that are influenced but are not inferable, where $\mathcal{P}'$ is the updated program. The status of an inferred literal w.r.t. the updated program can be either `out` or `undec`, and if it is `out` it is entailed by the status of a core or preserved literal that is `in`. Finally, the status of the literals not in $Lit_{\mathcal{P}'}$ can be readily determined to be `undec`.

Considering the program $\mathcal{P}_1$ and update $u = -({\sim}c \mathrel{-\!\!\prec} b)$, ${\sim}c$ and d are the only inferable literals, while b and c are core literals. The (updated) status of the inferable literal ${\sim}c$ is `out` as it is entailed by the (updated) status of its complementary literal, which is `in`; the status of the inferable literal d remains `undec`.

Efficiency. The incremental technique discussed in this section has been the subject of analysis in [11; 12], which report on a set of experiments comparing the incremental approach with full recomputation from scratch (that is, the direct computation of the status of all the literals in an updated DeLP program using the DeLP-Solver). It turned out that the incremental approach significantly outperforms computation from scratch. Specifically, the incremental algorithm takes only a few seconds for DeLP programs, while the approach from scratch takes almost 2 minutes.

6 Related Work

Overviews of key concepts in argumentation theory and formal models of argumentation in the field of Artificial Intelligence are presented in [29; 31; 91; 19]. Further discussion regarding uses of computational argumentation as an Agreement Technology can be found in [86].

A comprehensive introduction to the semantics of static abstract argumentation frameworks can be found in [21]. Although the idea underlying abstract argumentation frameworks is intuitive and straightforward, most of the semantics proposed so far suffer from a high computational complexity [58; 57; 59; 60; 64; 65; 66; 67]. Complexity bounds and evaluation algorithms for abstract argumentation frameworks have been intensely studied in the literature, but most of this research focused on static frameworks, whereas, in practice, argumentation frameworks are dynamic systems [42; 62; 25; 81; 24; 51]. In fact, in general, an AF represents a temporary situation, and new arguments and attacks can be added/retracted to model new available knowledge. For instance, for disputes among users of online social networks [76], arguments/attacks are continuously added or removed by users to express their point of view in response to the last move made by the adversaries (often disclosing as few arguments/attacks as possible).

There have been several significant efforts aimed at coping with the dynamic aspects of abstract argumentation. In [39; 40], the authors have investigated the principles according to which a grounded extension of a Dung abstract argumentation framework does not change when the set of arguments/attacks is changed. Meanwhile, in [35] a synthesis is presented concerning the characterization of changes based on the work presented in [44; 45; 33; 34] where the evolution of the set of extensions after performing a change operation is studied; here, a change operation can be about adding or removing one interaction or adding or removing one argument and a set of interactions.

Dynamic argumentation has been applied to the decision-making mechanisms of an autonomous agent by [18], where it is studied how the acceptability of arguments evolves when a new argument is added to the decision system. Other relevant works on dynamic aspects of Dung's argumentation frameworks include the following. [25] has proposed an approach exploiting the concept of the splitting of logic programs to deal with dynamic argumentation. The technique considers weak expansions of the initial argumentation framework, where added arguments never attack previous ones. [28] have investigated whether and how it is possible to modify a given argumentation framework so that a desired set of arguments becomes an extension, whereas [90] have studied equivalence between two argumentation frameworks when further information (another argumentation framework) is added to both argumentation frameworks. [26] has focused on expansions where new arguments and attacks may be added, but the attacks among the old arguments remain unchanged, while [27] have characterized update and deletion equivalence, where adding/deleting arguments/attacks is allowed (deletions were not considered by [90; 26]).

Several approaches for dividing argumentation frameworks into subgraphs have been explored in the context of dynamic argumentation frameworks. The division-based method, proposed in [81] and then refined in [24], divides the updated framework into two parts: *affected* and *unaffected*, where only the status of affected arguments is recomputed after updates. Using the results introduced in [81], the work presented in [80] investigated the efficient evaluation of the justification status of a subset of arguments in an argumentation framework (instead of the whole set of arguments), and proposed an approach based on answer-set programming for *local* computation. In [79], an argumentation framework is decomposed into a set of strongly connected components, yielding sub-argumentation frameworks located in layers, which are then used for incrementally computing the semantics of the given argumentation framework by proceeding layer by layer. Then, [97] introduced a matrix representation of argumentation frameworks and proposed a matrix reduction that, when applied to dynamic argumentation frameworks, resembles the division-based method in [81].

Changes in bipolar argumentation frameworks have been studied in the work [49], where it is shown how the addition of one argument together with one support that involves that argument (and without introducing any attack) impacts the extensions of the updated bipolar argumentation framework. However, these works do not address the incremental computation in dynamic bipolar argumentation frameworks, nor in extended argumentation frameworks modeling attacks towards the attack relation [82; 22] and defeasible support [38].

There have been fewer attempts to consider the dynamics of the defeasible argumentation in the field of structured argumentation [30]. As in the abstract argumentation case, there have been some works following the belief revision approach. In [63], the issue of modifying strict rules to become defeasible was analyzed in the context of revisions effected over a knowledge base, while in [87] the authors thoroughly explored the different cases that may occur when a DeLP program is modified by adding, deleting, or changing its elements. Neither of these works explored the implementation issues related to the problems studied here. Regarding implementations of approaches focusing on improving the tractability of determining the status of pieces of knowledge, in [42; 43], the authors consider several alternatives to avoid recomputing warrants. In [54], the authors focus on challenges arising in the development of recommender systems, addressing them via the design of novel architectures that improve the computation of answers. Finally, [73] makes use of heuristics designed to improve efficiency, and [92] deals with the computational complexity of performing recalculations in a structured argumentation setting by relying on an approximation algorithm.

We believe that the set of ideas proposed in this work may be a forerunner of similar techniques for the optimization of other structured argumentation frameworks such as, for example, ABA and ASPIC$^+$. Regarding ABA, the construction of deductions is very similar to that of arguments for DeLP, although the way arguments attack each other is different. Therefore, similarly, the ABA framework could be represented using hypergraphs (where assumptions may be modeled as defeasible facts) to identify irrelevant updates and restrict the hypergraph to compute the semantics of updated programs efficiently. The similarities between DeLP and ASPIC$^+$ are even more substantial: both have a distinction between strict and defeasible inference rules, and both use comparison criteria to resolve attacks into defeats; however, while ASPIC$^+$ evaluates arguments with the standard AF semantics, DeLP has a special-purpose definition of argument evaluation [71]. Therefore, the ideas developed here can be of inspiration to optimize the incremental computation of the semantics of ASPIC$^+$ programs.

7 Conclusions and Future Work

We have reviewed techniques for the incremental and efficient computation in dynamic abstract argumentation and defeasible knowledge bases. In the case of abstract argumentation, we have presented a technique enabling any non-incremental algorithm to be used as an incremental one for computing some extension of dynamic argumentation frameworks. The algorithm identifies a tighter portion of the updated argumentation framework to be examined for recomputing the semantics. The incremental algorithm proposed for Dung's frameworks enables a technique for the incremental computation of extensions of dynamic frameworks incorporating supports and second-order attacks (that we called extended argumentation frameworks). Recently, in [3], we have investigated incremental techniques for the ASAF framework [72], where attacks and support relations of any order are considered. For the case of structured argumentation, we have discussed an algorithm able to incrementally solve the problem determining the warrant status of literals in a DeLP program which is updated by adding or deleting strict or defeasible rules. The experimental analysis performed in [5; 7; 12; 11] showed that, in practice, the incremental approach, for both the cases of abstract and structured frameworks, turns out to be much more efficient than recomputing everything from scratch.

The notions behind the use of an incremental approach can be extended further, as done in [8; 1], where an incremental technique was recently proposed aimed at determining whether a given argument is skeptically preferred accepted in dynamic argumentation frameworks by exploiting the concept of influenced and reduced argumentation frameworks presented here in Section 3. Future work will be devoted to extending our technique to cope with other argumentation frameworks [13; 14] and other computational problems [2; 9; 15]. It would be interesting to deal with different interpretations of the support relation, *e.g.,* that one in [47; 48] where a meta argumentation approach is also adopted to deal with bipolarity. We plan to address the problem of incrementally enumerating *all* the extensions of an abstract argumentation framework. Following [8; 10], devising an incremental computation approach for the skeptical/credulous acceptance in dynamic argumentation frameworks, and its extensions (*e.g.,* bipolar argumentation frameworks and ASAFs), is another intriguing direction for future work. Finally, we believe the basic ideas presented for the case of structured argumentation could carry over to other frameworks, *e.g.,* ASPIC$^+$ or ABA; this is another research direction we are planning to pursue in future work.

Acknowledgments

The authors are grateful to the participants of the workshop *Current Trends in Formal Argumentation* (held at the University Centre of Bertinoro from November 3rd to 6th, 2019), and in particular to Matthias Thimm, for discussions on the limitations of the technique previously presented in [11] that improved the characterization of the irrelevant updates, and of the set of influenced literals, which are on the basis of the incremental approach for DeLP. Moreover, the authors wish to thank the reviewers for their comments and suggestions that helped us in improving the paper. Finally, this work was partly supported in Argentina by Universidad Nacional del Sur (UNS) under grant PGI 24/ZN34, Consejo Nacional de Investigaciones Cientificas y Técnicas (CONICET), and Agencia Nacional de Promoción Cientifica y Tecnológica under grant PICT-2018-0475.

References

[1] G. Alfano and S. Greco. Incremental skeptical preferred acceptance in dynamic argumentation frameworks. *IEEE Intelligent Systems*, 2021.

[2] Gianvincenzo Alfano, Marco Calautti, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. Explainable acceptance in probabilistic abstract argumentation: Complexity and approximation. In *Proceedings of the 17th International Conference on Principles of*

Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020, pages 33–43, 2020.

[3] Gianvincenzo Alfano, Andrea Cohen, Sebastian Gottifredi, Sergio Greco, Francesco Parisi, and Guillermo Ricardo Simari. Dynamics in abstract argumentation frameworks with recursive attack and support relations. In *Proc. of European Conference on Artificial Intelligence (ECAI)*, pages 577–584, 2020.

[4] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. Computing stable and preferred extensions of dynamic bipolar argumentation frameworks. In *Proc. of Workshop on Advances In Argumentation In Artificial Intelligence co-located with XVI International Conference of the Italian Association for Artificial Intelligence (AI*IA)*, pages 28–42, 2017.

[5] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. Efficient computation of extensions for dynamic abstract argumentation frameworks: An incremental approach. In *Proc. of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 49–55, 2017.

[6] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. Computing extensions of dynamic abstract argumentation frameworks with second-order attacks. In *IDEAS*, pages 183–192, 2018.

[7] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. A meta-argumentation approach for the efficient computation of stable and preferred extensions in dynamic bipolar argumentation frameworks. *Intelligenza Artificiale*, 12(2):193–211, 2018.

[8] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. An efficient algorithm for skeptical preferred acceptance in dynamic argumentation frameworks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 18–24, 2019.

[9] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. On scaling the enumeration of the preferred extensions of abstract argumentation frameworks. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, SAC 2019, Limassol, Cyprus, April 8-12, 2019*, pages 1147–1153. ACM, 2019.

[10] Gianvincenzo Alfano, Sergio Greco, and Francesco Parisi. Computing skeptical preferred acceptance in dynamic argumentation frameworks with recursive attack and support relations. In *Proceedings of COMMA*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 67–78. IOS Press, 2020.

[11] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, Gerardo I. Simari, and Guillermo R. Simari. An incremental approach to structured argumentation over dynamic knowledge bases. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR*, pages 78–87, 2018.

[12] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, Gerardo I. Simari, and Guillermo R. Simari. Incremental computation of warranted arguments in dynamic defeasible argumentation: The rule addition case. In *Proc. of the Symposium on Applied Computing (SAC)*, pages 911–917, 2018.

[13] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. On the se-

mantics of abstract argumentation frameworks: A logic programming approach. *Theory Pract. Log. Program.*, 20(5):703–718, 2020.

[14] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. On the semantics of recursive bipolar afs and partial stable models. In *Proceedings of the Workshop on Advances In Argumentation In Artificial Intelligence 2020 co-located with the 19th International Conference of the Italian Association for Artificial Intelligence (AIxIA 2020), Online, November 25-26, 2020*, volume 2777 of *CEUR Workshop Proceedings*, pages 16–30. CEUR-WS.org, 2020.

[15] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, and Irina Trubitsyna. Argumentation frameworks with strong and weak constraints: Semantics and complexity. In *Proc. of AAAI*, page (to appear), 2021.

[16] Leila Amgoud, Jean-François Bonnefon, and Henri Prade. An argumentation-based approach to multiple criteria decision. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU)*, pages 269–280, 2005.

[17] Leila Amgoud, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *Proc. of International Workshop on Non-Monotonic Reasoning (NMR)*, pages 1–9, 2004.

[18] Leila Amgoud and Srdjan Vesic. Revising option status in argument-based decision systems. *Journal of Logic and Computation*, 22(5):1019–1058, 2012.

[19] Katie Atkinson, Pietro Baroni, Massimiliano Giacomin, Anthony Hunter, Henry Prakken, Chris Reed, Guillermo R. Simari, Matthias Thimm, and Serena Villata. Towards artificial argumentation. *AI Magazine*, 38(3):25–36, 2017.

[20] Pietro Baroni, Guido Boella, Federico Cerutti, Massimiliano Giacomin, Leendert W. N. van der Torre, and Serena Villata. On the input/output behavior of argumentation frameworks. *Artificial Intelligence*, 217:144–197, 2014.

[21] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011.

[22] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. Encompassing attacks to attacks in abstract argumentation frameworks. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, pages 83–94, 2009.

[23] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. AFRA: Argumentation Framework with Recursive Attacks. *IJAR*, 52(1):19–37, 2011.

[24] Pietro Baroni, Massimiliano Giacomin, and Beishui Liao. On topology-related properties of abstract argumentation semantics. A correction and extension to dynamics of argumentation systems: A division-based method. *Artificial Intelligence*, 212:104–115, 2014.

[25] Ringo Baumann. Splitting an argumentation framework. In *Proc. of International Conference on Logic Programming and Non-Monotonic Reasoning (LPNMR)*, pages 40–53, 2011.

[26] Ringo Baumann. Normal and strong expansion equivalence for argumentation frameworks. *Artificial Intelligence*, 193:18–44, 2012.

[27] Ringo Baumann. Context-free and context-sensitive kernels: Update and deletion equivalence in abstract argumentation. In *Proc. of European Conference on Artificial Intelligence (ECAI)*, pages 63–68, 2014.

[28] Ringo Baumann and Gerhard Brewka. Expanding argumentation frameworks: Enforcing and monotonicity results. In *Proc. of Third International Conference on Computational Models of Argument (COMMA)*, pages 75–86, 2010.

[29] Trevor J. M. Bench-Capon and Paul E. Dunne. Argumentation in artificial intelligence. *Artificial Intelligence*, 171(10 - 15):619 – 641, 2007.

[30] Philippe Besnard, Alejandro J. Garcia, Anthony Hunter, Sanjay Modgil, Henry Prakken, Guillermo R. Simari, and Francesca Toni. Introduction to structured argumentation. *Argument & Computation – Special Issue: Tutorials on Structured Argumentation*, 5(1):1–4, 2014.

[31] Philippe Besnard and Anthony Hunter. *Elements of Argumentation.* MIT Press, 2008.

[32] Philippe Besnard and Anthony Hunter. Constructing argument graphs with deductive arguments: A tutorial. *Argument & Computation*, 5(1):5–30, 2014.

[33] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Change in argumentation systems: Exploring the interest of removing an argument. In *Proceedings of the 5th International Conference on Scalable Uncertainty Management (SUM)*, pages 275–288, 2011.

[34] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Changement dans un système d'argumentation : suppression d'un argument. *Rev. d'Intelligence Artif.*, 26(3):225–253, 2012.

[35] Pierre Bisquert, Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Characterizing change in abstract argumentation systems. In *Trends in Belief Revision and Argumentation Dynamics*, volume 48, pages 75–102. 2013.

[36] Stefano Bistarelli, Francesco Faloci, Francesco Santini, and Carlo Taticchi. Studying dynamics in argumentation with Rob. In *COMMA*, pages 451–452, 2018.

[37] Stefano Bistarelli, Lars Kotthoff, Francesco Santini, and Carlo Taticchi. Containerisation and dynamic frameworks in iccma'19. In *Proceedings of the Second International Workshop on Systems and Algorithms for Formal Argumentation (SAFA 2018) co-located with the 7th International Conference on Computational Models of Argument (COMMA 2018), Warsaw, Poland, September 11, 2018.*, pages 4–9, 2018.

[38] Guido Boella, Dov M. Gabbay, Leendert W. N. van der Torre, and Serena Villata. Support in abstract argumentation. In *Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010.*, pages 111–122, 2010.

[39] Guido Boella, Souhila Kaci, and Leendert W. N. van der Torre. Dynamics in argumentation with single extensions: Abstraction principles and the grounded extension. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning*

and Uncertainty (ECSQARU), pages 107–118, 2009.

[40] Guido Boella, Souhila Kaci, and Leendert W. N. van der Torre. Dynamics in argumentation with single extensions: Attack refinement and the grounded extension. In *Proc. of Sixth Int. Workshop on Argumentation in Multi-Agent Systems (ArgMAS)*, pages 150–159, 2009.

[41] Martin Caminada. Semi-stable semantics. In *Proc. of 1st International Conference on Computational Models of Argument (COMMA)*, pages 121–130, 2006.

[42] Marcela Capobianco, Carlos I. Chesñevar, and Guillermo R. Simari. Argumentation and the dynamics of warranted beliefs in changing environments. *Autonomous Agents and Multi-Agent Systems*, 11(2):127–151, 2005.

[43] Marcela Capobianco and Guillermo R. Simari. A proposal for making argumentation computationally capable of handling large repositories of uncertain data. In *Proc. of Scalable Uncertainty Management*, pages 95–110, 2009.

[44] Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Revision of an argumentation system. In *Proc. of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 124–134, 2008.

[45] Claudette Cayrol, Florence Dupin de Saint-Cyr, and Marie-Christine Lagasquie-Schiex. Change in abstract argumentation frameworks: Adding an argument. *Journal of Artificial Intelligence Research*, 38:49–84, 2010.

[46] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU)*, pages 378–389, 2005.

[47] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Bipolar abstract argumentation systems. In *Argumentation in Artificial Intelligence*, pages 65–84. 2009.

[48] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Coalitions of arguments: A tool for handling bipolar argumentation frameworks. *International Journal of Intelligent System*, 25(1):83–109, 2010.

[49] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Change in abstract bipolar argumentation systems. In *Proc. of International Conference on Scalable Uncertainty Management (SUM)*, pages 314–329, 2015.

[50] Laura A. Cecchi, Pablo R. Fillottrani, and Guillermo R. Simari. On the complexity of DeLP through game semantics. In *Proc. of the 11th International Workshop on Nonmonotonic Reasoning, Lake District, GBP*, pages 386–394, 2006.

[51] Günther Charwat, Wolfgang Dvorák, Sarah A. Gaggl, Johannes P. Wallner, and Stefan Woltran. Methods for solving reasoning problems in abstract argumentation - A Survey. *Artificial Intelligence*, 220:28–63, 2015.

[52] Carlos I. Chesñevar and Guillermo R. Simari. Modelling inference in argumentation through labelled deduction: Formalization and logical properties. *Logica Universalis*, 1(1):93–124, 2007.

[53] Andrea Cohen, Sebastian Gottifredi, Alejandro J. Garcia, and Guillermo R. Simari. A

survey of different approaches to support in argumentation systems. *The Knowledge Engineering Review*, 29(5):513–550, 2014.

[54] Cristhian A. D. Deagustini, Santiago E. Fulladoza Dalibón, Sebastian Gottifredi, Marcelo A. Falappa, Carlos I. Chesñevar, and Guillermo R. Simari. Relational databases as a massive information source for defeasible argumentation. *Knowledge-Based Systems*, 51:93–109, 2013.

[55] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

[56] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10-15):642–674, 2007.

[57] Paul E. Dunne. The computational complexity of ideal semantics. *Artificial Intelligence*, 173(18):1559–1591, 2009.

[58] Paul E. Dunne and Michael Wooldridge. Complexity of abstract argumentation. In *Argumentation in Artificial Intelligence*, pages 85–104. 2009.

[59] Wolfgang Dvorak, Reinhard Pichler, and Stefan Woltran. Towards fixed-parameter tractable algorithms for argumentation. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU)*, 2010.

[60] Wolfgang Dvorak and Stefan Woltran. Complexity of semi-stable and stage semantics in argumentation frameworks. *Information Processing Letters*, 110(11):425–430, 2010.

[61] Thomas Eiter, Hannes Strass, Miroslaw Truszczynski, and Stefan Woltran, editors. *Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation*, volume 9060. Springer, 2015.

[62] Marcelo A. Falappa, Alejandro J. Garcia, Gabriele Kern-Isberner, and Guillermo R. Simari. On the evolving relation between belief revision and argumentation. *The Knowledge Engineering Review*, 26(1):35–43, 2011.

[63] Marcelo A. Falappa, Gabriele Kern-Isberner, and Guillermo R. Simari. Explanations, belief revision and defeasible reasoning. *Artif. Intell.*, 141(1/2):1–28, 2002.

[64] Bettina Fazzinga, Sergio Flesca, and Francesco Parisi. Efficiently estimating the probability of extensions in abstract argumentation. In *Proc. of International Conference on Scalable Uncertainty Management (SUM)*, pages 106–119, 2013.

[65] Bettina Fazzinga, Sergio Flesca, and Francesco Parisi. On the complexity of probabilistic abstract argumentation frameworks. *ACM Transactions on Computational Logic*, 16(3):22, 2015.

[66] Bettina Fazzinga, Sergio Flesca, and Francesco Parisi. On efficiently estimating the probability of extensions in abstract argumentation frameworks. *International Journal of Approximate Reasoning*, 69:106–132, 2016.

[67] Bettina Fazzinga, Sergio Flesca, Francesco Parisi, and Adriana Pietramala. PARTY: A mobile system for efficiently assessing the probability of extensions in a debate. In *Proc. of International Conference on Database and Expert Systems Applications (DEXA)*, pages 220–235, 2015.

[68] Alejandro J. Garcia, Henry Prakken, and Guillermo R. Simari. A comparative study of some central notions of ASPIC$^+$ and DeLP. *Theory and Practice of Logic Programming*, 20(3):358–390, 2020.

[69] Alejandro J. Garcia and Guillermo R. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming (TPLP)*, 4(1-2):95–138, 2004.

[70] Alejandro J. Garcia and Guillermo R. Simari. Defeasible logic programming: DeLP-servers, contextual queries, and explanations for answers. *Argument & Computation*, 5(1):63–88, 2014.

[71] Alejandro Javier Garcia, Henry Prakken, and Guillermo Ricardo Simari. A comparative study of some central notions of ASPIC+ and delp. *Theory Pract. Log. Program.*, 20(3):358–390, 2020.

[72] Sebastian Gottifredi, Andrea Cohen, Alejandro J. Garcia, and Guillermo R. Simari. Characterizing acceptability semantics of argumentation frameworks with recursive attack and support relations. *Artif. Intell.*, 262:336–368, 2018.

[73] Sebastian Gottifredi, Nicolas D. Rotstein, Alejandro J. Garcia, and Guillermo R. Simari. Using argument strength for building dialectical bonsai. *Annals of Mathematics and Artificial Intelligence*, 69(1):103–129, 2013.

[74] Sergio Greco and Francesco Parisi. Efficient computation of deterministic extensions for dynamic abstract argumentation frameworks. In *Proc. of European Conference on Artificial Intelligence (ECAI)*, pages 1668–1669, 2016.

[75] Sergio Greco and Francesco Parisi. Incremental computation of deterministic extensions for dynamic argumentation frameworks. In *Proc. of European Conference On Logics In Artificial Intelligence (JELIA)*, pages 288–304, 2016.

[76] Nadin Kökciyan, Nefise Yaglikci, and Pinar Yolum. Argumentation for resolving privacy disputes in online social networks: (extended abstract). In *Proc. of International Conference on Autonomous Agents and Multiagent Sytems (AAMAS)*, pages 1361–1362, 2016.

[77] Markus Kröll, Reinhard Pichler, and Stefan Woltran. On the complexity of enumerating the extensions of abstract argumentation frameworks. In *Proc. of IJCAI*, pages 1145–1152, 2017.

[78] Mario A. Leiva, Gerardo I. Simari, Sebastian Gottifredi, Alejandro J. Garcia, and Guillermo R. Simari. DAQAP: defeasible argumentation query answering platform. In *Proceedings of the 13th International Conference on Flexible Query Answering Systems (FQAS)*, pages 126–138, 2019.

[79] Beishui Liao. Toward incremental computation of argumentation semantics: A decomposition-based approach. *Annals of Mathematics and Artificial Intelligence*, 67(3-4):319–358, 2013.

[80] Beishui Liao and Huaxin Huang. Partial semantics of argumentation: basic properties and empirical results. *Journal of Logic and Computation*, 23(3):541–562, 2013.

[81] Beishui Liao, Li Jin, and Robert C. Koons. Dynamics of argumentation systems: A

division-based method. *Artificial Intelligence*, 175(11):1790–1814, 2011.

[82] Sanjay Modgil. An abstract theory of argumentation that accommodates defeasible reasoning about preferences. In *Proc. of European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU)*, pages 648–659, 2007.

[83] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173(9-10):901–934, 2009.

[84] Sanjay Modgil and Henry Prakken. Revisiting preferences and argumentation. In *Proc. of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1021–1026, 2011.

[85] Sanjay Modgil and Henry Prakken. The ASPIC$^+$ framework for structured argumentation: A tutorial. *Argument & Computation*, 5(1):31–62, 2014.

[86] Sanjay Modgil, Francesca Toni, Floris Bex, Ivan Bratko, Carlos I. Chesnevar, Wolfgang Dvorak, Marcelo A. Falappa, Xiuyi Fan, Sarah Alice Gaggl, Alejandro J. Garcia, Maria P. Gonzalez, Thomas F. Gordon, Joao Leite, Martin Mouzina, Chris Reed, Guillermo R. Simari, Stefan Szeider, Paolo Torroni, and Stefan Woltran. *Agreement Technologies*, volume 8 of *Law, Governance and Technology*, chapter 21: The Added Value of Argumentation: Examples and Challenges, pages 357–404. Springer, 2013.

[87] Martin O. Moguillansky, Nicolas D. Rotstein, Marcelo A. Falappa, Alejandro J. Garcia, and Guillermo R. Simari. Dynamics of knowledge in DeLP through argument theory change. *TPLP*, 13(6):893–957, 2013.

[88] Farid Nouioua and Vincent Risch. Bipolar argumentation frameworks with specialized supports. In *Proc. of ICTAI*, pages 215–218, 2010.

[89] Farid Nouioua and Vincent Risch. Argumentation frameworks with necessities. In *Proc. of SUM*, pages 163–176, 2011.

[90] Emilia Oikarinen and Stefan Woltran. Characterizing strong equivalence for argumentation frameworks. *Artificial Intelligence*, 175(14-15):1985–2009, 2011.

[91] Iyad Rahwan and Guillermo R. Simari. *Argumentation in Artificial Intelligence.* Springer Publishing Company, Incorporated, 1st edition, 2009.

[92] Bas Testerink, Daphne Odekerken, and Floris Bex. A method for efficient argument-based inquiry. In *Proceedings of the 13th International Conference on Flexible Query Answering Systems (FQAS)*, pages 114–125, 2019.

[93] Matthias Thimm and Serena Villata. The first international competition on computational models of argumentation: Results and analysis. *Artificial Intelligence*, 252:267–294, 2017.

[94] Francesca Toni. A tutorial on assumption-based argumentation. *Argument & Computation*, 5(1):89–117, 2014.

[95] Ignacio D. Viglizzo, Fernando A. Tohmé, and Guillermo R. Simari. *Annals of Mathematics and Artificial Intelligence*, 57(2):181–204, 2009.

[96] Serena Villata, Guido Boella, Dov M. Gabbay, and Leendert W. N. van der Torre. Modelling defeasible and prioritized support in bipolar argumentation. *Annals of Mathematics and Artificial Intelligence*, 66(1-4):163–197, 2012.

[97] Yuming Xu and Claudette Cayrol. The matrix approach for abstract argumentation frameworks. In *Proc. of International Workshop on Theory and Applications of Formal Argumentation (TAFA)*, pages 243–259, 2015.

Received May 2021

Logic-Based Approaches to Formal Argumentation

Ofer Arieli
School of Computer Science, The Academic College of Tel-Aviv, Israel
oarieli@mta.ac.il

AnneMarie Borg
Department of Information and Computing Sciences, Utrecht University, The Netherlands
a.borg@uu.nl

Jesse Heyninck
Department of Computer Science, TU Dortmund, Germany
jesse.heyninck@tu-dortmund.de

Christian Straßer
Institute of Philosophy II, Ruhr University Bochum, Germany
christian.strasser@rub.de

Abstract

We study the logical foundations of Dung-style argumentation frameworks. Logic-based methods in the context of argumentation theory are described from two perspectives: (a) a survey of logic-based instantiations of argumentation frameworks, their properties and relations, and (b) a review of logical methods for the study of argumentation dynamics. In this chapter we restrict ourselves to Tarskian logics, based on (propositional) languages and corresponding (constructive) semantics or syntactic rule-based systems.

1 Motivation, Introduction and Scope

The purpose of this chapter is to study the *logical* foundations of formal argumentation and highlight its role in the modeling of defeasible reasoning. For this, we assume the availability of an underlying *logic* (that is, a pair of a formal propositional language and a corresponding (reflexive, monotonic, and transitive) consequence relation), upon which argumentation-based formalisms are defined. We then study logic-based approaches to formal argumentation from two perspectives. One perspective is concerned with instantiations of argumentation frameworks by logic-based formalisms. The need to instantiate Dung's abstract argumentation frameworks [85] by deductive (or, more generally, structured) approaches is well acknowledged in the literature (see, e.g., [66; 151; 153] for some papers on the subject), and is primarily motivated by giving logical justifications to the notions of arguments and counter-arguments. Moreover, several fundamental mathematical and philosophical notions that cannot be studied in an abstract context (or at least not natural to this context), can be investigated in a logic-based setting. This includes, for example, properties such as consistency, maximal consistency [155], deductive closure [60], logical omniscience, and so forth, as well as inference principles that can be related to general patterns of non-monotonic and paraconsistent reasoning, and which are better suited to a deductive (logic-based) setting.

The second perspective taken in this chapter is related to the use of logic-based machinery to describe (that is, represent and reason with) argumentation-based dynamics. Indeed, the availability of an underlying 'core' logic triggers a wide variety of methods for formally expressing argumentation-related processes. For instance, since modal logics allow to qualify statements, alethic arguments (about necessity and possibility), epistemic ones (about knowledge and belief) [128; 84], and deontic phrases (about obligations and permissios) [179; 104; 168] can be expressed, giving rise to different applications in linguistics, security and game theory (see e.g., [40] and [84]). Also, the presence of an underlying logic allows for incorporation of proof-theoretical methods [16] and related structural methodologies [114] to reason with argumentation frameworks and characterize their properties (see also [102; 103]).

This chapter is divided into two parts according to the two perspectives described above. The first part of the chapter, given in Section 2, is focused on the first perspective, namely: a study of logic-based approaches to formal argumentation. The formalisms that are investigated in this part are those that are based on some underlying (core) *logic* (in the traditional sense of this notion, described in Definition 1 and Remark 1). This means, in particular, that not only the arguments in these formalisms have a particular structure (as opposed to abstract argumentation frameworks [85; 23], where an abstraction is made of the structure of arguments), but also that their validity can be logically justified. It follows that not all the formalisms under the umbrella of structured argumentation will be considered in this chapter, but only those that are based on specific core logics.

To study the logical instantiations of formal Dung-style argumentation, we first recall, in Section 2.2, three central approaches that correspond to this line of research: logic-based deductive methods [35; 14; 38], assumption-based argumentation systems [46; 171; 73] and ASPIC systems [150; 146; 147]. Then, in Section 2.3, we consider the main properties of each approach, in particular: its relation to reasoning with maximal consistency, the rationality postulates that it satisfies, and the inference principles validated by the induced entailment relations. Finally, in Section 2.4, we study relations among these approaches, as well as their relations to other defeasible reasoning methods.

The second part of this chapter describes logic-based methods for representing and reasoning with argumentation dynamics. In this chapter, by 'dynamics' we mean *processes* in the context of a fixed argumentative framework.[1] Basic notions and concepts such as conflicting arguments, defending arguments, and Dung-style extensions are expressed by logical formulas, and corresponding reasoning processes, based on proof-theoretical methods, are described. The representations are divided between those that are based on propositional languages or their extensions by quantifications (Section 3.1), and those that incorporate modal operators (Section 3.2). The reasoning machinery described in this chapter (Section 3.3) is again one that takes into account the logical relationships among the arguments (although it can be easily adjusted to abstract entities). It can be seen as an extension of Gentzen-type proof calculi [110], in which the dynamics of arguments are taken into consideration, and so the proofs are dynamic, in the sense that a derived argument can be retracted in light of more-recently derived counter-arguments [15; 16].

We conclude the chapter with some final remarks (Section 4) and proofs of unpublished results (in the appendix). The general structure of this chapter is sketched in Figure 1.

We note, finally, that due to the broad scope of this chapter, some parts of it may be viewed as "second-order" surveys, pointing to other reviews on specific sub-topics of this chapter. In some other parts we give more detailed descriptions on specific formalisms. We do so mainly for illustrating our points, but this should not be taken as a preference of one method over the others.

2 Logical Instantiations

The first part of this chapter is devoted to logic-based instantiations of formal argumentation. We describe different approaches to logical argumentation (Section 2.2), consider some of

[1] A similar terminology is sometimes used in the context of revising argumentation frameworks, see also Chapters 8 [28] and 11 [1] in this handbook.

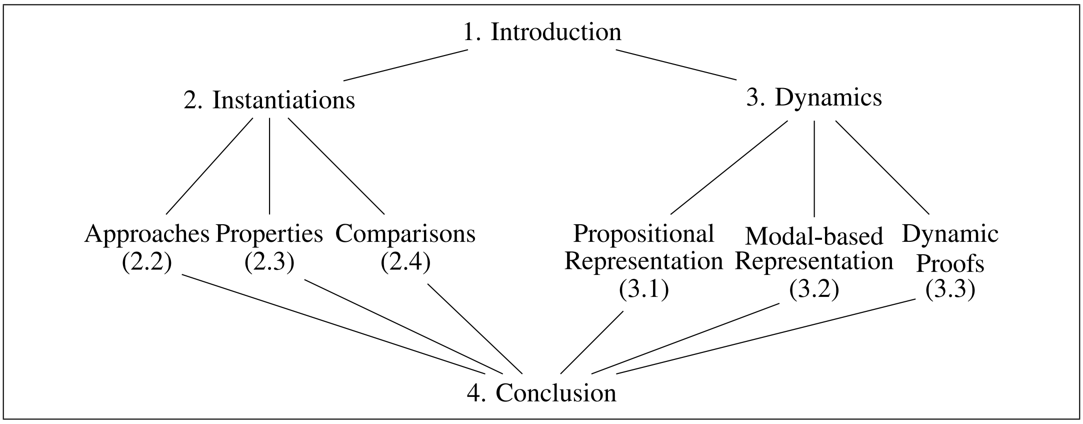


Figure 1: Schematic structure of the chapter

their properties (Section 2.3), and review the (known) relations among them (Section 2.4). First, we recall some common notions and notations.

2.1 Preliminaries

In what follows we shall assume that the underlying language $\mathcal{L}$ is propositional. Sets of formulas are denoted by $\mathcal{S}, \mathcal{T}$, finite sets of formulas are denoted by $\Gamma, \Delta, \Pi, \Theta$, formulas are denoted by $\phi, \psi, \delta, \gamma$, and atomic formulas are denoted by p, q, r, all of which can be primed or indexed. The set of all the atomic formulas of $\mathcal{L}$ is denoted $\mathsf{Atoms}(\mathcal{L})$, and the set of the (well-formed) formulas of $\mathcal{L}$ is denoted $\mathsf{WFF}(\mathcal{L})$.

All the approaches to formal argumentation considered in this chapter assume an underlying logic that forms the basis for specifying arguments and counter-arguments. The next definition is thus at the heart of our study.

Definition 1 (logic)**.** A (propositional) *logic* is a pair $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$, where $\mathcal{L}$ is a propositional language, and $\vdash$ is a (Tarskian, [170]) *consequence relation* for a language $\mathcal{L}$, that is: a binary relation between sets of formulas and formulas in $\mathcal{L}$, satisfying the following conditions:

- *Reflexivity*: if $\psi \in \mathcal{S}$ then $\mathcal{S} \vdash \psi$.
- *Monotonicity*: if $\mathcal{S} \vdash \psi$ and $\mathcal{S} \subseteq \mathcal{S}'$ then $\mathcal{S}' \vdash \psi$.
- *Transitivity*: if $\mathcal{S} \vdash \psi$ and $\mathcal{S}', \psi \vdash \phi$ then $\mathcal{S}, \mathcal{S}' \vdash \phi$.[2]

[2] As usual, we use the notation $\mathcal{S}, \mathcal{S}'$ on the left-hand side of the entailment symbol to denote $\mathcal{S} \cup \mathcal{S}'$. In case of singletons we shall usually omit the parenthesis and abbreviate $\mathcal{S} \cup \{\psi\}$ by $\mathcal{S}, \psi$.

In what follows we also assume that a consequence relation satisfies some further standard conditions:

- *Structurality*: for every $\mathcal{L}$-substitution θ,[3] if $\mathcal{S} \vdash \psi$ then $\theta(\mathcal{S}) \vdash \theta(\psi)$.
- *Non-Triviality*: $p \nvdash q$ for every two distinct atomic formulas p and q.
- *Finitariness*: if $\mathcal{S} \vdash \psi$, there is a *finite* set $\Gamma \subseteq \mathcal{S}$ such that $\Gamma \vdash \psi$.

Structurality means closure under substitutions of formulas. Non-triviality is convenient for excluding trivial logics, and finitariness is often essential for practical reasoning, such as being able to form arguments (based on a finite number of assumptions) for entailments with possibly infinite number of premises.

To some extent, Definition 1 determines the instantiations covered in Section 2.2 (and the scope of the whole chapter in general): not only that the arguments should have a specific structure (unlike, e.g., arguments in abstract argumentation frameworks that are of a purely abstract nature), but they should be based on (i.e., justified by) some underlying logic as well (see also Definitions 4 and 5).[4] As indicated in Definition 1, in the sequel we shall consider (arbitrary) propositional logics, although most of the formalisms can be easily extended to more generic logics (including first-ordered ones), since the relevant ideas and approaches can be represented at this level.

In what follows we shall assume that the language $\mathcal{L}$ contains at least the following connectives and constant:

a $\vdash$-*negation* $\neg$, satisfying: $p \nvdash \neg p$ and $\neg p \nvdash p$ (for every atomic p),

a $\vdash$-*conjunction* $\wedge$, satisfying: $\mathcal{S} \vdash \psi \wedge \phi$ iff $\mathcal{S} \vdash \psi$ and $\mathcal{S} \vdash \phi$,

a $\vdash$-*disjunction* $\vee$, satisfying: $\mathcal{S}, \phi \vee \psi \vdash \sigma$ iff $\mathcal{S}, \phi \vdash \sigma$ and $\mathcal{S}, \psi \vdash \sigma$,

a $\vdash$-*implication* $\supset$, satisfying: $\mathcal{S}, \phi \vdash \psi$ iff $\mathcal{S} \vdash \phi \supset \psi$,

a $\vdash$-*falsity* F, satisfying: $\mathsf{F} \vdash \psi$ for every formula ψ.[5]

[3] That is, θ is a finite set of pairs $\{(p_1, \psi_1), \dots (p_n, \psi_n)\}$, where for every $1 \leq i \leq n$, p_i is an atom and ψ_i is an $\mathcal{L}$-formula, such that for every $\mathcal{L}$-formula ϕ, the $\mathcal{L}$-formula $\theta(\phi)$ is obtained from ϕ by replacing in it each occurrence of p_i by ψ_i $(i = 1 \dots, n)$. We denote $\theta(\mathcal{S}) = \{\theta(\phi) \mid \phi \in \mathcal{S}\}$.

[4] Note that this means that some approaches to structured argumentation whose underlying formalisms do not meet the conditions of Definition 1 are not covered in Section 2.2, such as defeasible logic programming [106] and instances of ASPIC$^+$ where neither strict nor defeasible rules are based on a logic in the sense of Definition 1.

[5] In particular, F is not a standard atomic formula, since $\mathsf{F} \vdash \neg\mathsf{F}$.

In what follows, we shall abbreviate $(\phi \supset \psi) \wedge (\psi \supset \phi)$ by $\phi \leftrightarrow \psi$. For a set of formulas $\mathcal{S}$ we denote $\neg\mathcal{S} = \{\neg\psi \mid \psi \in \mathcal{S}\}$, and for a finite set of formulas Γ we denote by $\bigwedge\Gamma$ (respectively, by $\bigvee\Gamma$) the conjunction (respectively, the disjunction) of all the formulas in Γ. The powerset of $\mathcal{L}$ is denoted by $\wp(\mathcal{L})$. Now,

- We say that an $\mathcal{L}$-formula ψ is a $\vdash$-*theorem*, if $\emptyset \vdash \psi$.
- The $\vdash$-transitive closure of a set $\mathcal{S}$ of $\mathcal{L}$-formulas is defined by $Cn_{\vdash}(\mathcal{S}) = \{\psi \mid \mathcal{S} \vdash \psi\}$.
- We shall say that a set $\mathcal{S}$ is $\vdash$-*consistent* if $\mathcal{S} \nvdash \mathsf{F}$. In particular, if $\mathcal{S}$ is *not* $\vdash$-consistent (i.e, if it is $\vdash$-inconsistent), it is trivialized with respect to $\vdash$ in the sense that $Cn_{\vdash}(\mathcal{S})$ consists of every formula in $\mathcal{L}$. Note, in particular, that if $\mathcal{S}$ is $\vdash$-inconsistent, then $\mathcal{S} \vdash \neg\bigwedge\Gamma$ for $\Gamma \subseteq \mathcal{S}$.

When $\vdash$ is clear from the context we will sometimes omit it from the notations above (and say that a formula is a theorem, a set of formulas is consistent, and write $Cn(\mathcal{S})$ for the $\vdash$-transitive closure $\mathcal{S}$).

Remark 2. *To all of the instantiations considered here there are extensions in which the language contains also non-logical components such as priorities among the arguments. As we concentrate on purely logical approaches, these extensions will not be covered in this chapter.*

Definition 3 (explosive/contrapositive logic)**.** *A logic $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ is* explosive*, if for every $\mathcal{L}$-formula ψ the set $\{\psi, \neg\psi\}$ is $\vdash$-inconsistent.*[6] *We say that $\mathfrak{L}$ is* contrapositive*, if (a) $\vdash \neg\mathsf{F}$ and (b) for every nonempty Γ and ψ it holds that $\Gamma \vdash \neg\psi$ iff for every $\phi \in \Gamma$ we have: $\Gamma \setminus \{\phi\}, \psi \vdash \neg\phi$.*

2.2 Central Approaches to Logical Argumentation

In this section we review some central approaches to logical argumentation. Further details about these approaches, related approaches, and relevant references can be found in [152; 34; 38; 151].

[6] That is, $\psi, \neg\psi \vdash \mathsf{F}$. Thus, in explosive logics every formula follows from complementary assumptions.

2.2.1 Logic-Based Methods

A. Arguments. Some of the first works on logic-based formal argumentation used classical logic (CL) as the underlying base logic to generate arguments. This indeed is the most common approach in the study and implementation of such argumentation frameworks. To avoid trivial reasoning in such cases, the set of assumptions of an argument (the so-called argument's *support*) is assumed to be consistent and frequently also minimal, in the sense that no proper subset of the argument's support entails the argument's conclusion (see [35; 36; 111; 37; 38]). This leads to the following definition:

Definition 4 (classical argument)**.** A *classical argument* is a pair $A = \langle \Gamma, \psi \rangle$, where Γ is a finite set of formulas in the language of $\{\neg, \vee, \wedge, \supset, \mathsf{F}\}$ (with their usual bivalent interpretations), such that: (1) $\Gamma \vdash_{\mathsf{CL}} \psi$ (namely: ψ follows, according to classical logic, from Γ), (2) Γ is $\vdash_{\mathsf{CL}}$-consistent, and (3) for no $\Gamma' \subsetneq \Gamma$ it holds that $\Gamma' \vdash_{\mathsf{CL}} \psi$.

A more general view of arguments (which will be taken here) allows to base arguments on arbitrary logics, and relaxes the two assumptions (consistency and minimality) on their supports (see, e.g, [14; 38]):[7]

Definition 5 (argument)**.** Given a logic $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$, an $\mathfrak{L}$*-argument* (an *argument*, for short) is a pair $A = \langle \Gamma, \psi \rangle$, where Γ is a finite set of $\mathcal{L}$-formulas and ψ is an $\mathcal{L}$-formula, such that $\Gamma \vdash \psi$. We denote the set of all $\mathfrak{L}$-arguments by $\mathsf{Arg}_{\mathfrak{L}}$.

In what follows, we shall usually denote arguments by A, B, C, etc., possibly primed or indexed. Now:

- Given an argument $A = \langle \Gamma, \psi \rangle$, we shall call Γ the *support set* (or the *premise set*) of A, and ψ the *conclusion* (or the *claim*) of A, denoting them by $\mathsf{Sup}(A)$ and $\mathsf{Conc}(A)$, respectively. For a set S of arguments, we denote: $\mathsf{Sup}(\mathsf{S}) = \bigcup_{A \in \mathsf{S}} \mathsf{Sup}(A)$ and $\mathsf{Conc}(\mathsf{S}) = \{\mathsf{Conc}(A) \mid A \in \mathsf{S}\}$.

- The set of the $\mathfrak{L}$-arguments whose supports are subsets of $\mathcal{S}$ is denoted by $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$. That is: $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}) = \{A \in \mathsf{Arg}_{\mathfrak{L}} \mid \mathsf{Sup}(A) \subseteq \mathcal{S}\}$.

- Given an argument $A \in \mathsf{Arg}_{\mathfrak{L}}$, its set of *sub-arguments* is denoted by $\mathsf{Sub}(A)$. That is: $\mathsf{Sub}(A) = \{B \in \mathsf{Arg}_{\mathfrak{L}} \mid \mathsf{Sup}(B) \subseteq \mathsf{Sup}(A)\}$.

Remark 6. *An alternative notation for an argument* $\langle \Gamma, \psi \rangle$ *is* $\Gamma \Rightarrow \psi$ *(where* $\Rightarrow$ *is a new symbol, not appearing in the language of* Γ *and* ψ*). The latter resembles the way sequents*

[7] See, e.g., [17] for a comparison of Definitions 4 and 5.

are denoted in the context of proof theory [110]. This notation is frequently used in sequent-based argumentation *(see, e.g., [14; 16]) to emphasize the fact that the only requirement on* Γ *and* ψ *to form an argument is that the latter follows, according to the base logic, from the former.*

B. Attacks. Disagreements between arguments are often described in terms of counter-arguments. It is often said that a counter-argument *attacks* the argument that it challenges.[8] Attacks between arguments are usually described in terms of *attack rules* (with respect to the underlying logic). Table 1 lists some of them. Other attack rules between classical arguments are described e.g. in [111] and [38, Section 5.2]. For a variety of attacks in terms of sequents we refer to [14]. Attack rules incorporating modalities are introduced in [168].

Rule Name	**Acronym**	**Attacking Argument**	**Attacked Argument**	**Attack Conditions**
Defeat	Def	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \Gamma_2, \psi_2 \rangle$	$\vdash \psi_1 \supset \neg \bigwedge \Gamma_2$
Direct Defeat	DirDef	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \{\gamma_2\} \cup \Gamma_2', \psi_2 \rangle$	$\vdash \psi_1 \supset \neg \gamma_2$
Undercut	Ucut	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \Gamma_2' \cup \Gamma_2'', \psi_2 \rangle$	$\vdash \psi_1 \leftrightarrow \neg \bigwedge \Gamma_2'$
Canonical Undercut	CanUcut	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \Gamma_2, \psi_2 \rangle$	$\vdash \psi_1 \leftrightarrow \neg \bigwedge \Gamma_2$
Direct Undercut	DirUcut	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \{\gamma_2\} \cup \Gamma_2', \psi_2 \rangle$	$\vdash \psi_1 \leftrightarrow \neg \gamma_2$
Consistency Undercut	ConUcut	$\langle \emptyset, \neg \bigwedge \Gamma_2' \rangle$	$\langle \Gamma_2' \cup \Gamma_2'', \psi_2 \rangle$	
Rebuttal	Reb	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \Gamma_2, \psi_2 \rangle$	$\vdash \psi_1 \leftrightarrow \neg \psi_2$
Defeating Rebuttal	DefReb	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \Gamma_2, \psi_2 \rangle$	$\vdash \psi_1 \supset \neg \psi_2$
Big Argument Attack	BigArgAt	$\langle \Gamma_1, \psi_1 \rangle$	$\langle \{\gamma_2\} \cup \Gamma_2', \psi_2 \rangle$	$\vdash \bigwedge \Gamma_1 \supset \neg \gamma_2$

Table 1: Some attack rules. The support sets of the attacked arguments are assumed to be nonempty (to avoid attacks on theorems).

Rules like those specified in Table 1 form attack schemes that are applied to particular arguments according to the underlying logic. For instance, when classical logic is the underlying formalism, the attacks of $\langle p, p \rangle$ on $\langle \neg p, \neg p \rangle$ and of $\langle \neg p, \neg p \rangle$ on $\langle p \wedge q, p \rangle$[9] are

[8] Sometimes, mainly when priorities among arguments are introduced, or in the context of specific types of attacks, the term "defeat" is used for "successful attacks".

[9] Here and in what follows we omit the set signs when the support of the arguments are singletons.

obtained by applications of the Defeat rule (or other rules in the table). When an attack rule $\mathcal{R}$ is applied we shall sometimes say that its attacking argument $\mathcal{R}$*-attacks* the attacked argument.

Remark 7. *Clearly, the rules in Table 1 are related. The relations among some of the rules for classical arguments are considered in [111] and [38, Section 5.2]. Figure 2 shows that for any base logic as defined in Definition 1 these relations (together with other relations for ConUcut and BigArgAt) hold also for the more general definition of argument (Definition 5). In this figure, an arrow from $\mathcal{R}_1$ to $\mathcal{R}_2$ means that $\mathcal{R}_1 \subseteq \mathcal{R}_2$.*

$$\begin{tikzcd}
\text{DirUcut} \arrow[r] \arrow[dr] & \text{DirDef} \arrow[r] \arrow[dr] & \text{BigArgAt} \\
\text{ConUcut} \arrow[r] & \text{Ucut} \arrow[r] & \text{Def} \\
\text{Reb} \arrow[r] & \text{DefReb} \arrow[ur, dashed] &
\end{tikzcd}$$

Figure 2: Relations between attack relations from Table 1 (for any base logic). The dashed arrow concerns contrapositive base logics.

C. Argumentation Frameworks. A logical argumentation formalism may be represented as an argumentation framework in the style of Dung [85]. This is defined next.

Definition 8 (logical argumentation framework)**.** Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a logic and $\mathcal{A}$ a set of attack rules with respect to $\mathfrak{L}$. Let also $\mathcal{S}$ be a set of $\mathcal{L}$-formulas. The *(logical) argumentation framework* for $\mathcal{S}$, induced by $\mathfrak{L}$ and $\mathcal{A}$, is the pair $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}) = \langle \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}), \mathit{Attack}(\mathcal{A}) \rangle$, where $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$ is the set of the $\mathfrak{L}$-arguments whose supports are subsets of $\mathcal{S}$, and $\mathit{Attack}(\mathcal{A})$ is a relation on $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}) \times \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$, defined by $(A_1, A_2) \in \mathit{Attack}(\mathcal{A})$ iff there is some $\mathcal{R} \in \mathcal{A}$ such that A_1 $\mathcal{R}$-attacks A_2.

Argumentation frameworks that are induced by classical logic (and some attack rules), and whose arguments are classical (Definition 4), are called *classical (logical) argumentation frameworks*.

In what follows, somewhat abusing the notations, we shall sometimes identify the relation $\mathit{Attack}(\mathcal{A})$ with $\mathcal{A}$. To simplify the notations, we shall also frequently omit the subscripts $\mathfrak{L}$ and $\mathcal{A}$ in $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$, and just write $\mathcal{AF}(\mathcal{S})$.

Example 9. Let $\mathcal{AF}_{\mathsf{CL}}(\mathcal{S}) = \langle \mathsf{Arg}_{\mathsf{CL}}(\mathcal{S}), \mathit{Attack}(\mathcal{A})\rangle$ be a logical argumentation framework for the set $\mathcal{S} = \{p, q, \neg p \vee \neg q, r\}$, based on classical logic (CL), and in which $\mathit{Attack}(\mathcal{A})$ is obtained from the attack rules in $\mathcal{A}$, where $\{\text{ConUcut}\} \subseteq \mathcal{A} \subseteq \{\text{DirDef}, \text{DirUcut}, \text{ConUcut}\}$. The following arguments are in $\mathsf{Arg}_{\mathsf{CL}}(\mathcal{S})$:

$$\begin{array}{ll}
A_1 = \langle r,\ r\rangle & A_7 = \langle \{p, q\},\ p \wedge q\rangle \\
A_2 = \langle p,\ p\rangle & A_8 = \langle \{\neg p \vee \neg q, q\},\ \neg p\rangle \\
A_3 = \langle q,\ q\rangle & A_9 = \langle \{\neg p \vee \neg q, p\},\ \neg q\rangle \\
A_4 = \langle \neg p \vee \neg q,\ \neg p \vee \neg q\rangle & A_\top = \langle \emptyset,\ \neg(p \wedge q \wedge (\neg p \vee \neg q))\rangle \\
A_5 = \langle p,\ \neg((\neg p \vee \neg q) \wedge q)\rangle & A_\bot = \langle \{p, q, \neg p \vee \neg q\},\ \neg r\rangle \\
A_6 = \langle q,\ \neg((\neg p \vee \neg q) \wedge p)\rangle &
\end{array}$$

Figure 3 is a graphical representation of part of the logical argumentation framework with direct defeat and consistency undercut as the attack rules. Here, nodes represent arguments, and directed edges represent attacks (the direction of an edge represents the direction of the attack that it represents).

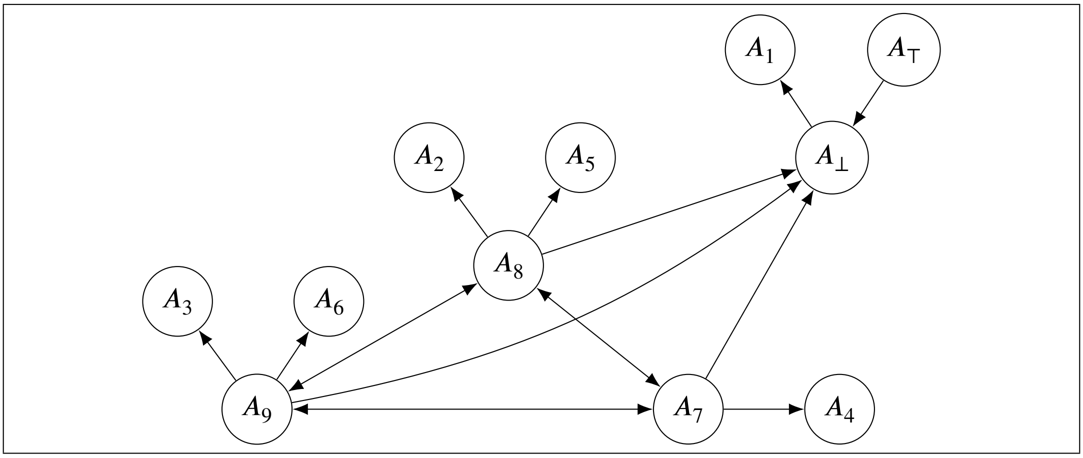


Figure 3: Part of the framework from Example 9.

D. Dung's Semantics. Given an argumentation framework, a key issue in its understanding is the question what combinations of arguments (called *extensions*) can collectively be accepted from this framework. According to Dung [85], this is determined as follows:

Definition 10 (extension-based semantics)**.** *Let $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}), \mathit{Attack}(\mathcal{A})\rangle$ be a logical argumentation framework, and let $\mathcal{E} \cup \{A\} \subseteq \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$. Below, maximality and minimality are taken with respect to the subset relation.*

- *We say that $\mathcal{E}$* attacks *an argument A, if there is an argument $B \in \mathcal{E}$ that attacks A (that is, $(B, A) \in Attack(\mathcal{A})$). The set of arguments in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$ that are attacked by $\mathcal{E}$ (called the* range *of $\mathcal{E}$) is denoted $\mathcal{E}^+$.*
- *We say that $\mathcal{E}$* defends *A, if $\mathcal{E}$ attacks every argument in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$ that attacks A.*
- *The set $\mathcal{E}$ is called* conflict-free *with respect to $\mathcal{AF}(\mathcal{S})$, if it does not attack any of its elements (i.e., $\mathcal{E}^+ \cap \mathcal{E} = \emptyset$). A set that is maximally conflict-free with respect to $\mathcal{AF}(\mathcal{S})$ is called a* naive extension *of $\mathcal{AF}(\mathcal{S})$.*
- *An* admissible extension *of $\mathcal{AF}(\mathcal{S})$ is a subset of $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$ that is conflict-free with respect to $\mathcal{AF}(\mathcal{S})$ and defends all of its elements. A* complete extension *of $\mathcal{AF}(\mathcal{S})$ is an admissible extension of $\mathcal{AF}(\mathcal{S})$ that contains all the arguments that it defends.*
- *The minimal complete extension of $\mathcal{AF}(\mathcal{S})$ is called the* grounded extension *of $\mathcal{AF}(\mathcal{S})$ and a maximal complete extension of $\mathcal{AF}(\mathcal{S})$ is called a* preferred extension *of $\mathcal{AF}(\mathcal{S})$. A complete extension $\mathcal{E}$ of $\mathcal{AF}(\mathcal{S})$ is called a* stable extension *of $\mathcal{AF}(\mathcal{S})$ if $\mathcal{E} \cup \mathcal{E}^+ = \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$.*
- *We will denote with $\mathsf{Naive}(\mathcal{AF}(\mathcal{S}))$ [respectively: $\mathsf{Adm}(\mathcal{AF}(\mathcal{S}))$, $\mathsf{Cmp}(\mathcal{AF}(\mathcal{S}))$, $\mathsf{Prf}(\mathcal{AF}(\mathcal{S}))$, $\mathsf{Stb}(\mathcal{AF}(\mathcal{S}))$] the set of all the naive [respectively: admissible, complete, preferred, stable] extensions of $\mathcal{AF}(\mathcal{S})$ and $\mathsf{Grd}(\mathcal{AF}(\mathcal{S}))$ for the unique grounded extension of $\mathcal{AF}(\mathcal{S})$.*

Remark 11. *In [85], preferred extensions are defined as the maximally admissible sets and stable extensions are the conflict-free extensions whose range consists of all the arguments not in the extension. It is well known that these definitions are equivalent to the ones in Definition 10. Furthermore, stable extensions are preferred (but not necessarily vice-versa), and as is shown in [85, Theorem 25], the grounded extension of an argumentation framework is unique. For more properties of the extensions defined above, further references, and other types of extensions, see, e.g., [24; 22; 23].*

Skeptical and credulous approaches for making inferences from the above-mentioned extensions are defined as follows:

Definition 12 (extension-based entailments)**.** *Let $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}), Attack(\mathcal{A})\rangle$ be a logical argumentation framework, and let $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Cmp}, \mathsf{Grd}, \mathsf{Prf}, \mathsf{Stb}\}$. We denote:*

- *$\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\mathsf{Grd}} \psi$ if there is an argument $\langle \Gamma, \psi\rangle \in \mathsf{Grd}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}))$,*[10] [11]

[10] We make a distinction between the grounded semantics and the other types of semantics, since unlike the other types, the grounded extension is unique (recall Remark 11).

[11] Recall that by the definition of $\mathsf{Grd}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}))$ it holds that $\Gamma \subseteq \mathcal{S}$. The same note holds for the other items in this definition.

- $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Sem}} \psi$ *if there is an argument* $\langle \Gamma, \psi \rangle \in \bigcup \mathsf{Sem}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}))$,
- $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Sem}} \psi$ *if there is an argument* $\langle \Gamma, \psi \rangle \in \bigcap \mathsf{Sem}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}))$,
- $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}} \psi$ *if for every* $\mathcal{E} \in \mathsf{Sem}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}))$ *there is an argument* $\langle \Gamma, \psi \rangle \in \mathcal{E}$.

Example 13. *Consider again the argumentation framework* $\mathcal{AF}_{\mathsf{CL}}(\mathcal{S})$ *from Example 9, where* $\mathcal{S} = \{r, p, q, \neg p \vee \neg q\}$. *In the notations of that example (see also Figure 3), the grounded extension of* $\mathcal{AF}_{\mathsf{CL}}(\mathcal{S})$ *is* $\mathsf{Arg}_{\mathsf{CL}}(\{A_\top, A_1\})$, *and the naive/preferred/stable extensions on* $\mathcal{AF}_{\mathsf{CL}}(\mathcal{S})$ *are* $\mathsf{Arg}_{\mathsf{CL}}(\mathcal{E}_i)$ *(*$i \in \{1, 2, 3\}$*), where:*

- $\mathcal{E}_1 = \{A_\top, A_1, A_2, A_3, A_5, A_6, A_7\}$,
- $\mathcal{E}_2 = \{A_\top, A_1, A_3, A_4, A_6, A_8\}$,
- $\mathcal{E}_3 = \{A_\top, A_1, A_2, A_4, A_5, A_9\}$.

It follows that for every entailment $\mathrel{|\!\sim}$ *considered in Definition 12 we have that* $\mathcal{S} \mathrel{|\!\sim} r$. *The other formulas in* $\mathcal{S}$ *can only be credulously inferred: for every* $\psi \in \mathcal{S} - \{r\}$ *and* $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$ *we have that* $\mathcal{S} \mathrel{|\!\sim}_{\cup\mathsf{Sem}} \psi$, *but* $\mathcal{S} \mathrel{|\!\not\sim}_{\cap\mathsf{Sem}} \psi$, $\mathcal{S} \mathrel{|\!\not\sim}_{\Cap\mathsf{Sem}} \psi$, *and* $\mathcal{S} \mathrel{|\!\not\sim}_{\mathsf{Grd}} \psi$. *Note, moreover, that for instance* $\mathcal{S} \mathrel{|\!\sim}_{\Cap\mathsf{Sem}} p \vee q$ *(but* $\mathcal{S} \mathrel{|\!\not\sim}_{\cap\mathsf{Sem}} p \vee q$*), since at least one of p or q (but not both) follows from each preferred/stable extension, from which* $p \vee q$ *is inferred.*

The next example, taken from [168], demonstrates the usefulness of incorporating modalities for having logic-based argumentative approaches to normative reasoning.

Example 14. *Consider the following example by Horty [129]:*

> *When a meal is served (m), one should not eat with fingers (f). However, if the meal is asparagus (a), one should eat with fingers.*

This scenario may be represented by the deontic logic SDL *(standard deontic logic, i.e., the normal modal logic* KD*), where the modal operator* O *intuitively represents obligations. In this setting, the statements above may be expressed, respectively, by the formulas* $m \supset \mathsf{O}\neg f$ *and* $(m \wedge a) \supset \mathsf{O}f$. *Now, in case that asparagus is indeed served* $(m \wedge a)$ *one expects to derive the (unconditional) obligation to eat with fingers* $(\mathsf{O}f)$ *rather than not to eat with fingers* $(\mathsf{O}\neg f)$.

This is a paradigmatic case of specificity*: a more specific obligation cancels (or overrides) a less specific obligation. An attack rule that reflects this intuition may be expressed as follows:*

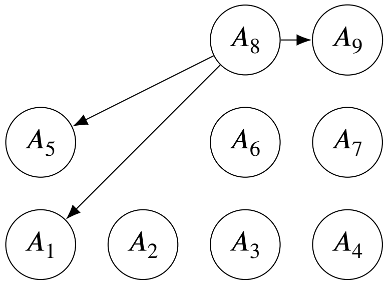
Specificity Undercut (SpecUcut):
$\langle \Gamma \cup \{\phi \supset \mathsf{O}\psi\}, \neg(\phi' \supset \mathsf{O}\psi')\rangle$ *attacks* $\langle \Gamma' \cup \{\phi' \supset \mathsf{O}\psi'\}, \sigma\rangle$ *if the following conditions are met: (i)* $\Gamma \vdash \phi$*, (ii)* $\phi \vdash \phi'$*, and (iii)* $\psi \vdash \neg\psi'$*.*

Condition (i) expresses that the conditional $\phi \supset \mathsf{O}\psi$ *is 'triggered' in view of* Γ*, Condition (ii) expresses that* ϕ *is logically at least as strong as* ϕ' *(i.e., the former is more specific than the latter), and Condition (iii) indicates that the conditionals have conflicting conclusions (after filtering the modalities).*

We thus consider an argumentation framework that is based on the following set:

$$\mathcal{S} = \{m,\ a,\ m \supset \mathsf{O}\neg f,\ (m \wedge a) \supset \mathsf{O}f\}.$$

Some arguments in $\mathsf{Arg}_{\mathsf{SDL}}(\mathcal{S})$ *are listed in Figure 4 (right). Figure 4 (left) shows an attack diagram where the sole attack rule is SpecUcut.*



$A_1 = \langle m \supset \mathsf{O}\neg f, m \supset \mathsf{O}\neg f\rangle$
$A_2 = \langle m, m\rangle$
$A_3 = \langle a, a\rangle$
$A_4 = \langle (m \wedge a) \supset \mathsf{O}f, (m \wedge a) \supset \mathsf{O}f\rangle$
$A_5 = \langle \{m, m \supset \mathsf{O}\neg f\}, \mathsf{O}\neg f\rangle$
$A_6 = \langle \{m, a\}, m \wedge a\rangle$
$A_7 = \langle \{m, a, (m \wedge a) \supset \mathsf{O}f\}, \mathsf{O}f\rangle$
$A_8 = \langle \{m, a, (m \wedge a) \supset \mathsf{O}f\}, \neg(m \supset \mathsf{O}\neg f)\rangle$
$A_9 = \langle \{m, a, m \supset \mathsf{O}\neg f, (m \wedge a) \supset \mathsf{O}f\}, \mathsf{OF}\rangle$

Figure 4: (Part of) the normative argumentation framework of Example 14.

It follows that we have the following expected deductions for every entailment $\mid\!\sim$ *in Definition 12:*

- $\mathcal{S} \not\mid\!\sim \mathsf{O}\neg f$*. Indeed, one cannot derive* $\mathsf{O}\neg f$*, since the application of Modus Ponens to* $m \supset \mathsf{O}\neg f$ *(depicted by argument* A_5*) gets attacked by* A_8*.*
- $\mathcal{S} \mid\!\sim \mathsf{O}f$*. Indeed,* A_7 *is not attacked by an argument in* $\mathsf{Arg}_{\mathsf{SDL}}(\mathcal{S})$*, thus it is part of every grounded, preferred, and stable extension of the underlying normative argumentation framework, and so its descendant follows from* $\mathcal{S}$*. (Note that* A_7 is *attacked by* SDL*-derivable arguments, but none of them is in* $\mathsf{Arg}_{\mathsf{SDL}}(\mathcal{S})$*).*

We refer to [168] for further examples of well-known puzzles, treated by SDL*-based argumentation frameworks.*

Remark 15. *Clearly, whenever a framework $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$ has* Sem*-extensions, it holds that if $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Sem}} \psi$ then $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}} \psi$. Also, if $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}} \psi$ then $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Sem}} \psi$ (thus both types of skeptical reasoning entail credulous reasoning). The converses, however, do not hold. Example 13 shows that for every $\mathsf{Sem} \in \{\mathsf{Prf}, \mathsf{Stb}\}$, $\mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Sem}} \not\subseteq \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Sem}}$, and $\mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Sem}} \not\subseteq \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}}$, and $\mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}} \not\subseteq \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Sem}}$. To see another example for the latter, consider the logical argumentation framework $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}')$, where $\mathcal{S}' = \{p \wedge q, p \wedge \neg q\}$, $\mathfrak{L} = \mathsf{CL}$, and $\mathcal{A} = \{\text{Ucut}\}$. Then $\mathcal{S}' \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Sem}} p$ but $\mathcal{S}' \not\mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Sem}} p$ (because $\bigcap \mathsf{Sem}(\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}'))$ consists only of tautological arguments, i.e., those with empty support sets).*

Proposition 16. *Let $\mathcal{AF}(\mathcal{S})$ be a logical argumentation framework for a finite $\mathcal{S}$, based on a contrapositive logic $\mathfrak{L}$ and the set $\mathcal{A} = \{DirUcut, ConUcut\}$. Then:*

1. $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\mathsf{Grd}} \psi$ *iff* $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Stb}} \psi$.
2. $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Stb}} \psi$.
3. $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Stb}} \psi$.

The above proposition is shown in [10], and some variations of it are proved in [11]. As mentioned there, the assumptions on the logic and the attack rules are essential for the proposition to hold.

2.2.2 The ASPIC System

ASPIC$^+$ [150; 145] is another well-known approach to structured argumentation, based on some underlying logic. It contains (at least) two types of premises: axioms (which cannot be questioned) and ordinary premises (which can be questioned/attacked). Also, there are two types of rules: strict and defeasible. The latter, unlike strict rules, allow for exceptions. A wide variety of research has been done on ASPIC$^+$, both from a theoretical perspective (e.g., rationality postulates were introduced in [60] for ASPIC, an earlier version of ASPIC$^+$, and the use of preferences has been investigated in [145]) and from an application perspective (See [147, Section 6] for an overview). We refer to [146; 147] for extensive surveys on ASPIC$^+$ and related approaches. Unless otherwise stated, the definitions in this section are taken from [147] (the chapter on ASPIC$^+$ in the first volume of the handbook).

Remark 17. *As noted in Remark 2, we only discuss purely logical instances of logical argumentation frameworks. For ASPIC$^+$ this means that we do not take into account any ordering over the defeasible elements.*

Definition 18 (ASPIC-based argumentation system)**.** *An* argumentation system *is a tuple $AS = \langle \mathcal{L}, ^{-}, \mathcal{R}, n \rangle$, where:*

- *$\mathcal{L}$ is a propositional language,*
- *$^{-}$ is a contrariness function from $\mathcal{L}$ to $2^{\mathcal{L}} \setminus \emptyset$,* [12]
- *$\mathcal{R} = \langle \mathcal{R}_s, \mathcal{R}_d \rangle$ consists of strict ($\mathcal{R}_s$) and defeasible ($\mathcal{R}_d$) inference rules of the form $\phi_1, \ldots, \phi_n \rightarrow \phi$ and $\phi_1, \ldots, \phi_n \Rightarrow \phi$ respectively, such that $\mathcal{R}_s \cap \mathcal{R}_d = \emptyset$,*
- *$n : \mathcal{R}_d \rightarrow \mathsf{WFF}(\mathcal{L})$ is a (possibly partial) function assigning names to defeasible rules.*

The contrariness function allows to specify conflicts between elements of the language. Strict rules are deductive in the sense that the truth of their premises $\phi_1, \ldots, \phi_n$ necessarily implies the truth of their antecedent ϕ. Unlike strict rules, a defeasible rule warrants the truth of its conclusion only provisionally: its application can be retracted in case counterarguments are encountered. A naming function associates a name $n(r)$ with some of the defeasible rules in $\mathcal{R}_d$. This will facilitate the formulation of the attack form *undercut* (see below).

Definition 19 (ASPIC theory)**.** *A* knowledge-base *in an argumentation system $AS = \langle \mathcal{L}, ^{-}, \mathcal{R}, n \rangle$ is a pair $\mathcal{K} = \langle \mathcal{K}_n, \mathcal{K}_p \rangle$ of $\mathcal{L}$-formulas that consists of two disjoint sets: $\mathcal{K}_n$ (the axioms) and $\mathcal{K}_p$ (the ordinary premises). An* ASPIC argumentation theory *is a pair $AT = \langle AS, \mathcal{K} \rangle$, where AS is an argumentation system and $\mathcal{K}$ is a knowledge-base in AS.*

Arguments in ASPIC$^+$ differ from arguments in logic-based argumentation frameworks. These are *inference trees* that are constructed from the rules of the argumentation system and the formulas in the knowledge base:

Definition 20 (ASPIC argument)**.** *An* ASPIC-argument *A on the basis of an ASPIC-theory AT is of one of the following forms:*

1. *ϕ, if $\phi \in \mathcal{K}_n \cup \mathcal{K}_p$. In this case we denote:*

   $\mathsf{Prem}(A) = \{\phi\}$;
   $\mathsf{Conc}(A) = \phi$;
   $\mathsf{Sub}(A) = \{\phi\}$;
   $\mathsf{Rules}(A) = \mathsf{DefRules}(A) = \mathsf{TopRules}(A) = \emptyset$.

2. *$A_1, \ldots, A_n \rightarrow \psi$, if $A_1, \ldots, A_n$ are ASPIC-arguments such that there exists a strict rule of the form $\mathsf{Conc}(A_1), \ldots, \mathsf{Conc}(A_n) \rightarrow \psi$ in $\mathcal{R}_s$. In this case we denote:*

   $\mathsf{Prem}(A) = \mathsf{Prem}(A_1) \cup \ldots \cup \mathsf{Prem}(A_n)$;
   $\mathsf{Conc}(A) = \psi$;

[12] In many publications, a distinction is made between *contraries* and *contradictories*. This distinction mainly plays a role when preferences over defeasible rules are taken into account and therefore is left out of this survey.

$\mathit{Sub}(A) = \mathit{Sub}(A_1) \cup \ldots \cup \mathit{Sub}(A_n) \cup \{A\}$;
$\mathit{Rules}(A) = \mathit{Rules}(A_1) \cup \ldots \cup \mathit{Rules}(A_n) \cup \{\mathit{Conc}(A_1), \ldots, \mathit{Conc}(A_n) \to \psi\}$;
$\mathit{TopRules}(A) = \bigcup_{B \in \mathsf{Sub}(A)} \mathit{TopRules}(B)$;
$\mathit{DefRules}(A) = \{r \in \mathcal{R}_d \mid r \in \mathit{Rules}(A)\}$.

3. $A_1, \ldots, A_n \Rightarrow \psi$*, if* $A_1, \ldots, A_n$ *are ASPIC-arguments such that there exists a defeasible rule of the form* $\mathit{Conc}(A_1), \ldots, \mathit{Conc}(A_n) \Rightarrow \psi$ *in* $\mathcal{R}_d$*. In this case we denote:*

   $\mathit{Prem}(A) = \mathit{Prem}(A_1) \cup \ldots \cup \mathit{Prem}(A_n)$;
   $\mathit{Conc}(A) = \psi$;
   $\mathit{Sub}(A) = \mathit{Sub}(A_1) \cup \ldots \cup \mathit{Sub}(A_n) \cup \{A\}$;
   $\mathit{Rules}(A) = \mathit{Rules}(A_1) \cup \ldots \cup \mathit{Rules}(A_n) \cup \{\mathit{Conc}(A_1), \ldots, \mathit{Conc}(A_n) \Rightarrow \psi\}$;
   $\mathit{TopRules}(A) = \{\mathit{Conc}(A_1), \ldots, \mathit{Conc}(A_n) \Rightarrow \psi\}$;
   $\mathit{DefRules}(A) = \{r \in \mathcal{R}_d \mid r \in \mathit{Rules}(A)\}$.

We denote the set of arguments that can be constructed from an argumentation theory $AT = \langle AS, \mathcal{K}\rangle$ *by* $\mathsf{Arg}(AT)$.

Example 21. *Let* $AS = \langle \mathcal{L}, \overline{}, \mathcal{R}, n\rangle$ *be an argumentation system, where* $\mathcal{L}$ *is a standard propositional language with* $\mathsf{Atoms}(\mathcal{L}) = \{p, q, r, n(r_1)\}$, $\overline{\phi} = \{\psi \mid \psi \equiv \neg\phi\}$ *for any* $\mathcal{L}$*-formula* ϕ*, the rules in* $\mathcal{R}_s$ *coincide with those of classical logic in the sense that* $\phi_1, \ldots, \phi_n \to \phi \in \mathcal{R}_s$ *iff* $\{\phi_1, \ldots, \phi_n\} \vdash_{\mathsf{CL}} \phi$ *for* $\mathcal{L}$*-formulas* $\phi_1, \ldots, \phi_n, \phi$*, and*

$$\mathcal{R}_d = \big\{r_1 : p \Rightarrow \neg q;\;\; r_2 : q \Rightarrow \neg n(r_1)\big\}, \quad \mathcal{K}_p = \{p, q, r\}, \quad \mathcal{K}_n = \emptyset$$

Among others, the following ASPIC-arguments can be constructed:

$$\begin{array}{lll} A_1 : r & A_4 : A_2 \Rightarrow \neg q & A_7 : A_2, A_4 \to p \wedge \neg q \\ A_2 : p & A_5 : A_3 \Rightarrow \neg n(r_1) & A_8 : A_3, A_4 \to \neg r \\ A_3 : q & A_6 : A_2, A_3 \to p \wedge q & A_9 : A_3, A_4 \to \neg p \end{array}$$

In ASPIC$^+$ arguments can be attacked on their defeasible rules (undercut), on conclusions of sub-arguments whose top-rule is defeasible (rebuttal) and on their ordinary premises (undermine attack):

Definition 22 (ASPIC-attack)**.** *An ASPIC-argument A* attacks *an ASPIC-argument B iff A* undercuts, rebuts *or* undermines *B, where:*

- *A* undercuts *B (on B′) iff* $\mathit{Conc}(A) \in \overline{n(\mathsf{Conc}(B_1), \ldots, \mathsf{Conc}(B_n) \Rightarrow \phi)}$ *for some* $B' \in \mathit{Sub}(B)$ *of the form* $B_1, \ldots, B_n \Rightarrow \phi$;
- *A* rebuts *B (on B′) iff* $\mathit{Conc}(A) \in \overline{\phi}$ *for some* $B' \in \mathit{Sub}(B)$ *of the form* $B''_1, \ldots, B''_n \Rightarrow \phi$.

- A undermines B *(on B') iff* $\mathit{Conc}(A) \in \overline{\phi}$ *for some* $B' = \phi$*, for some* $\phi \in \mathit{Prem}(B) \cap \mathcal{K}_p$.

Remark 23. *Note that attacks in ASPIC$^+$ always target defeasible elements of the attacked argument: undercuts attack a defeasible rule (for this the naming function was instrumental), rebuts always attack in the head of a defeasible rule, and undermining always targets defeasible premises. Also note the difference in terminology to logic-based argumentation: the* undercut *attack in the context of ASPIC$^+$ is quite different from the undercut attack for logic-based argumentation (see Table 1). The latter resembles more* undermining*-attacks in the context of ASPIC$^+$.*

Now, Dung-style argumentation frameworks are defined in ASPIC$^+$ as follows:

Definition 24 (ASPIC argumentation framework)**.** *Let $AT = \langle AS, \mathcal{K}\rangle$ be an ASPIC argumentation theory. An* (ASPIC) argumentation framework*, defined by AT, is a pair $\mathcal{AF}(AT) = \langle \mathsf{Arg}(AT), \mathit{Attack}\rangle$, where:*

- $\mathsf{Arg}(AT)$ *is the set of ASPIC-arguments constructed from AT, as in Definition 20; and*
- $(X, Y) \in \mathit{Attack}$ *iff X attacks Y, as in Definition 22.*[13]

Example 25 (Example 21 continued)**.** *In the argumentation theory from Example 21, we have that:*

- A_5 *undercuts* A_4*,* A_7*,* A_8 *and* A_9 *(all of them on* A_4*),*
- A_4 *undermines* A_3*,* A_5*,* A_6*,* A_8 *and* A_9 *(all on* A_3*),*
- A_3 *rebuts* A_4*,* A_7*,* A_8 *and* A_9 *(all on* A_4*).*

There are more attacks between $A_1, \ldots, A_9$ besides the ones listed here: the full attack relation between these arguments is shown in Figure 5.

Dung-style semantics, as defined in Definition 10, can now be applied to the frameworks defined above as well. For example, given $\mathcal{AF}(\mathrm{AT}) = \langle \mathsf{Arg}(\mathrm{AT}), \mathit{Attack}\rangle$, $\mathcal{E} \subseteq \mathsf{Arg}(\mathrm{AT})$ is an admissible extension of $\mathcal{AF}(\mathrm{AT})$ if it is conflict-free with respect to $\mathcal{AF}(\mathrm{AT})$ and defends all of its elements. Similarly, $\mathcal{E}$ is a complete extension of $\mathcal{AF}(\mathrm{AT})$ if it is an admissible extension of $\mathcal{AF}(\mathrm{AT})$ that contains all the arguments it defends. Like before, we will denote by $\mathsf{Sem}(\mathcal{AF}(\mathrm{AT}))$ all the Sem-extensions of $\mathcal{AF}(\mathrm{AT})$, for $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Adm}, \mathsf{Cmp}, \mathsf{Grd}, \mathsf{Prf}, \mathsf{Stb}\}$.

The next definition is a counterpart, for the ASPIC$^+$ system, of Definition 12:

[13] Note that, unlike logic-based argumentation, where frameworks may differ in their attack rules, in ASPIC systems always *all* the possible attack rules are applied.

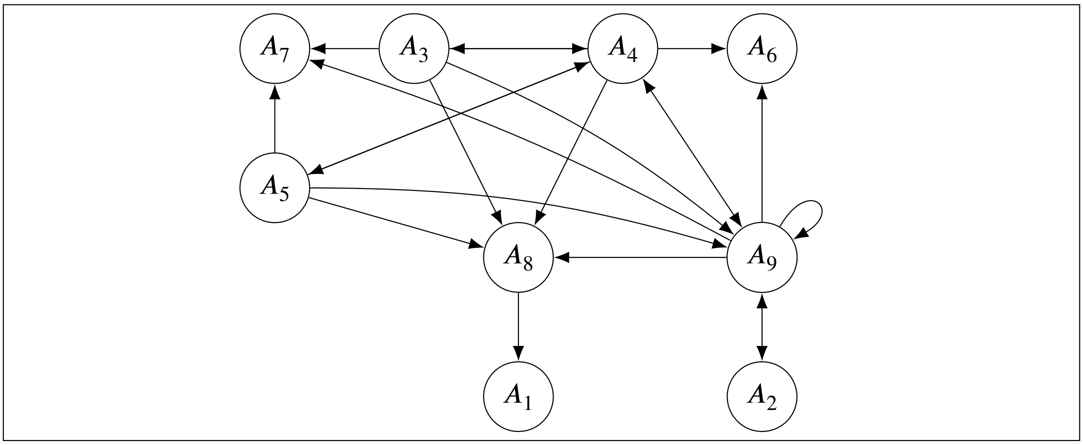


Figure 5: Part of the framework from Example 25.

Definition 26 (ASPIC extension-based entailments)**.** *Let $\mathcal{AF}(AT) = \langle \mathsf{Arg}(AT), Attack\rangle$ be an argumentation framework for some argumentation theory AT and let* $\mathsf{Sem} \in \{\mathsf{Grd}, \mathsf{Cmp}, \mathsf{Prf}, \mathsf{Stb}, \mathsf{Naive}\}$. *Then:*

- *$AT \mid\!\sim_{\cup\mathsf{Sem}} \psi$ if there is an argument $A \in \bigcup \mathsf{Sem}(\mathcal{AF}(AT))$ with $\mathsf{Conc}(A) = \psi$. In this case it is said that ψ is* credulously justified*;*
- *$AT \mid\!\sim_{\cap\mathsf{Sem}} \psi$ if there is an argument $A \in \bigcap \mathsf{Sem}(\mathcal{AF}(AT))$ with $\mathsf{Conc}(A) = \psi$. In this case it is said that ψ is* skeptically justified*;*
- *$AT \mid\!\sim_{\Cap\mathsf{Sem}} \psi$ if for every $\mathcal{E} \in \mathsf{Sem}(\mathcal{AF}(AT))$ there is an argument $A \in \mathcal{E}$ with $Conc(A) = \psi$. In this case it is said that ψ is* weakly skeptically justified*.*

As any Dung-style argumentation framework has a single grounded extension, the entailments $\mid\!\sim_{\cap\mathsf{Grd}}$, $\mid\!\sim_{\cup\mathsf{Grd}}$ and $\mid\!\sim_{\Cap\mathsf{Grd}}$ coincide, we will therefore sometimes omit the initial symbol from the subscript.

Remark 27. *Unlike standard consequence relations (Definition 1) and the extension-based entailments for the logic-based approach (Definition 12), which are relations between sets of formulas and formulas, the entailments above are relations between argumentation theories and formulas. This will not cause any confusion in what follows.*

Example 28 (Example 25 continued)**.** *In the argumentation framework from Example 25 shown in Figure 5, for the ASPIC argumentation theory AT from Example 21, we have that $Grd(\mathcal{AF}(AT)) = \emptyset$.*[14] *It is easy to see that there are two preferred extensions for this framework: one contains (among others) the arguments A_1, A_2, A_4 and A_7 and the other contains*

[14] Recall that we identify $\mathsf{Grd}(\mathcal{AF}(\mathrm{AT}))$ with its single set.

(among others) A_1, A_2, A_3, A_5 and A_6. Therefore, the following conclusions can be derived for $\mathsf{Sem} = \mathsf{Prf}$:

- *$AT \mid\!\sim_{\cap \mathsf{Prf}} \phi$ iff $\phi \in Cn(r \wedge p)$, since A_1 and A_2 occur in each preferred extension;*
- *$AT \mid\!\sim_{\Cap \mathsf{Prf}} \neg q \vee (\neg n(r_1) \wedge q)$ since A_4 occurs in one preferred extension and A_5 and A_3 in the other preferred extension;*
- *$AT \mid\!\sim_{\cup \mathsf{Prf}} \phi$ for $\phi \in \{p, \neg q, q\}$ (among others), since each of the arguments besides A_8 and A_9 from Example 21 is part of at least one preferred extension.*

Remark 29. *A similar result as that of Proposition 16 in the previous section is not available for ASPIC systems, since in the presence of odd attack cycles some preferred extensions may not attack all arguments in their complement (and therefore might not be stable). This can also lead to settings in which no stable extension exist. This is demonstrated in the next example.*

Example 30. *As in our previous example, let $\mathcal{R}_s$ be instantiated by classical logic. Let also $\overline{\phi} = \{\neg\phi\}$ for every formula ϕ, $\mathcal{K} = \langle \emptyset, \emptyset \rangle$, and let $\mathcal{R}_d$ consist of the following three rules: $r_1 : \Rightarrow \neg n(r_2)$, $r_2 : \Rightarrow \neg n(r_3)$, $r_3 : \Rightarrow \neg n(r_1)$. Note that, for instance, the arguments*

$$A_1 : \Rightarrow \neg n(r_2), \quad A_2 : \Rightarrow \neg n(r_3), \quad A_3 : \Rightarrow \neg n(r_1)$$

are involved in an odd attack cycle (of length 3). As a consequence, neither of the three arguments can be part of an admissible extension. Thus, the only preferred extension will consist of all strict arguments (which conclude classical theorems). Clearly, this extension will not be able to attack the three arguments above, and thus it is not stable.

We note, nevertheless, that there are instances of ASPIC$^+$ for which a similar result to that of Proposition 16 is available. This is especially the case when ASPIC$^+$ is instantiated by a contrapositive strict rule base, when the contrariness operator is defined by the negation of the language and no undercutting arguments can be generated from the knowledge base. See further discussions in Sections 2.3.1 and 2.4.

2.2.3 Assumption-Based Argumentation

Assumption-based argumentation (ABA, [46]) is another prominent formalism for logical argumentation. It was introduced in the 1990s as a computational framework to capture and generalize default and defeasible reasoning, inspired by Dung's semantics for abstract argumentation and by logic programming with its dialectical interpretation of the acceptability of negation-as-failure assumptions based on "no-evidence-to-the-contrary". In this section we recall the basic definitions that are related to this approach. For extensive surveys on ABA and related approaches, we refer to [87; 171; 72; 73]. ABA-based implementations are surveyed in [69, Section 3.2].

Definition 31 (assumption-based framework)**.** *An* assumption-based framework *(in short: ABF) is a tuple $\mathcal{ABF} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \sim \rangle$ where:*

- *$\mathcal{L}$ is a (propositional) language,*
- *$\mathcal{R}$ is a set of* strict rules*, whose elements are of the form $\psi_1, \ldots, \psi_n \to \psi$, where ψ, ψ_i $(1 \leq i \leq n)$ are $\mathcal{L}$-formulas,*
- *$\mathcal{A}$ is a nonempty set of $\mathcal{L}$-formulas, called the* defeasible (or candidate) assumptions*, and*
- *$\sim : \mathcal{A} \to \wp(\mathcal{L})$ is a* contrariness operator*, assigning a finite set of $\mathcal{L}$-formulas to every defeasible assumption in $\mathcal{L}$.*[15]

Somewhat like the rules in ASPIC, rules in ABFs can be chained to form *deductions*. Given a set $\mathcal{S} \subseteq \mathcal{A}$ of defeasible assumptions, an $\mathcal{S}$-based deduction may be viewed as a proof, i.e., a sequence of $\mathcal{L}$-formulas, where each element of the sequence is either a formula in $\mathcal{S}$ or is obtained from previous elements in the sequence by an application of a rule $\mathcal{R}$, just like an application of Modus Ponens.

Definition 32 ($\vdash_{\mathcal{R}}$)**.** *Let $\mathcal{R}$ be a set of inference rules over $\mathcal{L}$. We write $\mathcal{S} \vdash_{\mathcal{R}} \psi$ if there is an $\mathcal{S}$-deduction, based on the rules in $\mathcal{R}$, that culminates in ψ, i.e., there is a sequence $\phi_1, \ldots, \phi_n$ of $\mathcal{L}$-formulas such that $\phi_n = \psi$ and for each $1 \leq i \leq n$, $\phi_i \in \mathcal{S}$ or there are $\phi_{i_1}, \ldots, \phi_{i_m}$ for which $i_1, \ldots, i_m < i$ and $\phi_{i_1}, \ldots, \phi_{i_m} \to \phi_i \in \mathcal{R}$.*

For instance, if $p \to q \in \mathcal{R}$, then $p \vdash_{\mathcal{R}} q$.

As in logic-based argumentation and ASPIC, (defeasible) assertions in an ABF may be attacked in the presence of counter (defeasible) information. This is described in the next definition.

Definition 33 (attacks in ABFs)**.** Let $\mathcal{ABF} = \langle \mathsf{Atoms}(\mathcal{L}), \mathcal{R}, \mathcal{A}, \sim \rangle$ be an assumption-based framework, and let $\mathcal{S}, \mathcal{T} \subseteq \mathcal{A}$, $\psi \in \mathcal{A}$. We say that $\mathcal{S}$ *attacks* ψ if there are $\mathcal{S}' \subseteq \mathcal{S}$ and $\phi \in {\sim}\psi$ such that $\mathcal{S}' \vdash_{\mathcal{R}} \phi$. Accordingly, $\mathcal{S}$ attacks $\mathcal{T}$ if $\mathcal{S}$ attacks some $\psi \in \mathcal{T}$.

[15] Note that the contrariness operator is not a connective of $\mathcal{L}$, as it is restricted only to the candidate assumptions.

Remark 34. *In contrast to most of the logical argumentation frameworks defined in the preceding sections (as well as other approaches to structured argumentation, such as DeLP [106]), in which attacks are defined between individual arguments, in ABA systems attacks are defined between sets of assumptions. This may be viewed as a higher level of abstraction, operating on equivalence classes that consist of arguments generated from the same assumptions.*

Using the above notion of attack, Dung-style semantics is defined on ABFs just as in Definition 10. The only difference is that an extension $\mathcal{E}$ in an ABF is required to be *closed* with respect to the rules in $\mathcal{R}$, namely: $\mathcal{E} = Cn_{\vdash_{\mathcal{R}}}(\mathcal{E}) \cap \mathcal{A}$. Thus, for instance, for $\mathcal{S} \subseteq \mathcal{A}$ we say that

- $\mathcal{S}$ is *conflict-free* (with respect to $\mathcal{ABF}$) iff $\mathcal{S}$ does not attack itself.
- $\mathcal{S}$ *defends* (with respect to $\mathcal{ABF}$) a set $\mathcal{S}' \subseteq \mathcal{A}$ iff for every closed set $\mathcal{S}^\star$ that attacks $\mathcal{S}'$, $\mathcal{S}$ attacks $\mathcal{S}^\star$.
- $\mathcal{S}$ is *admissible* (with respect to $\mathcal{ABF}$) iff it is closed, conflict-free, and defends itself. An admissible set is called *complete*, if it does not defend any of its proper supersets.
- $\mathcal{S}$ is *stable* (with respect to $\mathcal{ABF}$) iff it is closed, conflict-free and attacks every $\phi \in \mathcal{A} \setminus \mathcal{S}$.

In ABA it is usual to refer also to the intersection of all the complete extensions of an ABF, which is called the *well-founded* extension of that ABF.

Like before, we denote by $\mathsf{Naive}(\mathcal{ABF})$ [respectively: $\mathsf{Adm}(\mathcal{ABF})$, $\mathsf{Cmp}(\mathcal{ABF})$, $\mathsf{Grd}(\mathcal{ABF})$, $\mathsf{Prf}(\mathcal{ABF})$, $\mathsf{Stb}(\mathcal{ABF})$, $\mathsf{WF}(\mathcal{ABF})$] the set of all the naive [respectively: admissible, complete, grounded, preferred, stable, well-founded] extensions of $\mathcal{ABF}$.[16]

If every set of assumptions $\mathcal{S} \subseteq \mathcal{A}$ is $\vdash_{\mathcal{R}}$-closed, the ABF is called*flat*. In [46] it is shown that most of the relations between the Dung extensions considered in Remark 11 carry on to flat ABFs (see also [73, Theorems 2.12 and 2.14], and [126] for prioritized settings). For non-flat ABFs, however, some of these relations cease to hold. For instance, there may be non-flat ABFs without complete extensions (cf. Item 2 of Proposition 38).

The following form of ABFs is considered in [117; 119; 121]:

Definition 35 (simple contrapositive ABFs)**.** *A* contrapositive assumption-based framework *is a tuple* $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, \sim \rangle$ *where:*

[16] Note that, as observed in [121], the grounded extension of an ABF may not be unique, thus (unlike the previous cases) this time $\mathsf{Grd}(\mathcal{ABF})$ is not an extension but a set of extensions.

- $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ *is an explosive and contrapositive logic,*[17]
- Γ *(the* strict assumptions*) and* Δ *(the* candidate/defeasible assumptions*) are distinct (countable) sets of* $\mathcal{L}$*-formulas, where the former is assumed to be* $\vdash$*-consistent and the latter is assumed to be nonempty,*
- $\sim \, : \, \Delta \to \wp(\mathcal{L})$ *is a* contrariness operator*, assigning a finite set of* $\mathcal{L}$*-formulas to every defeasible assumption in* Δ*, such that for every* $\vdash$*-consistent* $\psi \in \Delta$ *it holds that* $\psi \nvdash \bigwedge{\sim}\psi$ *and* $\bigwedge{\sim}\psi \nvdash \psi$*.*

A contrapositive ABF is called simple*, if its language* $\mathcal{L}$ *contains a negation* $\neg$*, and for every* $\psi \in \mathcal{A}$*,* ${\sim}\psi = \{\neg\psi\}$*.*

Given a simple contrapositive assumption-based framework $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, \sim \rangle$, the notion of attack and Dung-style semantics are defined as before, with the obvious adjustments using the consequence relation $\vdash$ of the base logic instead of the entailment $\vdash_{\mathcal{R}}$. For instance,

- $\mathcal{S} \subseteq \Delta$ *attacks* $\psi \in \Delta$ iff $\Gamma, \mathcal{S} \vdash \phi$ for some $\phi \in {\sim}\psi$. Accordingly, $\mathcal{S}$ attacks $\mathcal{T}$ if $\mathcal{S}$ attacks some $\psi \in \mathcal{T}$,
- $\mathcal{S} \subseteq \Delta$ is *closed* in $\mathcal{ABF}$ if $\mathcal{S} = \Delta \cap Cn_{\vdash}(\Gamma \cup \mathcal{S})$.

The other semantic notions remain exactly as before.

Given a (simple, contrapositive) assumption-based framework $\mathcal{ABF}$ and $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{WF}, \mathsf{Grd}, \mathsf{Prf}, \mathsf{Stb}\}$, we denote:

Definition 36 (ABA extension-based entailments)**.**

- $\mathcal{ABF} \mathrel{|\!\sim}_{\cup\mathsf{Sem}} \psi$ *iff* $\Gamma, \mathcal{E} \vdash \psi$ *for* some $\mathcal{E} \in \mathsf{Sem}(\mathbf{ABF})$.
- $\mathcal{ABF} \mathrel{|\!\sim}_{\cap\mathsf{Sem}} \psi$ *iff* $\Gamma, \bigcap \mathsf{Sem}(\mathbf{ABF}) \vdash \psi$.
- $\mathcal{ABF} \mathrel{|\!\sim}_{\Cap\mathsf{Sem}} \psi$ *iff* $\Gamma, \mathcal{E} \vdash \psi$ *for* every $\mathcal{E} \in \mathsf{Sem}(\mathbf{ABF})$.

The entailment relations in Definition 36 are again different from those in Definitions 1 and 12, as they are defined on ABFs and formulas (cf. Remark 27). Like before, this will not cause any confusion in the sequel.

Example 37. *Let* $\mathfrak{L} = \mathsf{CL}$*,* $\Gamma = \emptyset$*,* $\Delta = \{p, \neg p, q\}$*, and* ${\sim}\psi = \{\neg\psi\}$ *for every formula* ψ*. A corresponding attack diagram is shown in Figure 6.*[18]

[17] Classical logic CL, intuitionistic logic, the central logic in the family of constructive logics, and standard modal logics are all explosive and contrapositive logics.

[18] For reasons that will become apparent in the sequel (see Remark 41), we include in the diagram only *closed sets*. Thus, the set $\{p, \neg p\}$ is omitted from the diagram.

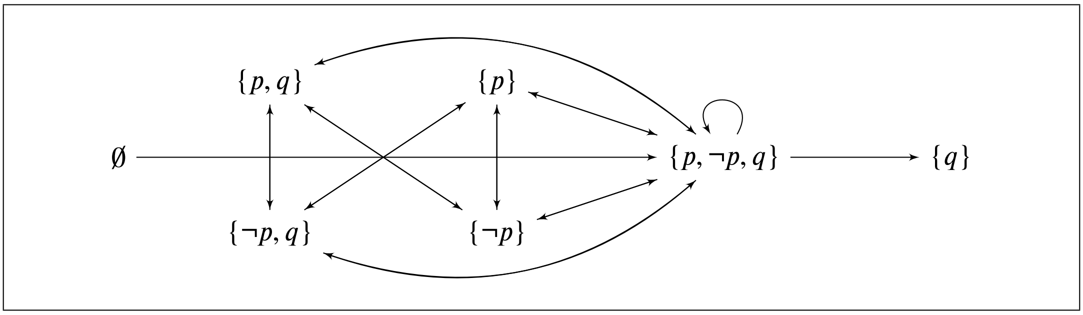


Figure 6: An attack diagram for Example 37

Here, $\mathsf{Naive}(\mathcal{ABF}) = \mathsf{Prf}(\mathcal{ABF}) = \mathsf{Stb}(\mathcal{ABF}) = \{\{p,q\},\{\neg p,q\}\}$*, and therefore* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{Sem}} q$ *for every* $\circ \in \{\cup,\cap,\Cap\}$ *and* $\mathsf{Sem} \in \{\mathsf{Naive},\mathsf{Prf},\mathsf{Stb}\}$.

Some interesting properties of simple contrapositive ABFs are given next (see [117; 119; 121]).

Proposition 38. *Let* $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, \neg \rangle$ *be a simple contrapositive ABF. Then:*

1. $\mathsf{Naive}(\mathcal{ABF}) = \mathsf{Prf}(\mathcal{ABF}) = \mathsf{Stb}(\mathcal{ABF})$.
2. *If* $\mathsf{F} \in \Delta$ *then* $\mathsf{Grd}(\mathcal{ABF}) = \mathsf{WF}(\mathcal{ABF})$.

The next example shows that the condition in Item 2 of the last proposition is indeed necessary:

Example 39. Let $\mathfrak{L}$ be an explosive logic, $\Delta = \{p,\neg p,q\}$ and $\Gamma = \{s, s \supset q\}$. Note that the emptyset is not admissible, since it is not closed (indeed, $\Gamma \vdash q$). Also, $\{q\}$ is not admissible since $p,\neg p,q \vdash \neg q$.[19] The two minimal complete extensions here are $\{p,q\}$ and $\{\neg p,q\}$, thus there is no *unique* grounded extension in this case.

Corollary 40. *Let* $\mathcal{ABF}$ *be a simple contrapositive ABF, and let* $\circ \in \{\Cap,\cup,\cap\}$*. Then for every* ψ *we have that:* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{Naive}} \psi$ *iff* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{Prf}} \psi$ *iff* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{Stb}} \psi$*. Moreover, if* $\mathsf{F} \in \Delta$ *then* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{Grd}} \psi$ *iff* $\mathcal{ABF} \mathrel{|\!\sim}_{\circ\mathsf{WF}} \psi$*.*

Remark 41. *Interestingly, as shown in [117], the closure requirement is redundant in the definition of extensions of simple contrapositive ABFs. Thus, for instance, if* $\mathcal{E} \subseteq \Delta$ *is conflict-free and attacks every* $\psi \in \Delta \setminus \mathcal{E}$ *then it is closed (so closure is assured in the definition of stable extensions), a maximally conflict-free subset of* Δ *is closed (thus closure is guaranteed in the definition of naive extensions), and so forth. For grounded and well-founded semantics, the closure requirement is redundant only if* $\mathsf{F} \in \Delta$.

[19] Note that q is also attacked by $\{p,\neg p\}$ and does not counterattack it. However, $\{p,\neg p\}$ is not closed, and for admissibility checking it is enough to consider only closed sets (see also Remark 41).

Remark 42. *In [126] other classes of ABFs are studied. It is shown there that also for so-called* well-behaved *ABFs, the preferred and stable extension coincide. Well-behaved ABFs are flat ABFs that satisfy a slightly weaker notion of contraposition than the one above, and a property called* sanity *that says that if* $\sim\phi$ *follows from a set of assumptions* Δ *then it follows from* $\Delta \setminus \{\phi\}$ *(which is also satisfied by contrapositive ABFs). Otherwise, no restrictions on the underlying language are imposed.*[20]

2.3 Properties of the Frameworks and Their Entailments

In order to evaluate and compare the various approaches to logical argumentation, different properties and postulates have been introduced in the literature. In this section we consider the three logical argumentation methods of Section 2.2 in light of these criteria. We do so from three perspectives:

- relations to reasoning with maximal consistency, following [155] (Section 2.3.1),
- rationality postulates for argumentative reasoning, following [60] (Section 2.3.2), and
- inference principles for non-monotonic reasoning, following [133] (Section 2.3.3).

In what follows we review the main results in the literature concerning the above-mentioned issues. We recall that it is not the purpose of this survey to resolve open questions or particular cases that were not addressed so far,[21] thus we do not pretend to have an exhaustive coverage of the subject.

2.3.1 Relations to Reasoning with Maximal Consistency

Reasoning with maximally consistent subsets (MCS), introduced in [155], is a well-known approach to handle inconsistencies within non-monotonic reasoning. The idea is to derive conclusions from inconsistent knowledge-bases, by considering the maximally consistent subsets of these knowledge bases. This idea has been applied in a variety of research directions within artificial intelligence, e.g.: knowledge-based integration systems [21], consistency operators for belief revision [131] and computational linguistics [140].

The relation between reasoning with maximally consistent subsets and formal argumentation has been studied extensively since this possibility was raised in [67]. In what follows

[20] For technical details we refer to the paper whose main focus is to study and compare systems of prioritized ABFs.

[21] The only exception are the (yet unpublished) results in the appendix of the chapter, which appear in a paper that is currently under review.

we survey some of the main results relating MCS-based reasoning and the logic-based methods of the previous section. For a more extensive overview of the subject we refer to [11; 10].

Reasoning with maximally consistent subsets of the premises is based on the following definition:

Definition 43 ($\mathsf{MCS}_{\mathfrak{L}}(\mathcal{S}), \mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S})$)**.** *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a logic and let $\mathcal{S}', \mathcal{S}$ be sets of $\mathcal{L}$-formulas (intuitively, $\mathcal{S}'$ are the strict assumptions and $\mathcal{S}$ are the defeasible ones).*

- $\mathsf{MCS}_{\mathfrak{L}}(\mathcal{S})$ *is the set of the maximally $\vdash$-consistent subsets of $\mathcal{S}$. I.e.,*

  $\mathsf{MCS}_{\mathfrak{L}}(\mathcal{S}) = \{\mathcal{T} \subseteq \mathcal{S} \mid \mathcal{T}$ *is $\vdash$-consistent and for every $\mathcal{T}'$ such that $\mathcal{T} \subsetneq \mathcal{T}' \subseteq \mathcal{S}$, $\mathcal{T}'$ is $\vdash$-inconsistent*$\}$.

- $\mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S})$ *is the set of the maximally $\vdash$-consistent subsets of $\mathcal{S}$, given $\mathcal{S}'$. I.e.,*

  $\mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S}) = \{\mathcal{T} \subseteq \mathcal{S} \mid \mathcal{T} \cup \mathcal{S}'$ *is $\vdash$-consistent and for every $\mathcal{T}'$ such that $\mathcal{T} \subsetneq \mathcal{T}' \subseteq \mathcal{S}$, $\mathcal{T}' \cup \mathcal{S}'$ is $\vdash$-inconsistent*$\}$.

The second item in the definition above, which defines maximally consistent subsets w.r.t. a set of strict assumptions, is known from [138] as *default assumptions*. Some of the corresponding entailment relations are defined in [138] as well, which is similar to those in Definitions 12, 26 and 36:

Definition 44 (MCS-based entailments)**.** *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a logic and let $\mathcal{S}', \mathcal{S}$ be sets of $\mathcal{L}$-formulas. We denote:*

- $\mathcal{S}', \mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$ *iff* $\psi \in Cn_{\mathfrak{L}}(\mathcal{S}' \cup \bigcap \mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S}))$*;*
- $\mathcal{S}', \mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$ *iff* $\psi \in \bigcap_{\mathcal{T} \in \mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S})} Cn_{\mathfrak{L}}(\mathcal{S}' \cup \mathcal{T})$*;*
- $\mathcal{S}', \mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cup\mathsf{mcs}} \psi$ *iff* $\psi \in \bigcup_{\mathcal{T} \in \mathsf{MCS}_{\mathfrak{L}}^{\mathcal{S}'}(\mathcal{S})} Cn_{\mathfrak{L}}(\mathcal{S}' \cup \mathcal{T})$*.*

In the definition above, $\mathcal{S}'$ is the set of the strict assumptions, and $\mathcal{S}$ is the set of defeasible assumptions. When $\mathcal{S}' = \emptyset$ we shall just omit it. In this case we have that:

- $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$ iff $\psi \in \mathrm{Cn}_{\mathfrak{L}}(\bigcap \mathsf{MCS}_{\mathfrak{L}}(\mathcal{S}))$;
- $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$ iff $\psi \in \bigcap_{\mathcal{T} \in \mathsf{MCS}_{\mathfrak{L}}(\mathcal{S})} \mathrm{Cn}_{\mathfrak{L}}(\mathcal{T})$;
- $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cup\mathsf{mcs}} \psi$ iff $\psi \in \bigcup_{\mathcal{T} \in \mathsf{MCS}_{\mathfrak{L}}(\mathcal{S})} \mathrm{Cn}_{\mathfrak{L}}(\mathcal{T})$.

Example 45. *Suppose that the base logic is classical logic (i.e., $\mathfrak{L} = \mathsf{CL}$).*

- *Let $\mathcal{S} = \{p, \neg p, q\}$. Then $\bigcap \mathsf{MCS}_{\mathsf{CL}}(\mathcal{S}) = \{q\}$, thus $\mathcal{S} \mid\!\sim^{\mathsf{CL}}_{\cap\mathsf{mcs}} q$ but $\mathcal{S} \not\mid\!\sim^{\mathsf{CL}}_{\cap\mathsf{mcs}} p$ and $\mathcal{S} \not\mid\!\sim^{\mathsf{CL}}_{\cap\mathsf{mcs}} \neg p$.*
- *Let $\mathcal{S} = \{p \wedge q, \neg p \wedge q\}$. Then $\bigcap \mathsf{MCS}_{\mathsf{CL}}(\mathcal{S}) = \emptyset$, thus $\mathcal{S} \mid\!\sim^{\mathsf{CL}}_{\cap\mathsf{mcs}} \psi$ only if ψ is a classical theorem. On the other hand, $\mathcal{S} \mid\!\sim^{\mathsf{CL}}_{\Cap\mathsf{mcs}} q$ (and still $\mathcal{S} \not\mid\!\sim^{\mathsf{CL}}_{\Cap\mathsf{mcs}} p$ and $\mathcal{S} \not\mid\!\sim^{\mathsf{CL}}_{\Cap\mathsf{mcs}} \neg p$).*
- *It is easy to verify that for any $\mathcal{S}$, if $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$ then $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$. As the previous item shows, the converse does not hold.*

The next result relates MCS-based entailments and entailments that are induced by argumentation frameworks that are based on classical logic:

Proposition 46. ([11, Propositions 4.3]),[50, Theorem 5][22] *Let $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$ be a logic-based argumentation framework, where $\mathfrak{L}$ is classical logic and $\emptyset \subset \mathcal{A} \subseteq \{\mathsf{Ucut}, \mathsf{Def}\}$. Then:*

- $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\mathsf{Grd}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Stb}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$.
- $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Stb}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\cup\mathsf{mcs}} \psi$.

If $\mathcal{A} = \{\mathsf{DirUcut}\}$, we have that:

- $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Prf}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Stb}} \psi$ *iff* $\mathcal{S} \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$.

Example 47. *By the last proposition, the correspondence between the examples in Remark 15 and those of Example 45 is not coincidental.*

We refer to [11] for many other results concerning the relations between reasoning with maximal consistency and logic-based argumentation (or, more precisely, sequent-based argumentation, a specific form of logic-based argumentation – see Remark 6).

The relation between ABA and maximally consistent subsets has been studied, e.g., in [48; 117; 121; 126]. In particular, a similar result as the one above is shown for simple contrapositive assumption-based frameworks (recall Definition 35).

Proposition 48. ([117, Theorems 1 and 3] and [48, Theorem 3]) *Let $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, \sim \rangle$ be a simple contrapositive assumption-based framework. Then:*

- $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Prf}} \psi$ *iff* $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\Cap\mathsf{Stb}} \psi$ *iff* $\Gamma, \Delta \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$.

[22] The results in [50] are phrased in the more general context of hypersequent-based argumentation. Since standard sequent calculi are special instances of hypersequent calculi, the results are applicable also to sequent-based argumentation.

- $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Prf}} \psi$ *iff* $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cup\mathsf{Stb}} \psi$ *iff* $\Gamma, \Delta \mid\!\sim^{\mathfrak{L}}_{\cup\mathsf{mcs}} \psi$.
- *If* $\mathsf{F} \in \Delta$ *then* $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\mathsf{Grd}} \psi$ *iff* $\Gamma, \Delta \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$.

If $\mathfrak{L}$ *is contrapositive then:*

- $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Prf}} \psi$ *iff* $\mathcal{ABF} \mid\!\sim^{\mathfrak{L},\mathcal{A}}_{\cap\mathsf{Stb}} \psi$ *iff* $\Gamma, \Delta \mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$.

Remark 49. *A result similar to the one of Proposition 48 is obtained in [126] for what is called there* well-behaved *assumption-based frameworks, which among other things requires closure of the underlying inference rules under contraposition. It is shown that for well-behaved assumption-based frameworks, it holds* $\mathsf{MCS}_{\mathfrak{L}}(\mathcal{ABF}) = \mathsf{Prf}(\mathcal{ABF}) = \mathsf{Stb}(\mathcal{ABF})$. *By including priorities, the results are further generalized to cover* preferred subtheories *[52].*

Example 50. *Recall Example 37 with the assumption-based framework for* $\mathfrak{L} = \mathsf{CL}$, $\Gamma = \emptyset$, $\Delta = \{p, \neg p, q\}$ *and* $\sim\psi = \{\neg\psi\}$ *for every formula* ψ. *Since* $\mathsf{Naive}(\mathcal{ABF}) = \mathsf{Prf}(\mathcal{ABF}) = \mathsf{Stb}(\mathcal{ABF}) = \{\{p, q\}, \{\neg p, q\}\}$, *we have* $\mathcal{ABF} \mid\!\sim_{\circ\mathsf{sem}} q$ *for* $\circ \in \{\Cap, \cup, \cap\}$ *and* $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$. *In view of Prop. 48 and Remark 49 it is not surprising that* $\mathsf{MCS}_{\mathsf{CL}}(\mathcal{S}) = \{\{p, q\}, \{\neg p, q\}\}$.

We turn now to MCS-based reasoning and ASPIC systems. In [145, §5.3.2] it is shown that Brewka's *preferred subtheories* [52] are an instance of ASPIC$^+$. Since no preference ordering is considered in this chapter, preferred subtheories correspond to maximally consistent subsets. The following proposition states this result in terms of sets of formulas.

Proposition 51. ([145, Theorem 34]) *Let* $\mathcal{AF}(AT) = \langle \mathsf{Arg}(AT), Attack \rangle$ *be an ASPIC-argumentation framework for some ASPIC-argumentation theory AT, based on a propositional language* $\mathcal{L}$, *a set* $\mathcal{S}$ *of* $\mathcal{L}$*-formulas, and where the rules are all strict. Suppose further that* $\Gamma \to \gamma \in \mathcal{R}$ *iff* γ *follows according to classical logic from* Γ. *Let* $\mathsf{Arg}(\Delta) \subseteq \mathsf{Arg}(AT)$ *be the arguments constructed from premises in* Δ. *Then:*

- *If* Δ *is a maximally consistent subset of* $\mathcal{S}$, *then* $\mathsf{Arg}(\Delta)$ *is a stable extension of* $\mathcal{AF}(AT)$.
- *If* $\mathcal{E}$ *is a stable extension of* $\mathcal{AF}(AT)$, *then* $\bigcup_{A \in \mathcal{E}} \mathsf{Prem}(A)$ *is a maximally consistent subset of* $\mathcal{S}$.

Example 52. *To illustrate the last result consider the ASPIC argumentation system* $AS = \langle \mathcal{L}, \overline{}, \mathcal{R}, n \rangle$, *where* $\mathcal{L}$ *is a propositional language with* $\mathsf{Atoms}(\mathcal{L}) = \{p, q\}$, *the rules in* $\mathcal{R}_s$ *coincide with those of classical logic as in Example 21,* $\mathcal{K}_p = \{p, \neg p, q\}$, $\mathcal{K}_n = \emptyset$, *and* $\overline{\phi} = \{\neg\phi\}$ *for any* $\mathcal{L}$*-formula* ϕ. *Among others, the following ASPIC-arguments can be constructed:*

$$A_1 : p \qquad A_2 : \neg p \qquad A_3 : q \qquad A_4 : A_1, A_2 \to \neg q$$

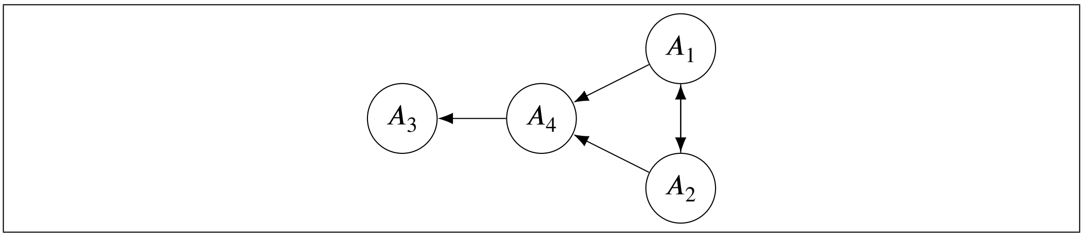


Figure 7: Part of the framework from Example 52.

The corresponding attack diagram is given in Figure 7.

$\mathcal{AF}(AT)$ has two stable extensions, one containing among others A_1 and A_3 and the second containing among others A_2 and A_3. As expected in view of Proposition 51, we see that these correspond to the two maximally consistent subsets of $\{p, \neg p, q\}$, namely: $\{p, q\}$ and $\{\neg p, q\}$.

Remark 53. *It is interesting to note that unlike some other frameworks (cf., e.g., Propositions 46 and 48), the grounded extension in the ASPIC framework of Example 52 does* not *contain the free formula q. This is since the inconsistent argument A_4 causes interferent behavior for the grounded semantics (see Section 2.3.2.B for more details).*

While the result in Proposition 51 above is about ASPIC-frameworks with only strict rules, one may also consider maximal consistent sets of formulas in the context of defeasible rules. In [127], maximal consistent sets of defeasible rules are defined as follows:

Definition 54 (MCS(AT)). *Let* AT $= \langle \mathrm{AS}, \mathcal{K} \rangle$ *be an ASPIC argumentation theory, where* $\mathcal{K} = \langle \mathcal{K}_n, \mathcal{K}_p \rangle$, AS $= \langle \mathcal{L}, \overline{}, \mathcal{R}, n \rangle$, *and* $\mathcal{R} = \mathcal{R}_d \cup \mathcal{R}_s$. *We define:*

- $\mathcal{R}_d^{\mathcal{K}} = \mathcal{R}_d \cup \{\Rightarrow \phi \mid \phi \in \mathcal{K}_p\}$.
- *A set of defeasible rules $\mathcal{D} \subseteq \mathcal{R}_d^{\mathcal{K}}$ is* AT-inconsistent *iff there are $\mathcal{L}$-formulas ϕ and $\psi \in \overline{\phi}$, for which $\mathcal{K}_n \vdash_{\mathcal{R}_s \cup \mathcal{D}} \psi$ and $\mathcal{K}_n \vdash_{\mathcal{R}_s \cup \mathcal{D}} \phi$. Otherwise, $\mathcal{D}$ is* AT-consistent.[23]
- *A rule $r = \psi_1, \ldots, \psi_n \Rightarrow \phi \in \mathcal{R}_d^{\mathcal{K}}$ is* triggered *by some $\mathcal{D} \subseteq \mathcal{R}_d^{\mathcal{K}}$ if $\mathcal{K}_n \vdash_{\mathcal{R}_s \cup \mathcal{D}} \psi_i$ for each $1 \leq i \leq n$.*
- *$\hat{\wp}(\mathcal{R}_d^{\mathcal{K}})$ is the set of all $\mathcal{D} \subseteq \mathcal{R}_d^{\mathcal{K}}$ such that every $r \in \mathcal{D}$ is triggered by $\mathcal{D}$.*
- MCS(AT) *is the set of all $\subseteq$-maximal consistent $\mathcal{D} \in \hat{\wp}(\mathcal{R}_d^{\mathcal{K}})$.*

[23] Maximally consistent sets of defeasible rules also play a role in constrained input/output logics, see [139]

Example 55. *Let* AT $= \langle \mathrm{AS}, \mathcal{K} \rangle$ *be an ASPIC argumentation theory, where* AS $= \langle \mathcal{L}, \overline{}, \mathcal{R}, n \rangle$*,* $\mathcal{R}_d = \{r_1 : \top \Rightarrow p,\ r_2 : p \Rightarrow q,\ r_3 : \top \Rightarrow \neg q\}$*,* $\mathcal{R}_s$ *is induced by classical logic, and* $\mathcal{K} = \emptyset$*. Then,*

- $\hat{\wp}(\mathcal{R}_d^{\mathcal{K}}) = \big\{\{r_1\}, \{r_1, r_2\}, \{r_1, r_2, r_3\}, \{r_1, r_3\}, \{r_3\}\big\}$*, and*
- $\mathsf{MCS}(\mathrm{AT}) = \big\{\{r_1, r_2\}, \{r_1, r_3\}\big\}$.

Note that $\{r_2, r_3\} \notin \hat{\wp}(\mathcal{R}_d^{\mathcal{K}})$ *since* r_2 *is not triggered by this set. Also,* $\{r_1, r_2, r_3\} \in \hat{\wp}(\mathcal{R}_d^{\mathcal{K}}) \setminus \mathsf{MCS}(\mathrm{AT})$ *since the set is inconsistent.*

For the next result we need also the following definition:

Definition 56 (contrapositive ASPIC theory, $\mathsf{Arg}(\mathcal{D})$)**.** *Let* AT $= \langle \mathrm{AS}, \mathcal{K} \rangle$ *be an ASPIC argumentation theory as in the previous definition. Then:*

- AT *is contrapositive if it satisfies*

  S1 *If* $\Delta, \psi \vdash_{\mathcal{R}_s} \phi'$ *for some* $\phi' \in \overline{\phi}$ *then* $\Delta, \phi \vdash_{\mathcal{R}_s} \psi'$ *for some* $\psi' \in \overline{\psi}$*; and*

  S2 *If* $\Delta \vdash_{\mathcal{R}_s} \phi'$ *for some* $\phi' \in \overline{\phi}$ *then* $\Delta \setminus \{\phi\} \vdash_{\mathcal{R}_s} \phi'$.

- *For* $\mathcal{D} \in \hat{\wp}(\mathcal{R}_d^{\mathcal{K}})$*, we define:* $\mathsf{Arg}(\mathcal{D}) = \{A \in \mathsf{Arg}(\mathrm{AT}) \mid \mathsf{DefRules}(A) \subseteq \mathcal{D} \cap \mathcal{R}_d\}$.

We get the following representation theorem for ASPIC$^+$ frameworks without undercut attacks:

Proposition 57. ([127, Theorem 6]) *For any contrapositive ASPIC argumentation theory* AT *without undercut attacks, it holds that:*

$$\mathsf{Prf}(\mathcal{AF}(\mathrm{AT})) = \mathsf{Stb}(\mathcal{AF}(\mathrm{AT})) = \{\mathsf{Arg}(\mathcal{D}) \mid \mathcal{D} \in \mathsf{MCS}(\mathrm{AT})\}.$$

Example 58 (Example 55 continued)**.** *In Example 55 we have the two stable resp. preferred extensions* $\mathsf{Arg}(\{r_1, r_2\})$ *and* $\mathsf{Arg}(\{r_1, r_3\})$.

Maximal consistency is also related to properties of extensions and of argumentation semantics, as will be shown in the next section. Here we only comment on one such property, which is directly related to the maximally consistent subsets of the premises.

Remark 59. *Consider the following property, investigated in [3; 177]:*

$$\mathsf{MCS}_{\mathsf{CL}}(\mathcal{S}) = \{\mathit{Sup}(\mathcal{E}) \mid \mathcal{E} \in \mathit{Sem}(\mathcal{AF}(\mathcal{S}))\}.$$

It is shown that in classical argumentation frameworks (i.e., those that consist of classical arguments in the sense of Definition 4), the equation above is met for both the stable (i.e, *when* $\mathsf{Sem} = \mathsf{Stb}$*) and preferred (*$\mathsf{Sem} = \mathsf{Prf}$*) semantics, and when the attack relation is either DirDef, DirUcut, or BigArgAt, while for the other attacks (Def, Ucut, Reb, DefReb) the above property ceases to hold.*

Other properties of the attack relations, as well as properties of the extensions and of the induced entailments will be considered in the next sections.

2.3.2 Rationality Postulates for Argumentative Reasoning

Since the introduction of the rationality postulates for ASPIC in [60], they have become a standard to assess approaches to structured argumentation. The postulates state that the conclusions of a framework should be closed under its strict rules (in approaches without a distinction between strict and defeasible rules, this simply means closure under the rules of the system), that the set of conclusions should be consistent, and that the set of formulas that is the result of the closure of the conclusions should be consistent as well. Another property states that an extension should also contain all the sub-arguments of its arguments. These postulates may formally be defined as follows:

Definition 60 (rationality postulates for extensions)**.** *Let* $\mathcal{AF} = \langle \mathsf{Arg}, \mathit{Attack} \rangle$ *be an argumentation framework,* $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ *a logic,* Sem *a semantics for it and* $\mathcal{E} \in \mathsf{Sem}(\mathcal{AF})$*. Then* $\mathcal{AF}$ *satisfies:*

- sub-argument closure*, iff for all* $A \in \mathcal{E}$*,* $\mathit{Sub}(A) \subseteq \mathcal{E}$*;*
- closure*, iff* $Cn_{\mathfrak{L}}(\mathsf{Conc}(\mathcal{E})) = Conc(\mathcal{E})$*;*
- direct consistency*, iff* $\mathsf{Conc}(\mathcal{E})$ *is* $\vdash$*-consistent; and*
- indirect consistency*, iff* $Cn_{\mathfrak{L}}(\mathsf{Conc}(\mathcal{E}))$ *is* $\vdash$*-consistent.*

In [60] it was shown that, if an argumentation framework $\mathcal{AF}$ satisfies indirect consistency, it satisfies direct consistency as well and if $\mathcal{AF}$ satisfies closure and direct consistency, it also satisfies indirect consistency.

Following [60], many related rationality postulates were introduced in the literature, some of them will be discussed in what follows. While the postulates in [60] are mainly concerned with the properties of the extensions of a framework (under certain semantics), there are other postulates that are related to the inferences relations induced by the frameworks. For instance, the *non-interference* and *crash-resistance* postulates, introduced in [61], guarantee that the entailment relation of argumentation frameworks do not collapse in view of inconsistent information. Next, we formalize these postulates.

For the next definitions, we say that two sets $\mathcal{S}_1, \mathcal{S}_2$ of $\mathcal{L}$-formulas are *syntactically disjoint* iff $\mathsf{Atoms}(\mathcal{S}_1) \cap \mathsf{Atoms}(\mathcal{S}_2) = \emptyset$.[24] This will be denoted by $\mathcal{S}_1 \mid \mathcal{S}_2$.

[24] Recall that $\mathsf{Atoms}(\mathcal{S})$ denotes the set of atoms occurring in the formulas of $\mathcal{S}$.

Definition 61 (rationality postulates for inferences)**.** *Let* $\mathrel{|\!\sim} \subseteq \wp(\mathcal{L}) \times \mathcal{L}$*.*

- *We say that* $\mathrel{|\!\sim}$ *satisfies* non-interference*, iff for every two sets* $\mathcal{S}_1, \mathcal{S}_2$ *of* $\mathcal{L}$*-formulas, and every* $\mathcal{L}$*-formula* ϕ *such that* $\mathcal{S}_1 \cup \{\phi\} \mid \mathcal{S}_2$*, it holds that* $\mathcal{S}_1 \mathrel{|\!\sim} \phi$ *iff* $\mathcal{S}_1, \mathcal{S}_2 \mathrel{|\!\sim} \phi$*.*

- *We say that* $\mathrel{|\!\sim}$ *satisfies* crash-resistance *iff there is no* $\mathrel{|\!\sim}$*-contaminating set* $\mathcal{S}$ *of* $\mathcal{L}$*-formulas, where a set* $\mathcal{S}$ *such that* $\mathsf{Atoms}(\mathcal{S}) \subsetneq \mathsf{Atoms}(\mathcal{L})$*, is called* contaminating *(w.r.t.* $\mathrel{|\!\sim}$*), if for every* $\mathcal{S}'$ *such that* $\mathcal{S} \mid \mathcal{S}'$ *and for every* $\mathcal{L}$*-formula* ϕ*, it holds that* $\mathcal{S} \mathrel{|\!\sim} \phi$ *iff* $\mathcal{S}, \mathcal{S}' \mathrel{|\!\sim} \phi$*.*

Remark 62. *In [61] it is shown that crash-resistance follows from non-Interference under some very weak criteria on the monotonic base logic.*

Note, for instance, that the consequence relation $\vdash_{\mathsf{CL}}$ of classical logic does not satisfy either of the properties of Definition 61. Indeed, where $\mathcal{S}_2$ is inconsistent, non-interference is violated, and any inconsistent set is $\vdash_{\mathsf{CL}}$-contaminating. We refer to [61] for more discussion on non-interference and crash-resistance.

Since rationality postulates are an important indicator of the usefulness of an argumentation system, extensive research has been conducted on the properties a system should satisfy in order for the rationality postulates to be satisfied. In the remainder of this section we will discuss the results of this research for the three approaches to logical argumentation frameworks discussed earlier.

A. Rationality postulates for logic-based methods

There are many studies on the properties of logic-based frameworks, including those in [111; 4; 2; 49; 12; 50]. Below, we survey the main results, starting with the postulates that are concerned with the properties of the attack rules and then those that are related to the properties of extensions and extension-based inferences.

Studies on requirements on the attack relation of a classical argumentation framework to fulfill rationality postulates are presented in [3; 177]. The conditions considered in those work are presented next.

Definition 63 (attack relation properties)**.** *Let* $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}(\mathcal{S}), Attack \rangle$ *be a classical argumentation framework. Then Attack is called:*

- conflict-dependent*, iff for each* $(A, B) \in Attack$*,* $\mathsf{Sup}(A) \cup \mathsf{Sup}(B) \vdash \mathsf{F}$*;*

- conflict-sensitive*, iff for each* $A, B \in \mathsf{Arg}(\mathcal{S})$*, if* $\mathsf{Sup}(A) \cup \mathsf{Sup}(B) \vdash \mathsf{F}$ *then* $(A, B) \in Attack$*;*

- valid, *iff for each* $\mathcal{E} \subseteq \mathsf{Arg}(\mathcal{S})$*, if* $\mathcal{E}$ *is conflict-free, then* $\mathsf{Sup}(\mathcal{E})$ *is consistent;*
- conflict-complete, *iff for every minimally inconsistent set* $\mathcal{T} \subseteq \mathcal{S}$*, for every* $\mathcal{T}_1, \mathcal{T}_2 \subseteq \mathcal{T}$ *such that* $\mathcal{T}_1 \neq \emptyset, \mathcal{T}_2 \neq \emptyset$ *and* $\mathcal{T}_1 \cup \mathcal{T}_2 = \mathcal{T}$ *and for every* $A \in \mathsf{Arg}(\mathcal{S})$ *with* $\mathsf{Sup}(A) = \mathcal{T}_1$ *there is an argument* $B \in \mathsf{Arg}(\mathcal{S})$ *with* $\mathsf{Sup}(B) = \mathcal{T}_2$ *such that* $(B, A) \in Attack$*;*
- symmetric, *iff when* $(A, B) \in Attack$ *also* $(B, A) \in Attack$*.*

We refer to [3; 177] for a discussion on these properties and the relations among them. Table 2 summarizes which of the properties above are satisfied by the attack rules from Table 1.[25]

Attack rule	**conflict-dependent**	**conflict-sensitive**	**valid**	**conflict-complete**	**svmmetric**
Def	✓	×	×	✓	×
DirDef	✓	×	×	×	×
Ucut	✓	×	×	✓	×
DirUcut	✓	×	×	×	×
ConUcut	✓	×	×	×	×
Reb	✓	×	×	×	✓
DefReb	✓	×	×	×	✓
Reb ∪ DirUcut	✓	×	×	×	×
BigArgAt	✓	×	×	×	×

Table 2: The satisfiability of the properties from Definition 63 for attack rules in Table 1.

Another study on the properties of attack relations in logic-based argumentation frameworks is given in [111]. Again, the study refers to classical argumentation framework, that is: the arguments meet the restrictions in Definition 4. An overview over various necessary and sufficient conditions on the attack relations considered in [111] is given in Table 3.

Proposition 64. ([111, Propositions 6 and 10]) *Where* $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}(\mathcal{S}), Attack \rangle$ *is a classical argumentation framework:*

[25] Note that, in this context, Reb ∪ DirUcut is the only union of attack rules considered in the literature.

Necessary conditions on attacks		
If $(A, B) \in$ Attack, then	$\{\mathsf{Conc}(A)\} \cup \mathsf{Sup}(B) \vdash \mathsf{F}$.	(D1)
	there is a $\phi \in \mathsf{Sup}(B)$ s.t. $\mathsf{Conc}(A) \vdash \neg\phi$.	(D1′)
	$\mathsf{Conc}(A) \vdash \neg\mathsf{Conc}(B)$.	(D1″)
	$\neg\mathsf{Conc}(A) \vdash \bigwedge \mathsf{Sup}(B)$,	(D5)
	there is a $\phi \in \mathsf{Sup}(B)$ s.t. $\neg\mathsf{Conc}(A) \vdash \phi$.	(D5′)
	$\neg\mathsf{Conc}(A) \vdash \mathsf{Conc}(B)$,	(D5″)
	there is a $\Gamma \subseteq \mathsf{Sup}(B)$ s.t. $\vdash \neg\mathsf{Conc}(A) \equiv \bigwedge \Gamma$.	(D5‴)
Sufficient conditions on attacks		
$(C, B) \in$ Attack if $(A, B) \in$ Attack and	$\vdash \mathsf{Conc}(A) \equiv \mathsf{Conc}(C)$	(D2)
	$\mathsf{Conc}(C) \vdash \mathsf{Conc}(A)$	(D2′)
$(A, C) \in$ Attack, if $(A, B) \in$ Attacks and	$\vdash \mathsf{Sup}(B) = \mathsf{Sup}(C)$	(D3)
	$\mathsf{Sup}(B) \subseteq \mathsf{Sup}(C)$	(D3′)
There is a C such that $\mathsf{Conc}(A) \vdash \mathsf{Conc}(C)$ and $(C, B) \in$ Attack, if	$\{\mathsf{Conc}(A)\} \cup \mathsf{Sup}(B) \vdash \mathsf{F}$	(D6)
	there is a $\phi \in \mathsf{Sup}(B)$ s.t. $\mathsf{Conc}(A) \vdash \neg\phi$	(D6′)
	$\mathsf{Conc}(A) \vdash \neg\mathsf{Conc}(B)$	(D6″)
$(A, B) \in$ Attack if	there is a $\Gamma \subseteq \mathsf{Sup}(B)$ s.t. $\vdash \mathsf{Conc}(A) \equiv \neg \bigwedge \Gamma$	(D6‴)
Sufficient and necessary conditions on attacks		
$(A, B) \in$ Attack iff $(A', B') \in$ Attack, if	$\vdash A \equiv A'$ and $\vdash B \equiv B'$	(D0)

Table 3: Conditions on the attack relations in [111].

- *Table 4, summarizes which of the postulates from Table 3 hold for the attack rules from Table 1.*

- *Table 5 summarizes by which of the postulates from Table 3 the different attack rela-*

tions are characterized.

	Def	**DirDef**	**Ucut**	**DirUcut**	**CanUcut**	**Reb**	**DefReb**
D0	✓	✓	✓	✓	✓	✓	✓
D1	✓	✓	✓	✓	✓	✓	✓
D2	✓	✓	✓	✓	✓	✓	✓
D2′	✓	✓	×	×	×	×	✓
D3	✓	✓	✓	✓	✓	×	×
D3′	✓	✓	✓	✓	×	×	×

Table 4: Overview of the constraints on the attack relation (Table 3) that are satisfied by the rules from Table 1 (Based on [111, Table 1 and Proposition 6])

	D1, D6	**D1′, D6′**	**D1″, D6″**	**D6‴**
D2′	Def	DirDef	DefReb	-
D2	CanUcut (D5)	DirUcut (D5′)	Reb (D5″)	-
-	-	-	-	Ucut (D5‴)

Table 5: Overview of the attack relation postulates from Table 3 that characterize the attack rules from Table 1. An attack rule is characterized by the conjunction of the attack relation postulates from the appropriate row, column and (where applicable) the cell. For example, the attack rule is direct undercut iff the attack relation postulates D1′, D2, D5′ and D6′ are satisfied (Based on [111, Table 2 and Proposition 10]).

Remark 65. *The interplay between logical principles about argumentation, on the one hand, and inference principles as studied in proof theory, on the other hand, is also studied in [70]. In that paper a series of logical principles of attack relations in argumentation frameworks is stated, and their collection leads to a characterization of classical logical consequence relations that only involves argumentation frameworks. We refer to [70] and [71] for further details.*

We turn now to postulates concerning the extensions of logic-based argumentation frameworks. Definition 66 lists rationality postulates studied in, e.g., [60; 111; 4; 2; 3; 12].[26]

Definition 66 (extension-based postulates)**.** *Let* $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}(\mathcal{S}), Attack\rangle$ *be an argumentation framework for* $\mathcal{S}$*, based on a logic* $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$*, and let* $\mathsf{Free}_{\mathfrak{L}}(\mathcal{S}) = \bigcap \mathsf{MCS}_{\mathfrak{L}}(\mathcal{S})$*.*[27] *The following postulates are defined with respect to the* Sem*-extensions of* $\mathcal{AF}(\mathcal{S})$*.*

Postulates on Individual Extensions, where $\mathcal{E} \in \mathsf{Sem}(\mathcal{AF}(\mathcal{S}))$*:*

- Support consistency: $\bigcup_{A\in\mathcal{E}} \mathsf{Sup}(A) \nvdash \mathsf{F}$*;*
- Consistency: $\bigcup_{A\in\mathcal{E}} \mathsf{Conc}(A) \nvdash \mathsf{F}$*;*
- Closure under support: *if* $\mathsf{Sup}(A) \subseteq \mathsf{Sup}(\mathcal{E})$ *then* $A \in \mathcal{E}$*;*
- Exhaustiveness: *if* $\mathsf{Sup}(A) \cup \{\mathit{Conc}(A)\} \subseteq \mathit{Conc}(\mathcal{E})$*, then* $A \in \mathcal{E}$*;*
- Strong exhaustiveness: *if* $\mathsf{Sup}(A) \subseteq \mathit{Conc}(\mathcal{E})$*, then* $A \in \mathcal{E}$*;*
- Support inclusion: $\mathsf{Sup}(\mathcal{E}) \subseteq \mathit{Conc}(\mathcal{E})$*;*
- Limited [strong] exhaustiveness: *[strong] exhaustiveness restricted to extensions* $\mathcal{E}$ *with* $\bigcup \mathsf{Sup}(\mathcal{E}) \neq \emptyset$*.*

Semantic-Wide Postulates:

- Core support consistency: $\bigcup_{A\in\bigcap \mathsf{Sem}(\mathcal{AF}(\mathcal{S}))} \mathsf{Sup}(A) \nvdash \mathsf{F}$*;*
- Core conclusion consistency: $\bigcap_{\mathcal{E}\in Sem(\mathcal{AF}(\mathcal{S}))} \mathit{Conc}(\mathcal{E}) \nvdash \mathsf{F}$*;*
- Core consistency: $\bigcup_{A\in\bigcap \mathsf{Sem}(\mathcal{AF}(\mathcal{S}))} \mathsf{Conc}(A) \nvdash \mathsf{F}$*;*
- Core closure: $\bigcap_{\mathcal{E}\in Sem(\mathcal{AF}(\mathcal{S}))} \mathit{Conc}(\mathcal{E}) = Cn_{\mathfrak{L}}(\bigcap_{\mathcal{E}\in Sem(\mathcal{AF}(\mathcal{S}))} \mathit{Conc}(\mathcal{E}))$*;*
- Non-triviality: *there is an* $\mathcal{S}$ *for which* $\mathsf{Arg}(\mathcal{S}) \setminus \mathsf{Arg}(\mathsf{Free}(\mathcal{S})) \neq \emptyset$ *and* $\mathsf{Arg}(\mathcal{S}) \neq \bigcup \mathit{Sem}(\mathcal{AF}(\mathcal{S}))$*;*
- Free precedence: $\mathsf{Arg}(\mathsf{Free}(\mathcal{S})) \subseteq \bigcap \mathit{Sem}(\mathcal{AF}(\mathcal{S}))$*;*
- Maximal consistency: $\mathit{Sem}(\mathcal{AF}(\mathcal{S})) = \{\mathsf{Arg}(\mathcal{T}) \mid \mathcal{T} \in \mathit{MCS}_{\mathfrak{L}}(\mathcal{S})\}$*;*
- Stability: $\mathit{Stb}(\mathcal{AF}(\mathcal{S})) \neq \emptyset$*;*
- Strong stability: $\mathit{Stb}(\mathcal{AF}(\mathcal{S})) = \mathit{Prf}(\mathcal{AF}(\mathcal{S}))$.

[26] We use naming conventions from [2; 12].

[27] When the underlying logic is clear from the context, we shall just write $\mathsf{Free}(\mathcal{S})$.

We start with the results in [111]:

Proposition 67. *Let $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}(\mathcal{S}), Attack\rangle$ be a classical argumentation framework. Table 6 summarizes which of the (semantic-wide) postulates from Definition 66 are satisfied in $\mathcal{AF}(\mathcal{S})$ with respect to a semantic* Sem *and the conditions in Table 3.*

Postulate	**Semantics**	**1,2,6**	**$1'$,2,$6'$**	**$1'$,2,$6''$**	**1,2,$6'''$**
Free precedence	Sem_1	✓	✓	✓	✓
Non-triviality	Sem_2	×	×	×	×
Non-triviality	Grd	✓	✓	✓	✓
Core support consistency	Sem_1	✓	✓	×	✓
$\mathsf{Grd}(\mathcal{AF}(\mathcal{S})) = \mathsf{Free}(\mathsf{Arg}(\mathcal{S}))$	Grd	✓	✓	×	✓
Consistency	Grd	✓	✓	×	✓
Consistency	Sem_1	×	+D3$'$ ✓	×	×

Table 6: Overview results of the (semantic-wide) postulates from Definition 66 that are satisfied by argumentation frameworks with semantics Sem (where $\mathsf{Sem}_1 \in \{\mathsf{Grd}, \mathsf{Cmp}, \mathsf{Prf}, \mathsf{Stb}\}$ and $\mathsf{Sem}_2 \in \{\mathsf{Cmp}, \mathsf{Prf}, \mathsf{Stb}\}$) and attacks satisfying the conditions in Table 3 (In the table, +D3$'$ denotes that the attack postulate D3$'$ is also required, in addition the postulates D1$'$, D2 and D6$'$).

Another investigation of the rationality postulates in Definition 66 for logic-based argumentation appears in [2] and [4]. Again, it is assumed that the supports of the arguments are consistent and minimal with respect to the subset relation. The core logic may be any explosive propositional logic, and the attack relations are divided according to the properties they have, which are specified in Definition 63 and in the following definition (see also [2, Definition 12]):

Definition 68 (postulates R_1 and R_2 for attack rules)**.** *Let $\mathcal{R}$ be an attack relation. The following conditions are verified with respect to every set $\mathcal{S}$ of $\mathcal{L}$-formulas:*[28]

R_1 *for every $A, B, C \in \mathsf{Arg}(\mathcal{S})$ such that $\mathsf{Sup}(A) \subseteq \mathsf{Sup}(B)$, it holds that if $(A, C) \in \mathcal{R}$ then $(B, C) \in \mathcal{R}$;*

[28] As usual, we freely exchange between the rule name and the corresponding relation.

R_2 *for every* $A, B, C \in \mathsf{Arg}(\mathcal{S})$ *such that* $\mathsf{Sup}(A) \subseteq \mathsf{Sup}(B)$*, it holds that if* $(C, A) \in \mathcal{R}$ *then* $(C, B) \in \mathcal{R}$*.*[29]

Proposition 69. *Let* $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}(\mathcal{S}), Attack\rangle$ *be an argumentation framework, for some explosive propositional logic* $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$ *and where the arguments are* $\vdash$*-consistent and* $\subseteq$*-minimal. Table 7 summarizes the results from [4; 2]. In particular, it shows which postulates are satisfied under the conditions in the left-most column.*[30]

In [12] and its extension in [47, Chapter 4] many of the postulates from Definitions 61 and 66 are investigated for sequent-based argumentation [14]. In particular, the arguments may be of the general form of Definition 5 (no constraints are posed on their supports). Also, the base logic is any logic satisfying the standard rules in Table 8. Therefore, the characterizations in [12] hold not only for classical logic, but also for many other logics, including intuitionistic logic and several modal logics.

Three classes of argumentation frameworks are studied:

- $\mathcal{AF}^{\mathsf{sub}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$: frameworks based on Defeat and/or Undercut, therefore it holds that $\mathcal{A} \cap \{\text{Def}, \text{Ucut}\} \neq \emptyset$;
- $\mathcal{AF}^{\mathsf{dir}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$: frameworks based on some and only direct attack rules, that is: $\emptyset \neq \mathcal{A} \subseteq \{\text{DirDef}, \text{DirUcut}\}$;
- $\mathcal{AF}^{\mathsf{con}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$: frameworks that, in addition to only direct attack rules, are based on Consistency Undercut, i.e., $\{\text{ConUcut}\} \subsetneq \mathcal{A} \subseteq \{\text{ConUcut}, \text{DirDef}, \text{DirUcut}\}$.

Proposition 70. ([12, Theorem 1]) *Let* $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$ *be a logic in which the rules of Table 8 are satisfied. Table 9 lists which rationality postulates are satisfied by the three classes of frameworks defined above, and with respect to which semantics* $\mathsf{Sem} \in \{\mathsf{Grd}, \mathsf{Cmp}, \mathsf{Prf}, \mathsf{Stb}\}$*.*

Remark 71. *The columns of* $\mathcal{AF}^{\mathsf{dir}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$ *and* $\mathcal{AF}^{\mathsf{con}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$ *in Table 9 show that all the postulates are compatible (that is, they can be satisfied together).*

In [49], relevance in structured argumentation is studied. In particular it is investigated, under which conditions the entailment relation induced by a framework of structured argumentation is robust under the addition of irrelevant information, i.e., information that can already be derived from it (semantic irrelevance) or information that is syntactically unrelated to the already available information (syntactic irrelevance). Rather than taking one of the

[29] Note that R_2 corresponds to $D3'$ in Table 3.

[30] Note that the results in Table 7 refer also to the ideal (Idl) and the semi-stable (SStb) semantics. We refer to [4; 2], as well as to [24; 22; 23] for their definitions.

[31] A logic $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$ is called *uniform* [136; 172], if for every two sets $\mathcal{S}_1, \mathcal{S}_2$ of $\mathcal{L}$-formulas and an $\mathcal{L}$-formula ψ it holds that $\mathcal{S}_1 \vdash \psi$ iff $\mathcal{S}_1, \mathcal{S}_2 \vdash \psi$ and $\mathcal{S}_2$ is a $\vdash$-consistent set such that $\mathsf{Atoms}(\mathcal{S}_2) \cap \mathsf{Atoms}(\mathcal{S}_1 \cup \{\psi\}) = \emptyset$.

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**
$\mathsf{Sem}(\mathcal{AF}(\mathcal{S})) = \emptyset$				✓		✓	
$\mathsf{Sem}(\mathcal{AF}(\mathcal{S})) = \emptyset + \mathrm{Cn}_{\mathfrak{L}}(\emptyset) \neq \emptyset$	×	×		✓		✓	
$\mathsf{Sem}(\mathcal{AF}(\mathcal{S})) = \emptyset + \mathsf{Free}(\mathcal{S}) \neq \emptyset$				✓		✓	×
$\mathsf{Sem}(\mathcal{AF}(\mathcal{S})) \neq \emptyset +$ $\mathcal{E} = \mathsf{Arg}(\mathsf{Supp}(\mathcal{E}))$			✓				
$\mathrm{Cn}_{\mathfrak{L}} \neq \emptyset + \mathsf{Sem} = \mathsf{Adm}$	×						
Closure	✓	✓					
Consistency				✓		✓	
Support consistency				✓	✓	✓	
Support consistency Conflict-dependent			Naive	✓	✓	✓	
Support cons. + Confl.-dep. + $\mathsf{Stb}(\mathcal{AF}(\mathcal{S})) \neq \emptyset$			Stb	✓	✓	✓	
Consistency + Sub arg. closure			✓	✓	✓	✓	
Consistency + $\mathcal{E} = \mathsf{Arg}(\mathsf{Supp}(\mathcal{E}))$				✓	✓	✓	
Conflict dependent							Sem_2
Conflict dependent + Sensitive			Sem_1				Sem_2
Conflict dependent + Symmetric + $\lvert C \rvert > 2$				×			Sem_2
Exhaustive + $\mathcal{E} = \mathsf{Arg}(\mathsf{Supp}(\mathcal{E}))$	✓	✓		✓	✓	✓	
$R_1 + R_2$			Sem_1				
R_2			Stb				

Table 7: Overview of the results from [4; 2], under the assumptions stated in Proposition 69. Legend of the postulates: P1 = closure, P2 = core closure, P3 = sub-argument closure, P4 = consistency, P5 = support consistency, P6 = core conclusion consistency, P7 = free precedence. Also, $\mathsf{Sem}_1 \in \{\mathsf{Grd}, \mathsf{Cmp}, \mathsf{Prf}, \mathsf{Idl}, \mathsf{Stb}, \mathsf{SStb}\}$ and $\mathsf{Sem}_2 \in \{\mathsf{Grd}, \mathsf{Prf}, \mathsf{Idl}, \mathsf{SStb}\}$. The condition $|C| > 2$ denotes that there is a minimal conflict of three or more formulas. Only the results from [4; 2] are shown: ✓ indicates that the postulate is satisfied for all considered semantics, Sem indicates that the postulate is satisfied for the particular semantics, × indicates that the postulate is not satisfied and an empty box indicates that the result is unknown, under the given conditions.

Rule Name	Rule's Conditions	Rule's Conclusion
Reflexivity		$\langle \psi, \psi \rangle$
Monotonicity	$\langle \Gamma, \Delta \rangle$	$\langle \Gamma \cup \Gamma', \Delta \cup \Delta' \rangle$
Transitivity	$\langle \Gamma_1, \Delta_1 \cup \{\psi\} \rangle$, $\langle \Gamma_2 \cup \{\psi\}, \Delta_2 \rangle$	$\langle \Gamma_1 \cup \Gamma_2, \Delta_1 \cup \Delta_2 \rangle$
Left-$\wedge$	$\langle \Gamma \cup \{\psi\} \cup \{\phi\}, \Delta \rangle$	$\langle \Gamma \cup \{\psi \wedge \phi\}, \Delta \rangle$
Right-$\wedge$	$\langle \Gamma, \Delta \cup \{\psi\} \rangle$, $\langle \Gamma, \Delta \cup \{\phi\} \rangle$	$\langle \Gamma, \Delta \cup \{\psi \wedge \phi\} \rangle$
Left-$\neg$	$\langle \Gamma, \Delta \cup \{\psi\} \rangle$	$\langle \Gamma \cup \{\neg\psi\}, \Delta \rangle$
Right-$\neg$	$\langle \Gamma \cup \{\psi\}, \Delta \rangle$	$\langle \Gamma, \Delta \cup \{\neg\psi\} \rangle$

Table 8: Rules for the base logics in Proposition 70.

main approaches to structured argumentation, a simple argumentation setting is introduced, into which the other approaches can be translated. The main results on syntactic relevance are based on the notion of *pre-relevance*, which is related to *basic relevance* known from relevance logics [18]. Intuitively, a consequence relation satisfies pre-relevance, if the derived conclusion can be derived from a relevant (w.r.t. shared atoms) subset of the antecedents.

Definition 72 (pre-relevance)**.** *A consequence relation* $\vdash \subseteq \wp(\mathcal{L}) \times \mathcal{L}$ *satisfies* pre-relevance*, if for each disjoint sets* $\mathcal{S}_1 \cup \{\phi\} \mid \mathcal{S}_2$*, if* $\mathcal{S}_1, \mathcal{S}_2 \vdash \phi$ *then there is some* $\mathcal{S}_1' \subseteq \mathcal{S}_1$ *such that* $\mathcal{S}_1' \vdash \phi$*.*

Example 73. *We list some entailment relations that satisfy pre-relevance:*

- *the consequence relation of the (semi-)relevance logic RM ([19, Proposition 6.5]),*
- *the entailment* $\vdash^{\mathsf{T}}_{\mathsf{CL}}$ *that is the restriction of* $\vdash_{\mathsf{CL}}$ *to pairs* (Γ, ϕ)*, for which it holds that* $\nvdash_{\mathsf{CL}} \neg \bigwedge \Gamma$*, and*
- *the entailment* $\mid\!\sim^{\mathsf{CL}}_{\cup\mathsf{mcs}}$ *(Definition 44).*[32]

The following proposition follows from [49, Theorem 1].

Proposition 74. *Let* $\vdash$ *be a pre-relevant consequence relation over the language* $\mathcal{L}$*,* $\mathcal{S}$ *be a set of* $\mathcal{L}$*-sentences,* $\mathsf{Arg}_{\vdash}(\mathcal{S}) = \{\langle \Gamma, \phi \rangle \mid \Gamma \vdash \phi\}$*, Attack is induced by direct attack rules*

[32] In [183] $\vdash^{\mathsf{T}}_{\mathsf{CL}}$ is used to obtain a crash-resistant version of ASPIC, and, similarly, in [112] the authors make use of $\vdash^{\mathsf{CL}}_{\cup\mathsf{mcs}}$ also for ASPIC.

Postulate	$\mathcal{AF}^{\mathsf{dir}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$	$\mathcal{AF}^{\mathsf{con}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$	$\mathcal{AF}^{\mathsf{sub}}_{\mathfrak{L},\mathcal{A}}(\mathcal{S})$
Closure	✓	✓	×
Closure under support	✓	✓	×
Sub-argument closure	✓	✓	✓
Support inclusion	✓	✓	✓
Consistency	✓	✓	×
Support consistency	✓	✓	×
Maximal consistency	Prf, Stb	Prf, Stb	×
Exhaustiveness	Prf, Stb	✓	×
Limited exhaustiveness	✓	✓	×
Strong exhaustiveness	Prf, Stb	✓	×
Limited strong exhaustiveness	✓	✓	×
Free precedence	Prf, Stb	✓	✓
Limited free precedence	✓	✓	✓
Stability	✓	✓	✓
Strong stability	✓	✓	✓
Non-interference	Prf, Stb	✓	✓
Crash-resistance	Prf, Stb	✓	✓

Table 9: Postulates satisfaction (Proposition 70, originally presented in [12]) for Sem $\in$ $\{$Grd, Cmp, Prf, Stb$\}$. Cells with ✓ indicate no conditions for the postulate, otherwise specific semantics with respect to which the postulate holds are indicated. Cells with × mean that the postulate does not hold. In case of non-interference and crash-resistance the base logic is assumed to be uniform.[31]

(such as DirDef and/or DirUcut) and let $\mathcal{AF}(\mathcal{S}) = \langle \mathsf{Arg}_{\vdash}(\mathcal{S}), Attack\rangle$ be the corresponding argumentation framework. Then $\mid\!\sim_{\star Sem}$ satisfies non-interference for $\star \in \{\cap, \Cap, \cup\}$ and $Sem \in \{Grd, Cmp, Prf\}$.

Remark 75. *Like the examples in items 2 and 3 of Example 73, consequence relations* $\vdash$

considered in Proposition 74 need not be induced by a logic in the technical sense of Definition 1. In fact, as is demonstrated in [49], structured argumentation frameworks such as ASPIC and ABA can be translated into the $\vdash$*-based argumentation frameworks of Proposition 74.*

B. Rationality postulates for ASPIC$^+$

Discussions on rationality postulates for ASPIC$^+$ can be found, among others, in [145; 147; 59]. For the completeness of the presentation we recall here some of the main results. For this, we need two notions, introduced in [147] and [90], respectively.

Definition 76 (well-formed argumentation framework). *An ASPIC argumentation framework defined by an ASPIC argumentation theory* $AT = \langle AS, \mathcal{K}\rangle$*, where* $AS = \langle \mathcal{L}, \bar{\ }, \mathcal{R}, n\rangle$ *and* $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_p$*, is called* well-formed*, if whenever* ϕ *is a contrary of* ψ *(i.e.,* $\phi \in \overline{\psi}$ *while* $\psi \notin \overline{\phi}$*), then* $\psi \notin \mathcal{K}_n$ *and* ψ *is not the consequent of a strict rule.*

Definition 77 (self-contradiction axiom; closure under transposition). *An ASPIC argumentation framework* $\mathcal{AF}(AT) = \langle \mathsf{Arg}(AT), Attack\rangle$*, defined by an ASPIC argumentation theory* $AT = \langle AS, \mathcal{K}\rangle$*, where* $AS = \langle \mathcal{L}, \bar{\ }, \mathcal{R}, n\rangle$ *and* $\mathcal{K} = \mathcal{K}_n \cup \mathcal{K}_p$ *satisfies:*

- the self-contradiction axiom*, if for each minimally inconsistent set* $\mathcal{S}$ *of* $\mathcal{L}$*-formulas it holds that* $\{\neg\phi \mid \phi \in \mathcal{S}\} \subseteq Cn_{\mathcal{R}_s}(\mathcal{S})$*;*[33]
- closure under transposition*, if for each* $\phi_1, \ldots, \phi_n \to \phi \in \mathcal{R}_s$*, for each* $i \in \{1, \ldots, n\}$*,* $\phi_1, \ldots, \phi_{i-1}, \neg\phi, \phi_{i+1}, \ldots, \phi_n \to \neg\phi_i \in \mathcal{R}_s$ *as well.*

Proposition 78. ([90],[147]) *Let* $\mathcal{AF}(AT) = \langle \mathsf{Arg}(AT), Attack\rangle$ *be an argumentation framework and let* $\mathcal{E} \in \mathsf{Cmp}(\mathcal{AF}(AT))$*. Table 10 lists the rationality postulates that are satisfied under the different conditions of Definitions 76 and 77.*

Remark 79. *The results in [147] are given for prioritized frameworks (i.e., with a preference relation defined on the arguments of* $\mathcal{AF}(AT)$*). However, since the non-prioritized setting is a special case of the prioritized setting, the results still apply here.*

The satisfaction of the non-interference and crash-resistance postulates for ASPIC$^+$ are not so straightforward. For example, when the strict rules are based on classical logic, explosion might still occur. See [59] for an extensive discussion on non-interference and crash-resistance for ASPIC$^+$. One of the challenges when trying to resolve these issues is that the postulates from [60] should still be satisfied by the resulting framework.

[33] A set $\mathcal{S}$ of $\mathcal{L}$-formulas is *minimally inconsistent* if there is some formula ϕ such that $\phi \in \mathrm{Cn}_{\mathcal{R}_S}(\mathcal{S})$ and $\overline{\phi} \in \mathrm{Cn}_{\mathcal{R}_S}(\mathcal{S})$, and for each $\mathcal{S}' \subsetneq \mathcal{S}$ no such ϕ exists.

Postulate	–	Well-formed framework	Self-contradiction axiom	Closure under transposition
Sub-argument closure	✓	✓	✓	✓
Closure	×	✓	✓	✓
Direct consistency	×	✓	✓	✓
Indirect consistency	×	✓	✓	✓

Table 10: Overview of the postulates that are satisfied by ASPIC$^+$ argumentation frameworks given some condition on the set of strict rules and a contrary relation. The column titled – denotes that there are no requirements placed on the framework.

Several variants of ASPIC$^+$ have been proposed in the literature, some of them satisfy non-interference and crash-resistance. An overview of some of these systems, the settings in which they have been studied and the postulates that they satisfy, can be found in Table 11.[34]

Remark 80. *Below are some further explanations and notes that are related to the results in Table 11.*

- *The variant* ASPIC Lite*, introduced in [183], is obtained by filtering all inconsistent arguments out of the argumentation framework. An argument A is inconsistent if* $\{\mathsf{Conc}(B) \mid B \in \mathsf{Sub}(A)\}$ *is inconsistent. It is then shown that non-interference and crash-resistance are satisfied for complete semantics, while the postulates from [60] are still satisfied as well. For the proof it is necessary that at least one extension exists. Among others, that is why other semantics are not discussed in that particular paper. Moreover, it is shown that the results do not hold when preferences are introduced.*
- *A weaker version of crash-resistance, called* non-triviality *is discussed in [112]. This variant, called* ASPIC**, restricts the application of strict rules. In particular, chaining of strict rules and applying strict rules to inconsistent sets of antecedents is not allowed.*
- *ASPIC$^-$ [63] is a variant of ASPIC$^+$ that uses the attack form of* unrestricted rebut*. Its violation of non-interference is shown in [124]. Closure is also violated if inconsistent arguments are filtered out, in the presence of priorities.*

[34] As for ASPIC$^+$ with filtering out inconsistent arguments: no results are known, even though ASPIC Lite is its subsystem.

System	Priorities	Incons. arg. filtered	Direct consistency	Closure	Crash resistance
ASPIC$^+$	Yes	No	✓	✓	×
ASPIC Lite	No	Yes	Cmp	Cmp	Cmp
ASPIC Lite	Yes	Yes	Cmp	×	Cmp
ASPIC*	Yes	No	✓	✓†	✓†
ASPIC$^-$	Yes	No	✓	✓	×
ASPIC$^-$	No	Yes	✓	✓	✓
ASPIC$^-$	Yes	Yes	✓	×	✓
ASPIC$^\ominus$	Yes	No	Grd	Grd	Grd

Table 11: Overview of the different variants to ASPIC$^+$ and the conditions under which some of the postulates are satisfied. "Yes" means that the results also hold when taking into account priorities over the defeasible rules, whereas "no" means that when priorities are taken into account, counter-examples to the results exist. In columns 4–6, ✓ denotes that the postulate is satisfied, × denotes that the postulate is not satisfied, and Cmp [resp. Grd] denotes that the postulate is studied and satisfied for complete [resp. grounded] semantics Finally, ✓† denotes that a weaker variant of the postulate is satisfied.

- *Another variant of ASPIC$^+$ with unrestricted rebut, called ASPIC$^\ominus$, is studied in [124] and [125]. In ASPIC$^\ominus$, the notion of unrestricted rebut is generalized such that an argument can attack another argument if its conclusion claims that a subset of the commitments of the attacked argument are not tenable together. It is shown that the resulting framework ASPIC$^\ominus$, where the priority relation is a preorder using the so-called* weakest link principle, *satisfies the rationality postulates from both [60] and [61] under grounded semantics.*

C. Rationality postulates for ABA

Recall from Section 2.2.3 that an extension is a set of assumptions (i.e., $\mathcal{E} \subseteq \mathcal{A}$ for every extension $\mathcal{E}$ of an assumption-based framework $\mathcal{ABF} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \sim \rangle$) that is also closed with respect to the rules in $\mathcal{R}$ (i.e., $\mathcal{E} = Cn_{\vdash_{\mathcal{R}}}(\mathcal{E})$). From this it follows immediately that the closure postulate from Definition 60 is satisfied. Thus, from the rationality postulates in

[60], it remains to show consistency. In the context of flat assumption-based argumentation frameworks, this postulate can be defined as follows [75; 126]:

> **Consistency:** for all extensions $\mathcal{E}$, it holds that there are no $\phi, \psi \in \mathcal{E}$ such that $\phi \in \sim\psi$.[35]

In the non-prioritized setting, as discussed in this chapter, it follows immediately that extensions for any of the considered semantics are consistent, since otherwise these would not be conflict-free (recall the definition of attack in assumption-based frameworks, Definition 33). However, as shown in e.g., [75; 126], whether an assumption-based framework satisfies consistency in a prioritized setting depends on the definition of the preference ordering and the notion of conflict-freeness. A discussion of this is beyond the scope of this chapter.[36]

The rationality postulates for inferences (recall Definition 61) have been studied for simple contrapositive assumption-based frameworks (recall Definition 35) in [121]. Note that, since the entailment relation for assumption-based frameworks is defined for frameworks and not for sets of formulas (as in the case of the discussed logic-based approaches), the notion of syntactically disjoint sets of formulas has to be lifted to assumption-based frameworks. Two assumption-based frameworks $\mathcal{ABF}_1 = \langle \mathfrak{L}, \Gamma_1, \Delta_1, \sim_1 \rangle$ and $\mathcal{ABF}_2 = \langle \mathfrak{L}, \Gamma_2, \Delta_2, \sim_2 \rangle$ are *syntactically disjoint* if $(\Gamma_1 \cup \Delta_1) \mid (\Gamma_2 \cup \Delta_2)$. Besides this new notion of syntactically disjointness, the definitions of non-interference and crash-resistance remain the same as for logic-based argumentation and the ASPIC-family.

Proposition 81. *([121, Theorems 7 and 8]) Let $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, \sim \rangle$ be a simple contrapositive assumption-based framework. Table 12 lists under what conditions the corresponding entailment relations satisfy non-interference for $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}, \mathsf{Grd}, \mathsf{WF}\}$.*

In [49] it is shown that

- ABA frameworks with domain-specific rules and whose contrariness relation do not introduce syntactic discontinuities, i.e., for all formulas ϕ we have that $\mathsf{Atoms}(\sim\phi) \subseteq \mathsf{Atoms}(\phi)$, satisfy non-interference, and
- ABA frameworks whose inference rules $\mathcal{R}$ are induced by logics $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ for which $\vdash$ is pre-relevant (see Definition 72), i.e., $\phi_1, \ldots, \phi_n \to \psi \in \mathcal{R}$ iff $\phi_1, \ldots, \phi_n \vdash \psi$, satisfy non-interference.

[35] Since [75; 126] restrict their attention to flat assumption-based argumentation frameworks, this notion of consistency is equivalent to the following formulation, which bears closer similarities to *indirect consistency*: for all extensions $\mathcal{E}$, it holds that there are no $\phi, \psi \in \mathcal{L}$ s.t. $\mathcal{E} \vdash_S \phi$ and $\mathcal{E} \vdash_S \psi$ and $\phi \in \sim\psi$.

[36] In contexts where besides the contrariness relation there are other negations (e.g., when translating extended logic programs into ABA), various notions of consistency may have to be considered (see e.g., [180]).

Entailment	–	$\mathsf{F} \in Ab$
$\mid\!\sim_{\Cap\mathsf{Sem}}$	$\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$	$\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$
$\mid\!\sim_{\cup\mathsf{Sem}}$	$\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$	$\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}\}$
$\mid\!\sim_{\cap\mathsf{Sem}}$	×	$\mathsf{Sem} \in \{\mathsf{Grd}, \mathsf{WF}\}$

Table 12: Results from [121] concerning the conditions and semantics under which simple-contrapositive assumption-based frameworks satisfy non-interference. × denotes that non-interference is not satisfied for any $\mathsf{Sem} \in \{\mathsf{Naive}, \mathsf{Prf}, \mathsf{Stb}, \mathsf{Grd}, \mathsf{WF}\}$.

2.3.3 Inference Principles for Non-Monotonic Reasoning

Next, we examine the argumentation-based entailment relations in Definitions 12, 26 and 36, relative to general patterns for non-monotonic reasoning, originally studied in [162], [98], [133; 134], and [137]. These works study how to adjust the set of conclusions (which may be reduced, not necessarily increased) upon a growth in the set of assumptions. In our case, since the assumptions are divided to strict premises and defeasible premises, it will be useful to distinguish between the two ways of increasing the set of premises: we shall use the operator $\uplus$ for the addition of strict premises and $\Cup$ for the addition of defeasible premises. Accordingly, we define:

Definition 82 (premise addition). *Let $\mathcal{S} = \langle \mathcal{S}_s, \mathcal{S}_d \rangle$ be a pair of sets of formulas in a language $\mathcal{L}$.*[37] *We denote:*

- $\mathcal{S} \Cup \phi \; = \; \langle \mathcal{S}_s, \mathcal{S}_d \rangle \Cup \phi \; = \; \langle \mathcal{S}_s, \; \mathcal{S}_d \cup \{\phi\} \rangle$,
- $\mathcal{S} \uplus \phi \; = \; \langle \mathcal{S}_s, \mathcal{S}_d \rangle \uplus \phi \; = \; \langle \mathcal{S}_s \cup \{\phi\}, \; \mathcal{S}_d \rangle$.

Note that logic-based argumentation is considered here only with respect to defeasible assumptions, therefore $\uplus$ will not be used in that context, and the meaning of $\Cup$ in case the logic-based argumentation is simply the union, $\cup$. For the other formalisms, addition of premises is defined as follows:

Definition 83 (premise addition in ASPIC). *Let $AT = \langle\langle \mathcal{L}, ^{-}, \mathcal{R}, n \rangle, \langle \mathcal{K}_n, \mathcal{K}_p \rangle\rangle$ be an ASPIC argumentation theory, and let ϕ be an $\mathcal{L}$-formula. We define:*

- $AT \Cup \phi = \langle\langle \mathcal{L}, ^{-}, \mathcal{R}, n \rangle, \mathcal{K} \Cup \phi \rangle$ *(where $\phi \notin \mathcal{K}_n$),*
- $AT \uplus \phi = \langle\langle \mathcal{L}, ^{-}, \mathcal{R}, n \rangle, \mathcal{K} \uplus \phi \rangle$ *(where $\phi \notin \mathcal{K}_p$).*

[37] The subscripts 's' and 'd' indicate that, intuitively, the first component consists of the strict premises and the second component is the set of defeasible premises.

Definition 84 (premise addition in ABA)**.** *Let $\mathcal{ABF} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, {\sim} \rangle$ be an assumption-based argumentation framework, and let ϕ be an $\mathcal{L}$-formula. We define:*

- $\mathcal{ABF} \Cup \phi = \langle \mathcal{L}, \mathcal{R} \setminus \{\Theta \to \phi \mid \Theta \subset \mathsf{WFF}(\mathfrak{L})\}, \mathcal{A} \cup \{\phi\}, {\sim}\rangle$,[38]
- $\mathcal{ABF} \uplus \phi = \langle \mathcal{L}, \mathcal{R} \cup \{\to \phi\}, \mathcal{A} \setminus \{\phi\}, {\sim}\rangle$.[39]

Let $\mathcal{ABF} = \langle \mathfrak{L}, \Gamma, \Delta, {\sim} \rangle$ be a (simple) contrapositive assumption-based argumentation framework, and let ϕ be an $\mathcal{L}$-formula. We define:

- $\mathcal{ABF} \Cup \phi = \langle \mathfrak{L}, \Gamma, \Delta \cup \{\phi\}, {\sim}\rangle$,
- $\mathcal{ABF} \uplus \phi = \langle \mathfrak{L}, \Gamma \cup \{\phi\}, \Delta, {\sim}\rangle$.[40]

Using the operators $\uplus$ and $\Cup$ we can now consider known postulates for non-monotonic reasoning, adjusted to the two types of information updates. To make the presentation more compact we will define the properties for ASPIC, ABA, MCS-based reasoning and logic-based argumentation in one definition. For this we call a *knowledge base* one of the following:

◇ an ASPIC argumentation theory AT $= \langle\langle \mathcal{L}, \overline{}, \mathcal{R}, n\rangle, \langle \mathcal{K}_n, \mathcal{K}_p\rangle\rangle$,
◇ an assumption-based framework $\mathcal{ABF}$,
◇ a set of $\mathcal{L}$-formulas for logic-based argumentation with a logic $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$, or
◇ a pair of $\mathcal{L}$-formulas $\langle \mathcal{S}', \mathcal{S}\rangle$ in MCS-based reasoning and a logic $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$.

In the context of a fixed language $\mathcal{L}$ resp. a fixed logic $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$ resp. a fixed set of strict rules $\mathcal{R}_s$, it will also be useful to consider *empty knowledge bases*, written $\mathsf{KB}_\emptyset$ and denoting, the argumentation theory AT $= \langle\langle \mathcal{L}, \overline{}, \emptyset, n\rangle, \langle \emptyset, \emptyset\rangle\rangle$ in the context of ASPIC, resp. the assumption-based framework $\langle \mathcal{L}, \mathcal{R}_s, \emptyset, \emptyset\rangle$ in the context of assumption-based argumentation, resp. the pair of empty sets of $\mathcal{L}$-formulas $\langle \emptyset, \emptyset\rangle$ in the context of MCS-based reasoning, resp. the empty set of $\mathcal{L}$-formulas in the context of logic-based argumentation.

Definition 85 (properties for non-monotonic reasoning)**.** *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash\rangle$ be a propositional logic,* KB *a knowledge base, ϕ, ψ, σ $\mathcal{L}$-formulas, and $\sqcup \in \{\Cup, \uplus\}$. For an entailment relation $\mid\sim\ \subseteq \wp(\mathcal{L}) \times \mathcal{L}$ we define:*

$\sqcup$-Cautious Reflexivity *($\sqcup$-**CREF**):* $\mathsf{KB}_\emptyset \sqcup \phi \mid\sim \phi$ *where ϕ is $\vdash$-consistent.*

$\sqcup$-Reflexivity *($\sqcup$-**REF**):* $\mathsf{KB}_\emptyset \sqcup \phi \mid\sim \phi$.

[38] Removing $\Theta \to \phi$ from Γ ensures that $\mathcal{ABF} \Cup \phi$ is flat if so is $\mathcal{ABF}$, and is proposed in [76]. Furthermore, we let ${\sim}\phi = \emptyset$ and ${\sim}\psi$ is defined as in the original $\mathcal{ABF}$ for any $\psi \in \mathcal{A}$.

[39] ${\sim}\psi$ is defined as in the original $\mathcal{ABF}$ for any $\psi \in \mathcal{A} \setminus \{\phi\}$.

[40] Since in the context of simple contrapositive assumption-based frameworks is is not necessary to restrict attention to flat assumption-based frameworks, ϕ is not removed from Δ.

Right Weakening *(***RW***): If* $\mathsf{KB} \mathrel{|\!\sim} \phi$ *and* $\phi \vdash \psi$ *then* $\mathsf{KB} \mathrel{|\!\sim} \psi$*.*

$\sqcup$-Cautious Monotonicity *(*$\sqcup$-**CM***): If* $\mathsf{KB} \mathrel{|\!\sim} \phi$ *and* $\mathsf{KB} \mathrel{|\!\sim} \psi$ *then* $\mathsf{KB} \sqcup \{\phi\} \mathrel{|\!\sim} \psi$*.*

$\sqcup$-Cautious Cut *(*$\sqcup$-**CC***): If* $\mathsf{KB} \mathrel{|\!\sim} \psi$ *and* $\mathsf{KB} \sqcup \{\psi\} \mathrel{|\!\sim} \phi$ *then* $\mathsf{KB} \mathrel{|\!\sim} \phi$*.*

$\sqcup$-Left Logical Equivalence *(*$\sqcup$-**LLE***): If* $\vdash \phi \equiv \psi$ *and* $\mathsf{KB} \sqcup \phi \mathrel{|\!\sim} \sigma$ *then* $\mathsf{KB} \sqcup \psi \mathrel{|\!\sim} \sigma$*.*

$\sqcup$-OR *(*$\sqcup$-**OR***): If* $\mathsf{KB} \sqcup \phi \mathrel{|\!\sim} \delta$ *and* $\mathsf{KB} \sqcup \psi \mathrel{|\!\sim} \delta$ *then* $\mathsf{KB} \sqcup \{\phi \vee \psi\} \mathrel{|\!\sim} \delta$*.*

$\sqcup$-Rational Monotonicity *(*$\sqcup$-**RM***): If* $\mathsf{KB} \mathrel{|\!\sim} \psi$ *and* $\mathsf{KB} \not\mathrel{|\!\sim} \neg\phi$ *then* $\mathsf{KB} \sqcup \phi \mathrel{|\!\sim} \psi$*.*[41]

Remark 86. *We refer to [133; 134] for a detailed discussion on* **CM**, **RW**, **LLE**, **OR**, *and* **RM** *and to [98] for a discussion on* **CC**. *All of these properties are well-known and have been extensively examined in different contexts and for different purposes involving inference in a non-monotonic way.*

Some interesting variations of these properties have been considered in the literature but have, to the best ouf our knowledge, not been studied for argumentative consequence relations. For example, an interesting weaker variant of cautious monotony is known as very cautious monotony *(***VCM***) [116] or* conjunctive cautious monotony *[43] and is defined as follows: if* $\Gamma \mathrel{|\!\sim} \phi \wedge \psi$ *then* $\Gamma \sqcup \phi \mathrel{|\!\sim} \psi$*. This variant has not been studied yet in structured argumentation.*

Another variation is semi-monotonicity *(***SM***) [7], stating that when adding defeasible information, every extension (according to a given semantics) of the original framework is a subset of some extension of the supplemented framework. For more variants of the properties discussed here, we refer the reader to [43; 95] in which many more variants are discussed and studied.*

The properties in Definition 85 are often gathered for defining systems for non-monotonic inference.

Definition 87 (systems for non-monotonic inference)**.** *Let* $\sqcup \in \{\Cup, \uplus\}$*. We say that an entailment* $\mathrel{|\!\sim}$ *is:*

- $\sqcup$-cumulative*, if it satisfies* $\sqcup$-**REF**, **RW**, $\sqcup$-**LLE**, $\sqcup$-**CM** *and* $\sqcup$-**CC**.
- $\sqcup$-cautiously cumulative*, if it satisfies* $\sqcup$-**CREF**, **RW**, $\sqcup$-**LLE**, $\sqcup$-**CM** *and* $\sqcup$-**CC**.
- $\sqcup$-(cautiously) preferential*, if it is* $\sqcup$*-(cautiously) cumulative and satisfies* $\sqcup$-**OR**.
- $\sqcup$-(cautiously) rational*, if it is* $\sqcup$*-(cautiously) preferential and satisfies* $\sqcup$-**RM**.

[41] In ASPIC this has to be rephrased in terms of the contrariness relation instead of negation: If $\mathsf{KB} \mathrel{|\!\sim} \psi$ and $\mathsf{KB} \not\mathrel{|\!\sim} \phi'$ for all $\phi' \in \overline{\phi}$, then $\mathsf{KB} \sqcup \phi \mathrel{|\!\sim} \psi$.

Table 13 classifies the argumentation-based entailment relations according to Definition 87.[42]

	MCS reasoning		logic-based arg.		simple contrap. ABA			ASPIC
System.	$\mathrel{\mid\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}}$	$\mathrel{\mid\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$	$\mathrel{\mid\!\sim}^{\mathfrak{L},\mathcal{A}_{\mathsf{UD}}}_{\cap\mathsf{gps}}$	$\mathrel{\mid\!\sim}^{\mathsf{CL},\mathsf{DirUcut}}_{\Cap\mathsf{ps}}$	$\mathrel{\mid\!\sim}_{\Cap\mathsf{ps}}$	$\mathrel{\mid\!\sim}_{\cap\mathsf{gps}}$	$\mathrel{\mid\!\sim}_{\mathsf{Grd}}$ (†)	$\mathrel{\mid\!\sim}_{\Cap\mathsf{Stb}}$ (‡)
$\Cup$-ccum.	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
$\uplus$-cum.	Yes	Yes	–	–	Yes	Yes	Yes	Yes
$\Cup$-cpref.	No	Yes	No	Yes	Yes	No	No	Yes
$\uplus$-pref.	No	Yes	–	–	Yes	No	No	Yes
$\Cup$-crat.	No	No	No	No	No	No	No	No
$\uplus$-rat.	No	No	–	–	No	No	No	No

Table 13: Overview over the properties of non-monotonic inference. In the table, "(c)cum." means "(cautiously) cumulative", "(c)pref." means "(cautiously) preferential", and "(c)rat." means "(cautiously) rational". We let: $\emptyset \subset \mathcal{A}_{\mathsf{UD}} \subseteq \{\mathsf{Ucut}, \mathsf{Def}\}$, $\mathsf{gps} \in \{\mathsf{Grd}, \mathsf{Prf}, \mathsf{Stb}\}$, and $\mathsf{ps} \in \{\mathsf{Prf}, \mathsf{Stb}\}$, Also, (†) means that $\mathsf{F} \in \Delta$, (‡) means "without defeasible rules", and "–" means that the property is not applicable in the context of the given entailment.

The positive results presented in Table 13 follow from the representational results in Propositions 46, 48 and 51, using the next two propositions:

Proposition 88. *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a propositional logic. The entailments $\mathrel{\mid\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}}$ and $\mathrel{\mid\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ are $\Cup$-cautiously cumulative and $\uplus$-cumulative.*

Proposition 89. *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a propositional logic and let $\sqcup \in \{\Cup, \uplus\}$. The entailment $\mathrel{\mid\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ is $\sqcup$-preferential.*

Proofs of the last two propositions are given in Appendix A.

Remark 90. *Some of the results in Table 13 have been shown before. For instance, in [30] it is shown that $\mathrel{\mid\!\sim}^{\mathsf{CL}}_{\Cap\mathsf{mcs}^{\emptyset}_{\mathfrak{L}}}$ is $\Cup$-preferential, the results for simple contrapositive ABFs are shown in [117], and the results concerning the $\Cup$-cautious cumulativity and the non $\Cup$-cautious preferentiality of $\mathrel{\mid\!\sim}^{\mathfrak{L},\mathcal{A}_{\mathsf{UD}}}_{\cap\mathsf{gps}}$ follow from [16, Proposition 16 and Note 10].*

[42] Since the credulous entailment is often monotonic (see [31] for MCS-based reasoning and [50, Proposition 8] for argumentation-based reasoning), the results in Table 13 refer to skeptical entailments.

Counter-examples for $\sqcup$-**OR** which justify the negative results in Table 13 are easy to find. We give some examples for MCS-based reasoning, which in view of the cited representational results immediately generalize for the listed argumentation systems in Table 13.

Example 91 (Counter-Example, $\uplus$-OR, $\mathrel{|\!\sim}_{\cap\mathsf{mcs}}$)**.** *Suppose that the underlying logic $\mathfrak{L}$ is classical logic, and let $\mathcal{S} = \{\neg p \wedge r, \neg q \wedge r\}$. In this case we have:*

- $\langle\{p\}, \mathcal{S}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*, since* $\mathsf{MCS}^{\{p\}}(\mathcal{S}) = \{\{\neg q \wedge r\}\}$*,*
- $\langle\{q\}, \mathcal{S}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*, since* $\mathsf{MCS}^{\{q\}}(\mathcal{S}) = \{\{\neg p \wedge r\}\}$*, while*
- $\langle\{p \vee q\}, \mathcal{S}\rangle \not\mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*, since* $\mathsf{MCS}^{\{p\vee q\}}(\mathcal{S}) = \{\{\neg p \wedge r\}, \{\neg q \wedge r\}\}$.

Example 92 (Counter-example, $\Cup$-OR, $\mathrel{|\!\sim}_{\cap\mathsf{mcs}}$)**.** *Suppose again that the underlying logic $\mathfrak{L}$ is classical logic, and let $\mathcal{S} = \{\neg p, \neg q, \neg p \supset r, \neg q \supset r\}$. Then we have:*

- $\langle\emptyset, \mathcal{S} \cup \{p\}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*,*
  since $\mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{p\}) = \{\{p, \neg q, \neg p \supset r, \neg q \supset r\}, \{\neg p, \neg q, \neg p \supset r, \neg q \supset r\}\}$ *and thus* $\bigcap \mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{p\}) = \{\neg q, \neg p \supset r, \neg q \supset r\}$*,*
- $\langle\emptyset, \mathcal{S} \cup \{q\}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*,*
  since $\mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{q\}) = \{\{\neg p, q, \neg p \supset r, \neg q \supset r\}, \{\neg p, \neg q, \neg p \supset r, \neg q \supset r\}\}$ *and thus* $\bigcap \mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{q\}) = \{\neg p, \neg p \supset r, \neg q \supset r\}$*, while*
- $\langle\emptyset, \mathcal{S} \cup \{p \vee q\}\rangle \not\mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} r$*,*
  since $\mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{p \vee q\}) = \{\{p \vee q, \neg p, \neg p \supset r, \neg q \supset r\}, \{p \vee q, \neg q, \neg p \supset r, \neg q \supset r\}, \{\neg p, \neg q, \neg p \supset r, \neg q \supset r\}\}$ *and thus* $\bigcap \mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{p \vee q\}) = \{\neg p \supset r, \neg q \supset r\}$.

Example 93 (Counter-example, $\sqcup$-**RM**, $\mathrel{|\!\sim}_{\Cap\mathsf{mcs}}$)**.** *Let $\mathfrak{L}$ be classical logic and $\mathcal{S} = \{r, p \wedge q \wedge \neg r, (p \wedge r) \supset \neg q, \neg p \wedge q\}$. We have* $\mathsf{MCS}^{\emptyset}(\mathcal{S}) = \{\{r, (p \wedge r) \supset \neg q, \neg p \wedge q\}, \{p \wedge q \wedge \neg r, (p \wedge r) \supset \neg q\}\}$. *One of the two elements of* $\mathsf{MCS}^{\emptyset}(\mathcal{S})$ *does not imply $\neg p$, while both of them imply q. Thus, $\langle\emptyset, \mathcal{S}\rangle \mathrel{|\!\sim}_{\Cap\mathsf{mcs}} q$ and $\langle\emptyset, \mathcal{S}\rangle \not\mathrel{|\!\sim}_{\Cap\mathsf{mcs}} \neg p$.*

Now, consider $\langle\emptyset, \mathcal{S} \cup \{p\}\rangle$ and $\langle\{p\}, \mathcal{S}\rangle$. We have:

- $\mathsf{MCS}^{\emptyset}(\mathcal{S} \cup \{p\}) = \{\{r, (p \wedge r) \supset \neg q, \neg p \wedge q\}, \{p \wedge q \wedge \neg r, (p \wedge r) \supset \neg q, p\}, \{r, p, (p \wedge r) \supset \neg q\}\}$ *and*
- $\mathsf{MCS}^{\{p\}}(\mathcal{S}) = \{\{p \wedge q \wedge \neg r, (p \wedge r) \supset \neg q\}, \{r, (p \wedge r) \supset \neg q\}\}$.

As a consequence, $\langle\emptyset, \mathcal{S} \cup \{p\}\rangle \not\mathrel{|\!\sim}_{\Cap\mathsf{mcs}} q$ and $\langle\{p\}, \mathcal{S}\rangle \not\mathrel{|\!\sim}_{\Cap\mathsf{mcs}} q$. Thus, neither $\uplus$-**RM** *nor* $\Cup$-**RM** *holds in this case.*

Not so many results on inferential properties are known for fragments of ASPIC$^+$ and ABA that are beyond those that coincide with reasoning with maximally consistent subsets. To the best of our knowledge, for ABA frameworks, inferential behavior for these fragments has only been studied in [126], where the following results are shown:

Remark 94. *For flat ABFs that are not necessarily simple contrapositive but whose strict rule set is contrapositive (see Remark 49), [126] show the following additional results:*

- $\mid\!\sim_{\Cap\mathsf{Grd}}$ *satisfies* $\uplus$-**CM** *and* $\uplus$-**CC**
- $\mid\!\sim_{\Cap\mathsf{Prf}}$ *satisfies* $\uplus$-**CC**
- *if* $\mathcal{ABF}$ *is well-behaved (recall Remark 49), then* $\mid\!\sim_{\Cap\mathsf{sem}}$ *satisfies* $\uplus$-**CM** *for* $\mathsf{sem} \in \{\mathsf{Prf}, \mathsf{Stb}\}$.[43]

Another study of inferential behavior of assumption-based argumentation is given in [74] (in [76] it is extended to ABA$^+$), where yet another set of postulates is studied. For example, cautious cut and cautious monotony are defined in [74] as follows:

Definition 95. Given $\mathcal{ABF} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \sim \rangle$, for an arbitrary extension $\mathcal{E} \in \mathsf{Sem}(\mathcal{ABF})$ $\mathcal{L}$-formula $\phi \notin \mathcal{A}$, and $\sqcup \in \{\Cup, \uplus\}$, we define:

$\sqcup$**-SCC:** If $\phi \in Cn_{\vdash_\mathcal{R}}(\mathcal{E})$, then for every $\mathcal{E}' \in \mathsf{Sem}(\mathcal{ABF} \sqcup \phi)$, $Cn_{\vdash_\mathcal{R}}(\mathcal{E}) \subseteq Cn_{\vdash_\mathcal{R}}(\mathcal{E}')$.

$\sqcup$**-WCC:** If $\phi \in Cn_{\vdash_\mathcal{R}}(\mathcal{E})$, then for some $\mathcal{E}' \in \mathsf{Sem}(\mathcal{ABF} \sqcup \phi))$, $Cn_{\vdash_\mathcal{R}}(\mathcal{E}) \subseteq Cn_{\vdash_\mathcal{R}}(\mathcal{E}')$.

$\sqcup$**-SCM:** If $\phi \in Cn_{\vdash_\mathcal{R}}(\mathcal{E})$, then for every $\mathcal{E}' \in \mathsf{Sem}(\mathcal{ABF} \sqcup \phi))$, $Cn_{\vdash_\mathcal{R}}(\mathcal{E}) \supseteq Cn_{\vdash_\mathcal{R}}(\mathcal{E}')$.

$\sqcup$**-WCM:** If $\phi \in Cn_{\vdash_\mathcal{R}}(\mathcal{E})$, then for some $\mathcal{E}' \in \mathsf{Sem}(\mathcal{ABF} \sqcup \phi))$, $Cn_{\vdash_\mathcal{R}}(\mathcal{E}) \supseteq Cn_{\vdash_\mathcal{R}}(\mathcal{E}')$.

It can be shown that, for each $\sqcup \in \{\Cup, \uplus\}$, $\cup$-**CC** and $\sqcup$-**CM**, defined for $\mid\!\sim_{\cup\mathsf{Sem}}$, imply, respectively, $\sqcup$-**WCC** and $\cup$-**WCM** (and, obviously, $\sqcup$-**SCC** and $\sqcup$-**SCM** also respectively imply the two latter rules).

The following proposition and examples are shown in [74]:

Proposition 96. *For each* $\sqcup \in \{\Cup, \uplus\}$,

- *grounded semantics satisfies* $\sqcup$-**SCC** *and* $\sqcup$-**SCM**,
- *preferred and stable semantics satisfy* $\sqcup$-**WCC** *and* $\sqcup$-**WCM**.

Here are counter-examples to $\uplus$-**SCC** and $\uplus$-**SCM** for preferred and stable semantics:

[43] The satisfaction of the postulates for $\mid\!\sim_{\cap\mathsf{sem}}$ and $\mid\!\sim_{\cup\mathsf{sem}}$-entailments are not studied in [126], and neither is satisfaction of properties such as $\uplus$-**REF**, $\uplus$-**LLE**, **RW** or $\uplus$-**OR**. The same holds for any of the $\Cup$-properties.

Example 97. *Let $\mathcal{ABF} = \langle \{p, q, r, p', q', r', s\}, \mathcal{R}, \mathcal{A}, \sim\rangle$ with*

$$\mathcal{A} = \{p, q, r\},$$

$$\mathcal{R} = \{p \to q';\ r \to p';\ q \to p';\ q \to s;\ s \to r'\}, \text{ and}$$

$$\sim x = \{x'\} \text{ for any } x \in \mathcal{A}.$$

*A fragment of the attack diagram of this ABF is given in Figure 8a. Here $\{q\}$ is the unique preferred and stable extension and $\{q\} \vdash_{\mathcal{R}} s$. Consider now $\mathcal{ABF} \uplus \{s\}$ (see Figure 8b for a fragment of the attack diagram). Now there are two preferred (and stable) extensions: $\{q\}$ and $\{p\}$. Since $Cn_{\mathcal{R}}(\{p\}) \nsubseteq Cn_{\mathcal{R}}(\{q\})$, it follows that $\uplus$-**SCM** is violated. Likewise, since $Cn_{\mathcal{R}}(\{p\}) \nsupseteq Cn_{\mathcal{R}}(\{q\})$, it follows that $\uplus$-**SCC** is violated.*

*Notice that this example is also a counter-example to $\uplus$-**CM** for $\mathrel{|\!\sim}_{\Cap\mathsf{Sem}}$ with $\mathsf{Sem} \in \{\mathsf{Prf}, \mathsf{Stb}\}$, as $\mathcal{ABF} \mathrel{|\!\sim}_{\Cap\mathsf{Sem}} s$ and $\mathcal{ABF} \mathrel{|\!\sim}_{\Cap\mathsf{Sem}} q$, yet $\mathcal{ABF} \uplus \{s\} \not\mathrel{|\!\sim}_{\Cap\mathsf{Sem}} q$.*

Here are counter-examples to $\Cup$-**SCC** and $\Cup$-**SCM** for the preferred semantics:

Example 98. *Let $\mathcal{ABF}$ be as in Example 97. Observe that:*

$$\mathcal{ABF} \Cup \{s\} = \langle \{p, q, r, s, p', q', r', s'\}, \mathcal{R}, \mathcal{A}, \sim\rangle, \text{ with}$$

$$\mathcal{A}' = \{p, q, r, s\},$$

$$\mathcal{R}' = \{p \to q';\ r \to p';\ q \to p';\ s \to r'\}, \text{ and}$$

$$\sim x = \{x'\} \text{ for any } x \in \mathcal{A}.$$

A fragment of the attack diagram of this ABF is given in Figure 8c.

*The framework $\mathcal{ABF} \Cup \{s\}$ has two preferred (and stable) extensions: $\{q, s\}$ and $\{p, s\}$. In this case $\Cup$-**SCM** is violated, since $Cn_{\mathcal{R}}(\{q\}) \nsubseteq Cn_{\mathcal{R}}(\{p, s\})$. Likewise, $\Cup$-**SCC** is violated, since $Cn_{\mathcal{R}}(\{q\}) \nsupseteq Cn_{\mathcal{R}}(\{p, s\})$.*

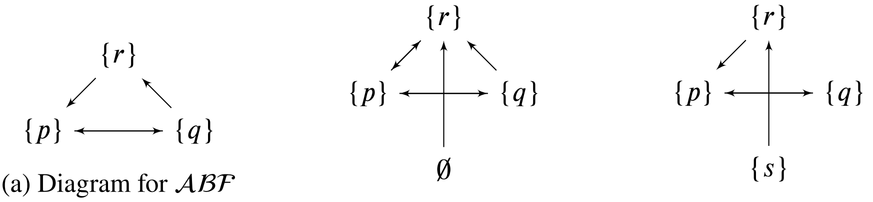
*As in Example 97, this example can also be seen to be a counter-example to $\Cup$-**CM**.*

In [135], inference properties are studied for ASPIC$^+$. However, right weakening, left logical equivalence and reflexivity are defined there in a different way. In more detail, [135] study the following alternative versions of these rules:

Definition 99 (alternative inference properties)**.** *Given an ASPIC argumentation theory $AT = \langle\langle \mathcal{L}, \bar{\ }, \mathcal{R}, n\rangle, (\mathcal{K}_n, \mathcal{K}_p)\rangle$, $\mathcal{L}$-formuals ϕ, ψ, an operator $\sqcup \in \{\uplus, \Cup\}$ and an entailment relation $\mathrel{|\!\sim}$ as in Definition 26, we say that $\mathrel{|\!\sim}$ satisfies:*

REF^d *if $\phi \in \mathcal{K}_p$ then $AT \mathrel{|\!\sim} \phi$*

REF^s *if $\phi \in \mathcal{K}_n$ then $AT \mathrel{|\!\sim} \phi$*



(a) Diagram for $\mathcal{ABF}$

(b) Diagram for $\mathcal{ABF} \uplus \{s\}$

(c) Diagram for $\mathcal{ABF} \Cup \{s\}$

Figure 8: Attack diagrams for Examples 97 and 98. To avoid clutter only attacks from minimal sets are included.

RW^d *if* $AT \mathrel{|\!\sim} \phi$ *and* $\phi \Rightarrow \psi \in \mathcal{R}_d$ *then* $AT \mathrel{|\!\sim} \psi$

RW^s *if* $AT \mathrel{|\!\sim} \phi$ *and* $\phi \rightarrow \psi \in \mathcal{R}_s$ *then* $AT \mathrel{|\!\sim} \psi$

$\sqcup$-**LLE**^{d} *if* $\phi \Rightarrow \psi \in \mathcal{R}_d$*,* $\psi \Rightarrow \phi \in \mathcal{R}_d$ *and* $AT \sqcup \phi \mathrel{|\!\sim} \sigma$ *then* $AT \sqcup \psi \mathrel{|\!\sim} \sigma$

$\sqcup$-**LLE**^{s} *if* $\phi \rightarrow \psi \in \mathcal{R}_s$*,* $\psi \rightarrow \phi \in \mathcal{R}_s$ *and* $AT \sqcup \phi \mathrel{|\!\sim} \sigma$ *then* $AT \sqcup \psi \mathrel{|\!\sim} \sigma$

Notice that **RW** implies **RW**^{s} and $\sqcup$-**LLE** implies $\sqcup$-**LLE**^{s} (for any $\sqcup \in \{\uplus, \Cup\}$), **REF**^{s} implies $\uplus$-**REF** (but not vice versa) and **REF**^{d} implies $\Cup$-**REF** (but not vice versa).

The main positive results of [135] are the following:

Proposition 100.

- $\mathrel{|\!\sim}_{\Cap\mathsf{Grd}}$ *satisfies* **REF**^{s}*,* **RW**^{s}*,* **LLE**^{s}*,* $\Cup$-**CM** *and* $\Cup$-**CC***.*
- $\mathrel{|\!\sim}_{\Cap\mathsf{Prf}}$ *satisfies* **REF**^{s}*,* **RW**^{s}*,* **LLE**^{s} *and* $\Cup$-**CC***.*
- $\mathrel{|\!\sim}_{\cap\mathsf{Prf}}$ *and* $\mathrel{|\!\sim}_{\cup\mathsf{Prf}}$ *satisfy* **REF**^{s}*,* **RW**^{s} *and* **LLE**^{s}*.*

We conclude this section by making some observations on both the significance of satisfaction or violations of the properties discussed in this section and the current state of the art. On one hand, there is a long tradition in non-monotonic logic which claims or assumes the properties for cumulative inference relations to "constitute a basic set of principles that any reasonable account of defaults must obey" [108]. As such, the satisfaction of such properties can be seen as a minimal condition on any formalization of non-monotonic reasoning. However, the generality of this claim has been put into doubt by, e.g. Bochman [41; 42; 43], who posits a distinction between *explanatory* and *preferential* reasoning, where only for the latter cumulativity is feasible. Furthermore, some of the properties considered in this section are not outside of controversy, such as rational monotony (cf., for instance, [163]). In sum, we submit that the feasibility of the postulates for non-monotonic reasoning depends on the precise context of application. Once this is decided, the results in this section offer some indications of which formalisms are appropriate for specific needs.

Finally, it is evident from this survey that the formalizations of the properties differ greatly in different works, making it difficult to compare results and transfer them between systems. Therefore, we think that it is an important direction for future work to study the relations between the different formulations of the properties studied in this section, and – more generally – to express some other criteria for relating and comparing the different approaches to logic-based argumentations, as well as their relations to other forms of non-monotonic reasoning. Some steps in this direction are reviewed in the next section.

2.4 Comparative Study

In this section we review some results concerning the inter-relations among the three logic-based approaches to formal argumentation considered in Section 2.2, as well as some of their connections to related methods to defeasible reasoning.

2.4.1 Relations among the Logic-Based Approaches

From the descriptions of logic-based argumentation, assumption-based argumentation and ASPIC$^+$ given above, the similarities of the frameworks are clear: they all use the same *pipeline-methodology* where an argumentation framework is constructed from the following components:

- a core (base) logic that determines the underlying language and the consequence relation for the arguments,
- attack rules relating arguments with counterarguments,
- a knowledge-base, encoding the set of the 'global' assumptions of the framework,
- an argumentation semantics, according to which sets of jointly acceptable arguments and their respective accepted conclusions are determined.

However, the formalisms outlined in Section 2.2 clearly differ in the specific ways formal substance is given to this general methodology. Table 14 gives an overview of the specific instantiations of the main argumentative concepts by logic-based argumentation (LBA), assumption-based argumentation (ABA) and ASPIC$^+$.

An important question that arises in such a comparison is concerned with the impact of the different choices on the resulting inference relation. Such a question can be partly answered by considering the exact relationship between the formalisms under consideration. This can be done in several ways, for instance by

Concept	LBA	ABA	ASPIC$^+$
Knowledge-base	$\mathcal{S}$ and $\mathfrak{L}$	$\langle \mathcal{L}, \mathcal{R}_s, \mathcal{K}_p, \sim \rangle$	$\langle \mathcal{L}, ^-, \mathcal{R}, n \rangle$, $\langle \mathcal{K}_n, \mathcal{K}_p \rangle$
Arguments	support-conclusion pairs	sets of assumptions	proof trees
Attacks	various	direct defeat	undermining, rebut, undercut

Table 14: Argumentative concepts and their instantiations in logic-based frameworks

1. comparing the inference relations associated with the respective formalisms,
2. investigating translations between the different formalisms, and
3. comparing the relative expressivity of the different formalisms.

Several works, including [150; 11; 117; 48; 121; 126], have concluded that logic-based argumentation, assumption-based argumentation and ASPIC$^+$ agree on what we could call a *core fragment*, namely when the underlying (strict) base logic is classical logic (or even any contrapositive Tarskian logic), and the defeasible assumptions are some propositional formulas. Indeed, it follows from Propositions 16, 46 and 48 that all three frameworks give rise to the same inference relation for the above-mentioned fragment and that this core fragment coincides with MCS-based reasoning.

When moving away from this core fragment, the formalisms start to behave in fundamentally different ways. First, it should be noted that logic-based argumentation as represented here, is restricted to (usually contrapositive) Tarskian logics, where the knowledge-base consists of defeasible propositional formulas.[44] In contrast, ABA and ASPIC$^+$, do allow to use not only defeasible, but also strict assumptions. Moreover, ASPIC$^+$ allows to reason with defeasible rules in addition to defeasible premises, i.e., with ASPIC$^+$ one can make inferences from knowledge bases that ABA cannot handle.

As we will describe below, there are ways to express defeasible rules with the help of defeasible premises and strict rules, but it seems equally interesting to compare the inferential behavior of ABA and ASPIC$^+$ for knowledge bases whose only defeasible elements are premises. In [150, Corollary 8.10] it is shown that given a *flat* assumption-based framework

[44] We note that this restriction can be lifted by adding strict assumptions and applying the attack rules only on the defeasible arguments. See [48] for the details. Here we follow the main line of research so far that combines logic-based framework with defeasible information only.

$\mathcal{ABF} = \langle \mathsf{Atoms}(\mathcal{L}), \mathcal{R}, \mathcal{A}, \sim \rangle$ (i.e, when for no $\Theta \cup \{\theta\} \subseteq \mathcal{A}$, $\Theta \vdash_{\mathcal{R}} \theta$.), the ASPIC-based argumentation framework $\mathrm{AT}_{\mathcal{ABF}} = \langle \langle \mathsf{Atoms}(\mathcal{L}), \bar{\ }, \langle \mathcal{R}, \emptyset \rangle, n \rangle, \langle \emptyset, \mathcal{K} \rangle \rangle$ gives rise to the same inferences.

Proposition 101. *Let $\mathcal{ABF} = \langle \mathsf{Atoms}(\mathcal{L}), \mathcal{R}, \mathcal{A}, \sim \rangle$ be a flat assumption-based framework. Consider the ASPIC-based argumentation framework $AT_{\mathcal{ABF}} = \langle \langle \mathsf{Atoms}(\mathcal{L}), \bar{\ }, \langle \mathcal{R}, \emptyset \rangle, n \rangle, \langle \emptyset, \mathcal{K} \rangle \rangle$ for arbitrary n[45] and where $\bar{\ }$ is defined by $\overline{\phi} = \sim\phi$ for any $\phi \in \mathcal{A}$ and $\overline{\phi} = \emptyset$ otherwise. Then for any $\dagger \in \{\cup, \cap, \Cap\}$ and $\mathsf{Sem} \in \{\mathsf{Grd}, \mathsf{Prf}, \mathsf{Cmp}, \mathsf{Stb}\}$, $\mathcal{ABF} \mathrel{|\!\sim}_{\dagger\mathsf{Sem}} \psi$ iff $AT_{\mathcal{ABF}} \mathrel{|\!\sim}_{\dagger\mathsf{Sem}} \psi$.*

It follows that for knowledge-bases with a flat rule-base and any semantics subsumed by complete semantics ABA and ASPIC$^+$ provide the same inferences. However, for non-flat knowledge-bases, this correspondence breaks down, as demonstrated by the next example.

Example 102. *Let $\mathsf{Atoms}(\mathcal{L}) = \{p, q\}$, $\mathcal{R} = \{p \to q\}$, and $\mathcal{ABF} = \langle \{p, q\}, \mathcal{R}, \{p, q\}, \sim \rangle$ where $\sim p = \emptyset$ and $\sim q = \{q\}$. For this ABF, the unique preferred extension is $\emptyset$. Indeed, $\{p\}$ is not admissible since it is not closed (since $\{p\} \vdash_{\mathcal{R}} q$) and any set containing q is not admissible (since q attacks itself).*

If we move to ASPIC$^+$ we have the argumentation theory $AT_{\mathcal{ABF}} = \langle \langle \{p, q\}, \bar{\ }, \langle \mathcal{R}, \emptyset \rangle, n \rangle, \langle \emptyset, \{p, q\} \rangle \rangle$, and the arguments $A = \langle p \rangle$, $B = \langle q \rangle$, $C = A \to q$.

There is an attack from B to itself and from C to B. Notice furthermore that C is unattacked (Recall here that no rebuttals are possible in the heads of strict rules, which is why C does not rebut itself). This means that $\{A, C\}$ is the unique stable and preferred extensions.

It is perhaps interesting to note that $\{A, C\}$ presents a violation of the rationality postulate of consistency from [58] (see Section 2.3.2, and in particular definition 60). It is an open question if there are any differences in inferential behavior between ASPIC$^+$ and non-flat ABA for knowledge-bases whose extensions satisfy all the rationality postulates.

Translation methods. Given both the conceptual differences (as displayed in Table 14) and the diverging inferential behavior of LBA, ABA and ASPIC$^+$, the correspondences described above have been supplemented by *translations* among the formalisms. Particular attention has been paid to translations from ASPIC$^+$ into ABA. Conceptually, this corresponds to asking if one can model defeasible rules as defeasible premises. Such a question has been answered positively in [90] and [123], sharing the same underlying idea: given an ASPIC-based argumentation framework $\langle \mathcal{L}, \bar{\ }, \mathcal{R}_s \cup \mathcal{R}_d, n \rangle$, the underlying language $\mathcal{L}$ is extended to $\mathcal{L}'$ as to contain a name $N(r)$ for every $r \in \mathcal{R}_d$. This name is then added as

[45] Note that n can be safely ignored since the set of defeasible rules $\mathcal{R}_d$ is empty.

a defeasible assumption in the ABF.[46] The strict rule-base is then supplemented with rules that ensure that the names of the defeasible rules are handled adequately in the argumentative inference process. In particular, for every rule $r = \phi_1, \ldots, \phi_n \Rightarrow \psi \in \mathcal{R}_d$, the following rules are added (resulting in $\mathtt{R}(\mathcal{R}_d)$):[47]

- $N(r), \phi_1, \ldots, \phi_n \to \psi$, which ensures that ψ is (defeasibly) derivable from $\{\phi_1, \ldots, \phi_n\}$;
- $\overline{\psi} \to \overline{N(r)}$ which enables an attack on $N(r)$ if the contrary of the consequent of r is derivable (thus mirroring rebuttal);
- $\overline{n(r)} \to \overline{N(r)}$, which enables an attack on $N(r)$ if $\overline{n(r)}$ is derivable (thus mirroring undercut).

In [123] it is shown that this translation is adequate for *flat argumentation theories* for admissible, preferred and stable semantics. In [90], it is shown that their translation is adequate for any semantics subsumed by complete semantics. In the following, given a flat[48] argumentation theory $AT = \langle\langle \mathcal{L}, \bar{\ }, \mathcal{R}_s \cup \mathcal{R}_d, n\rangle, \langle \mathcal{K}_n, \mathcal{K}_n\rangle\rangle$, let

$$\mathcal{ABF}(AT) = \langle \mathcal{L}, \mathcal{R}_s \cup \mathtt{R}(\mathcal{R}_d) \cup \{\to \phi \mid \phi \in \mathcal{K}_n\}, \mathcal{K}_p \cup \{N(r) \mid r \in \mathcal{R}_d\}, \sim\rangle$$

We now recall the adequacy result from [123]

Proposition 103. *Given a flat argumentation theory AT, $\dagger \in \{\cap, \cup, \Cap\}$, and* $\mathsf{Sem} \in \{\mathsf{Prf}, \mathsf{Stb}\}$*: $AT \mathrel{|\!\sim}_{\dagger\mathsf{Sem}} \phi$ iff $\mathcal{ABF}(AT) \mathrel{|\!\sim}_{\dagger\mathsf{Sem}} \phi$.*

No adequate translation is known for non-flat argumentation theories.

Expressivity, Complexity and Representation of Arguments. A third way to compare the logic-based approaches to formal argumentation considered in this chapter is by studying their expressiveness. In other words, one may compare the answers to the question: "what kind of problems can be solved by this formalism" [165]. In terms of feasibility, this often boils down to questions of computational complexity. In that respect, we note that while the complexity of ABA has been studied in [83], for LBA and ASPIC$^+$ similar complexity results are missing. As noted in [147], the complexity of these formalisms is indeed an important open question.

[46] In [90] the language is also extended with an atom $\mathsf{not}\psi$ for every $\phi_1, \ldots, \phi_n \Rightarrow \psi$ such that in the translated ABF, $\mathsf{not}\ \psi$ is a defeasible assumption similar to negation as failure.

[47] For simplicity, we denote by $\overline{\phi}$ any $\phi' \in \overline{\phi}$.

[48] An argumentation theory $AT = \langle\langle \mathcal{L}, \bar{\ }, \mathcal{R}_s \cup \mathcal{R}_d, n\rangle, \langle \mathcal{K}_n, \mathcal{K}_p\rangle\rangle$ is *flat* if there is no $A \in \mathsf{Arg}(AT)$ such that $\mathsf{Conc}(A) \in \mathcal{K}_p \setminus \mathtt{Prem}(A)$.

Another point of difference between the formalisms is related to how exactly arguments are represented. In ASPIC$^+$ and logic-based argumentation, arguments are formed for specific conclusions. In ABA, on the other hand, nodes of an argumentation graph are made up of sets of assumptions, without a specific conclusion. In this sense, ABA can be said to operate on the level of equivalence classes of arguments with the same support. For this reason, given a finite set of defeasible assumptions, ABA will give rise to an argumentation graph bounded by the size of the power set of the set of defeasible assumptions. Logic-based argumentation and ASPIC$^+$, on the other hand, might still generate an infinite argumentation graph since the underlying base logic might generate an infinite set of conclusions for every set of defeasible assumptions. On the other hand, this also means that in ASPIC$^+$ and logic-based argumentation, all the possible conclusions are present in the argumentation graph, whereas in ABA these conclusions have to still be derived. Altogether, we can summarize this difference as follows: ABA represents arguments in a more compact way, which has both positive aspects (e.g. boundedness of the argumentation graph) and negative aspects (e.g. some information might not be readily present in the argumentation graph). In [5], a procedure is developed to compute a finite *core* of a logic-based argumentation system, which returns all the results of the original system. Similarly, in [16] congruence relations (and their corresponding structures) are discussed for argumentation frameworks in the context of sequent-based argumentation, e.g., based on equivalent support sets of arguments. For ASPIC$^+$, the problem of having infinite number of arguments out of a finite set of assumptions is avoided in [77; 78] in the context of dialectical argumentation frameworks and depth-bounded logics. This approach involves preferences among arguments and is concentrated on classical logic as the base logic of the framework.

2.4.2 Connections to Other Approaches

Next, we discuss relations between the logic-based argumentation formalisms presented in this chapter and other formalisms for defeasible reasoning. Clearly, it is not possible to formally and fully define here all the related formalisms, thus in what follows we just give some general description of each related formalism, together with some references for further reading. This means also that we will not be able to express the relations between the formalisms in detail, but instead we shall provide the general underlying ideas and references to papers where the relations are fully described.

It was arguably one of the goals of Dung in [85] to show that the way conflicts are handled in abstract argumentation theory correspond to the way conflicts are handled in many different kinds of formalisms for defeasible reasoning. In [85], Dung showed that this is the case by proving *representation results* for several formalisms for defeasible reasoning. He showed how to construct argumentation graphs for several such formalisms in a way that is both intuitive and gives rise to an adequate representation when applying the abstract argumentation semantics to the resulting argumentation graph.

Since then, various additional argumentative characterizations of formalisms for defeasible reasoning have been proposed. We have already mentioned in Section 2.3.1 argumentative characterizations of reasoning with maximal consistent subsets [155] by logic-based argumentation, assumption-based argumentation and ASPIC$^+$. In the rest of this section we use these formalisms for argumentative characterizations of *adaptive logics* [26; 167], *default assumptions* [138], *logic programming* [8], *default logic* [154] and *autoepistemic logic* [148]. An illustration of these relations in given in Figure 9 at the end of this section.

A. Adaptive Logics Adaptive logics offer a general framework for defeasible reasoning. A plethora of forms of defeasible reasoning has been explicated in the adaptive logic framework. Some examples are: the modeling of abduction (e.g., [142; 107]), inductive generalization (e.g., [27; 25]), default reasoning (e.g., [166]), reasoning from incompatible obligations (e.g., [29; 174]), causal discovery (e.g., [175]), reasoning with vague predicates (e.g., [176]), diagnostic reasoning (e.g., [182]), etc.

Adaptive logics come with a dynamic proof theory extending a Tarskian core logic with a set of retractable inferences which are associated with defeasible assumptions. More specifically, these assumptions are sets of formulas of a predefined 'abnormal' form that are assumed to be false in the given inference. When an assumption turns out to be dubious in view of a premise set, the inference associated with it gets retracted.

Semantically, adaptive logics are based on preferential semantics that are adequate relative to the dynamic proof theory. Given a Tarskian core logic $\mathfrak{L}$, not all the $\mathfrak{L}$-models of the premises are considered when determining the consequences, but only a sub-class is "selected", namely those models which are "sufficiently normal". Different types of adaptive logics follow different *strategies* that offer specifications of what it means to be sufficiently normal. For instance, in adaptive logics that follows the minimal abnormality strategy, those models are selected for which there are no models that verify less abnormal formulas.

As shown in [123], there is a straightforward translation of the framework of adaptive logics into ABA: given an adaptive logic $\mathbf{AL} = \langle \mathfrak{L}, \Omega \rangle$, where $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ is a Tarskian logic and $\Omega \subseteq \mathcal{L}$ is a set of abnormalities, and a set of premises Γ, the corresponding ABF is defined as $\mathcal{ABF}_{\mathbf{AL}} = \langle \mathfrak{L}, \Gamma, \{\neg\phi \mid \phi \in \Omega\}, \sim \rangle$, where $\sim\neg\phi = \phi$. It is shown that for preferred, naive and stable semantics, this translation is adequate to represent different types of adaptive strategies.

B. Logic Programming Logic programming (LP) is one of the most popular approaches to knowledge representation and has been widely studied, implemented and applied [8].

(Propositional) logic programs are set of rules of the form:

$$\phi_1 \vee \ldots \vee \phi_n \leftarrow \psi_1, \ldots, \psi_m, {\sim}\psi_{m+1}, \cdots {\sim}\psi_{m+l}$$

where ϕ_i, ψ_j are formulas for any $1 \leq i \leq n$ and $1 \leq j \leq m + l$. The left-hand side of the implication is call the rule's *head* and the right-hand side of the implication is the rule's *body*. Now,

- If in all the rules of the program, every ϕ_i $(1 \leq i \leq n)$ and ψ_j $(1 \leq j \leq m + 1)$ is atomic, the program is called a *disjunctive logic program*, and

- If, in addition, $n \leq 1$ for every rule in the program, the program is called *normal*.

There are many ways of giving semantics to logics programs. One of the better-known one is based on the notion of a *reduct*, which is a set of rules that is calculated on the basis of a set of atoms. For example,

> the *Gelfond-Lifschitz reduct* [109] $\frac{\mathcal{P}}{\Delta}$, of a normal logic program $\mathcal{P}$ with respect to a set of atoms Δ, is constructed as follows: $\phi \leftarrow \psi_1, \ldots, \psi_m \in \frac{\mathcal{P}}{\Delta}$ iff $\phi \leftarrow \psi_1, \ldots, \psi_m, {\sim} \psi_{m+1}, \cdots {\sim}\psi_{m+l} \in \mathcal{P}$ and $\psi_i \notin \Delta$ for any $m < i \leq m + l$.

Based on such a reduct, the semantics of logic programming then describe ways to select sets of atoms which count as models. For example,

> the *stable model semantics* says that a set of atoms is a stable model if it is the minimal model of its own Gelfond-Lifschitz reduct.[49]

The translation of logic programming into assumption-based argumentation has been the subject of several publications (e.g., [157; 89; 65; 118]). The basic idea underlying all of these publications is the same: the set of assumptions is made up of negated atoms, and the contrary of a negated atom is the positive atom. The (strict) rules consist of the rules of the logic programs. Thus, a set of negated atoms will attack a negated atom if the logic program and the attacking set allows to derive the positive version of the attacked negated atom. Therefore, the underlying idea is to assume the 'absence' of any atom A appearing in the logic program (the defeasible assumptions), unless, on the basis of attacks derived by the programs rules, some set of assumptions indicates that A holds.

The correspondence results in Table 15 where proven in [65] for normal logic programs.

Remark 104. *It is interesting to note that L-stable models (i.e. 3-valued stable models that are maximal w.r.t. atoms assigned a definite truth value) do* not *correspond to semi-stable sets of assumptions (see [65, Example 13]), although both of these semantics are based on the same idea of maximizing the assignment of determinate truth values.*

[49] That is, Δ is a stable model of $\mathcal{P}$ if for every $p \leftarrow q_1, \ldots q_n \in \frac{\mathcal{P}}{\Delta}$, either $p \in \Delta$ or $q_i \notin \Delta$ for some $1 \leq i \leq n$, and there is no $\Delta' \subsetneq \Delta$ with the same property.

ABA Extension	LP Model
complete	stable (3-valued)
grounded	well-founded
preferred	regular
stable	stable (2-valued)
ideal	ideal

Table 15: Correspondence between model of normal logic programs and extensions of ABA frameworks

The results above were extended in [118] to disjunctive logic programming under stable model semantics. Furthermore, argumentative characterizations of the so-called *well-justified* [159] and *well-founded* [181] semantics of *general* or *first-order* logic programs (i.e., logic programs where any first-order formula can occur in the head or the body of a rule) are provided in [89]. These generalizations are based on the same idea as [65]: the assumptions consist of negated atoms and attacks occur when the attacking set allows to derive the positive version of the attacked (negated) atom. What changes, however, is the derivability relation used to determine if attacks occur. For example, in [118] in addition to allowing for *modus ponens* on the rules of the program as in [65], one has also to allow for *reasoning by cases* and *resolution* in the derivations. Likewise, in [89] both *modus ponens* on the rules of the program and any deduction valid in first-order logic are allowed. In [180] *extended logic programs* [109] under three- and two-valued stable semantics are translated into assumption-based argumentation. These translations have been used to obtain explanations of (non-)derivability of literals in [158] and explaining and characterizing inconsistencies of logic programs [156].

C. Default Logics Reiter's default logic [154] has also been translated in assumption-based argumentation. Again, here we just we recall the basics of default logic in an informal way. Defaults are objects of the form

$$\frac{\phi \ : \ M\psi_1, \ \dots \ , M\psi_n}{\psi}.$$

Here, $\phi, \psi_1, \dots, \psi_n, \psi$ are formulas in the language, and the intuitive meaning of this expression is the following:

> if ϕ holds, and none of $\neg\psi_1, \dots, \neg\psi_n$ is provable, then normally one may suppose that ψ also holds.

An *extension* of a set of defaults Δ is a set of formulas Θ, such that Θ is a fixed point under the operator ∇_Δ, i.e., $\nabla_\Delta(\Theta) = \Theta$, where the operator ∇_Δ is defined as follows: given a set Θ, $\nabla_\Delta(\Theta)$ is the smallest set such that:

1. for every $\frac{\phi : M\psi_1,\ldots,M\psi_n}{\psi} \in \Delta$, if $\phi \in \Theta$ and $\neg\psi_i \notin \Theta$ for every $1 \leqslant i \leqslant n$, then $\psi \in \nabla_\Delta(\Theta)$, and
2. $\nabla_\Delta(\Theta) = Cn(\nabla_\Delta(\Theta))$,

The translation into ABA proposed in [46] works as follows: the language is that of classical logic extended with $M\phi$ for any $\phi \in \mathcal{L}$. The assumptions are $M\phi$ for any $\phi \in \mathcal{L}$, i.e., we assume (defeasibly) that for any formula $\phi \in \mathcal{L}$, its negation is not provable. The rules are generated by taking the default rules together with a set of rules that captures (classical) first-order logic. Finally, the contrary of $M\phi$ is defined as $\neg\phi$ (recall that $M\phi$ is interpreted as $\neg\phi$ not being provable): a positive proof of $\neg\phi$ gives us a counter-argument to the assumption $M\phi$.

In [46] it is shown that under this translation, stable extensions in ABA correspond to Reiter's default extensions. An interesting open question is whether similar results hold for other semantics for default logic, such as those from [55; 6; 81].

D. Autoepistemic Logics Moore's autoepistemic logic [148] is another well-established formalisms for defeasible reasoning. It involves theories consisting of formulas in a doxastic language, which is typically the closure $\mathcal{L}^{\mathbf{L}}$ of a propositional language $\mathcal{L}$ under a belief operator $\mathbf{L}$. The intuitive meaning of $\mathbf{L}\phi$ is that 'ϕ is believed'. Thus, autoepistemic logic is a formal logic for the representation and reasoning of knowledge about knowledge, and theories containing formulas of the form $\mathbf{L}\phi$ are viewed are representing "knowledge of a perfect, rational, introspective agent" [148; 132; 45]. An autoepistemic theory $\Delta \subseteq \mathcal{L}^{\mathbf{L}}$ represents both positive and negative introspection of a logically perfect agent, i.e., $\phi \in \Delta$ iff $\mathbf{L}\phi \in \Delta$ and $\phi \notin \Delta$ iff $\neg\mathbf{L}\phi \in \Delta$. Autoepistemic logic has been shown to have connections to many other formalisms for defeasible reasoning, such as several variants of default and priority logic [130], and several classes of logic programming [141].

A translation of autoepistemic logics to ABA frameworks is provided in [46]. According to this translation, the set of assumptions is made up of the assumption of both negative and positive knowledge: $Ab = \{\mathbf{L}\phi, \neg\mathbf{L}\phi \mid \phi \in \mathcal{L}\}$. Thus, both negative and positive knowledge are assumed equally plausible. However, there are asymmetric treatments when it comes to the definition of contraries: the contrary of positive knowledge $\mathbf{L}\phi$ is the negative knowledge (or absence of knowledge) of $\neg\mathbf{L}\phi$ (i.e., $\overline{\mathbf{L}\phi} = \neg\mathbf{L}\phi$). The contrary of absence of knowledge of a formula, on the other hand, is the formula itself, that is: $\overline{\neg\mathbf{L}\phi} = \phi$. The rule-base is a set of rules capturing first-order logic, but formulated over the modal language

$\mathcal{L}^{\mathbf{L}}$. It is interesting to note, however, that within the rule-base, no rules for the modal operator are defined. Under this translation, the strict premises consist of the autoepistemic theory Δ. [46] shows that stable extensions of the translation in ABA correspond to the so-called *consistent stable expansions* [148] of the translated autoepistemic theory. For other semantics, no correspondences are known.

Figure 9 provides a schematic description of the relations among the formalisms described in this section.

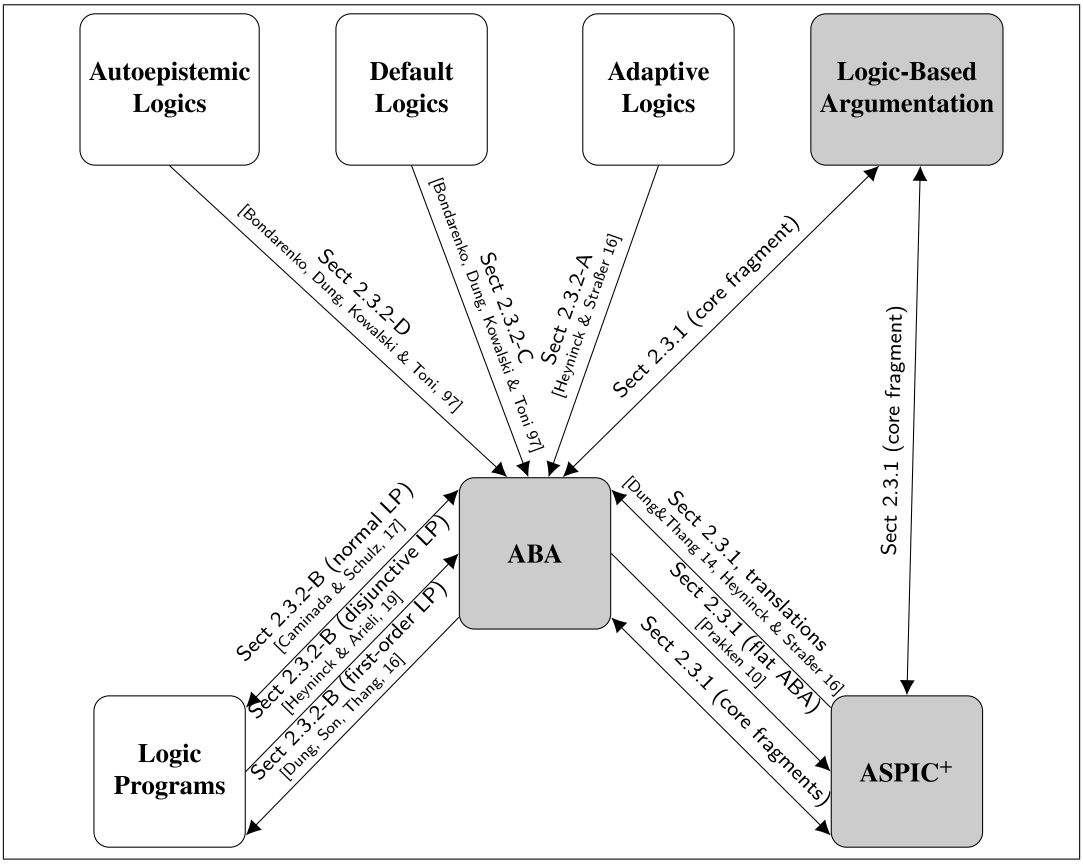


Figure 9: Argumentative representations of formalisms for modeling defeasible reasoning, presented in Sectrion 2.4

Besides the translations discussed above, we mention the following additional translations which are beyond the scope of this paper:

- In [48] a generalization of sequent-based argumentation, called *assumptive sequent-*

based argumentation, is shown to capture assumption-based argumentation, adaptive logics and default assumptions.

- We note that in [65] it is also shown that assumption-based argumentation can be translated in logic-programming.
- In [64] translations from normal logic programming to abstract argumentation and vice-versa have been presented which are adequate for most (but not all) argumentation semantics.
- In [120] it is shown that *approximation fixpoint theory* [80], a general approach to the study of non-monotonic reasoning, can be translated into assumption-based argumentation. This allows for the straightforward translation of many semantic variations on logic programming, default logic and auto-epistemic logic into assumption-based argumentation.
- Relationships (and further references) of ASPIC$^+$ to defeasible logic programming [106], classical logical argumentation frameworks (see the paragraph below Definition 8) and prioritized formalisms, such as Brewka's preferred subtheories [52] and prioritized default logic [53], are described in [146; 147].
- Translations of *abstract dialectical frameworks* [54] into logic programming respectively system Z [108] are shown in [164] respectively [122].

3 Logical Methods for Studying Argumentation Dynamics

There are a variety of methods for studying the dynamics of argumentation systems.[50] This includes, among others, dialectic games (see [144]), discussions [58], and, to some extent, even machine learning algorithms [56]. Other approaches involve formal (logic) programming methods, such as reductions to answer set programs (ASP), defeasible logic programs (DeLP) and constraint satisfaction problems (CSP) (see, e.g., [68] for a description of these methods and further references).

The common ground of the methods that are described in this section (following the scope of this chapter) is that all of them assume the availability of an underlying Tarskian logic and apply related formal methods (e.g., satisfiability of formulas in the underlying language or proof procedures that allow to make inferences by derivation sequences). In the first two subsections (3.1 and 3.2) we survey several logic-basic representation methods that

[50] Recall that 'dynamics' means here processes of a (fixed) argumentative framework and not its revision.

are adequate for expressing the selection of arguments in view of argumentation semantics and epistemic notions such as beliefs and their justifications in an argumentative setting. In the last subsection (3.3) we consider proof-theoretic methods that are adequate for structured argumentation.

3.1 Representation Methods Based on [Quantified] Propositional Languages

As indicated in, e.g., [33] and [94], given a finite argumentation framework, computing its admissible sets or its complete extensions can be done by a straightforward encoding, in propositional classical logic, of the requirements in the fourth item of Definition 10. Indeed, given an abstract argumentation framework $\mathcal{AF}$, one may associate a propositional atom with every argument in $\mathcal{AF}$ (in what follows, to ease the notations, we shall use the same symbol for an argument and its propositional variable), and accordingly construct the following formula:

$$\mathsf{ADM}(\mathcal{AF}) = \bigwedge_{p \in \mathsf{Arg}} \Big(\big(p \supset \bigwedge_{(q,p) \in Attack} \neg q\big) \wedge \big(p \supset \bigwedge_{(q,p) \in Attack} (\bigvee_{(r,q) \in Attack} r)\big)\Big).^{51}$$

Clearly, the arguments of an admissible set of $\mathcal{AF}$ correspond to the atoms that are verified (i.e., those that are assigned the truth value 'true') by a model of $\mathsf{ADM}(\mathcal{AF})$ and, conversely, every model of $\mathsf{ADM}(\mathcal{AF})$ is associated with an admissible set of $\mathcal{AF}$, the elements of which correspond to the verified atoms of the model. Similar considerations hold for the following formula, representing the complete extensions of $\mathcal{AF}$:

$$\mathsf{CMP}(\mathcal{AF}) = \bigwedge_{p \in \mathsf{Arg}} \Big(\big(p \supset \bigwedge_{(q,p) \in Attack} \neg q\big) \wedge \big(p \leftrightarrow \bigwedge_{(q,p) \in Attack} (\bigvee_{(r,q) \in Attack} r)\big)\Big).$$

Another, more informative way, of representing admissible and/or complete extensions, is to turn to signed formulas (and so to an underlying three-valued semantics). By this, it is possible not only to identify the arguments in the extensions (those that are verified by the models of the formulas), but also identify the arguments that are attacked by the extensions (those that are falsified by the models of the formulas). Briefly, the idea is to associate every argument in the framework with a *pair* $\langle p^+, p^- \rangle$ of ("signed") atoms, the truth values of which describe the status of the associated argument: accepted (p^+ is verified, p^- is falsified),

[51] Recall that $\bigwedge \emptyset = \mathsf{T}$ (truth) and $\bigvee \emptyset = \mathsf{F}$ (falsity).

rejected (p^+ is falsified, p^- is verified), and undecided (both p^+ and p^- are falsified).[52] [53] [54]

Now, consider the following formula :

$$\mathsf{CMP}^{\pm}(\mathcal{AF}) = \bigwedge_{\langle p^+,p^-\rangle \in \mathsf{Arg}} \left\{ \begin{array}{lr} \Big((p^+ \wedge \neg p^-) \supset \bigwedge_{(\langle q^+,q^-\rangle,\langle p^+,p^-\rangle)\in Attack}(\neg q^+ \wedge q^-)\Big), & (1) \\ \Big((\neg p^+ \wedge p^-) \supset \bigvee_{(\langle q^+,q^-\rangle,\langle p^+,p^-\rangle)\in Attack}(q^+ \wedge \neg q^-)\Big), & (2) \\ \Big((\neg p^+ \wedge \neg p^-) \supset & \\ \qquad \big(\neg\big(\bigwedge_{(\langle q^+,q^-\rangle,\langle p^+,p^-\rangle)\in Attack}(\neg q^+ \wedge q^-)\big) \wedge & \\ \qquad\quad \neg\big(\bigvee_{(\langle q^+,q^-\rangle,\langle p^+,p^-\rangle)\in Attack}(q^+ \wedge \neg q^-)\big) \big)\Big), & (3) \\ \neg\Big(p^+ \wedge p^-\Big) & (4) \end{array} \right\}.$$

- the subformula denoted by (1) states that any argument that attacks an accepted argument must be rejected,
- the subformula denoted by (2) states that any rejected argument must be attacked by at least one accepted argument,
- the subformula denoted by (3) states that for undecided arguments the previous conditions do not hold,[55] and
- the subformula denoted by (4) states that an argument may be either accepted, rejected, or undecided (i.e., a fourth state depicted by $p^+ \wedge p^-$ is excluded).

The next proposition (proved in [13]) shows the one-to-one correspondence between the models of $\mathsf{CMP}^{\pm}(\mathcal{AF})$ and the complete extensions of $\mathcal{AF}$.

Proposition 105. *Let $\mathcal{AF} = \langle \mathsf{Arg}, Attack \rangle$ be an argumentation framework. Then:*

- *For every complete extension $\mathcal{E} \in \mathsf{Cmp}(\mathcal{AF})$ there is a model $\mathcal{M}$ of $\mathsf{CMP}^{\pm}(\mathcal{AF})$ such that*

[52] The superscripts + and − have several meaning in different contexts, as A^+ (respectively, A^-) denotes the set of arguments that are attacked by (respectively, that attack) A. This notational overloading will not cause any confusion in what follows. Signed formulas were used in the context of inconsistency-tolerant reasoning in [39].

[53] Again, we freely switch between an argument and the pair of atomic formulas that is associated with it, so a pair $\langle p^+, p^-\rangle$ of (signed) atoms also stands for an argument in the framework.

[54] For a representation in terms of four-valued semantics, where both p^+ and p^- may be verified, we refer to [9].

[55] These three subformulas state conditions that correspond to Caminada's *complete labeling* (see [23]). See also Remark 106.

- $\mathsf{In}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = t,\ \mathcal{M}(p^-) = f\} = \mathcal{E}$,
- $\mathsf{Out}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = f,\ \mathcal{M}(p^-) = t\} = \mathcal{E}^+$,
- $\mathsf{Undec}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = f,\ \mathcal{M}(p^-) = f\} = \mathsf{Arg} \setminus (\mathcal{E} \cup \mathcal{E}^+)$.

- *For every model $\mathcal{M}$ of* $\mathsf{CMP}^{\pm}(\mathcal{AF})$ *there is a complete extension* $\mathcal{E} \in \mathsf{Cmp}(\mathcal{AF})$ *such that*

  - $\mathcal{E} = \mathsf{In}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = t,\ \mathcal{M}(p^-) = f\}$
  - $\mathcal{E}^+ = \mathsf{Out}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = f,\ \mathcal{M}(p^-) = t\}$,
  - $\mathsf{Arg} \setminus (\mathcal{E} \cup \mathcal{E}^+) = \mathsf{Undec}(\mathcal{M}) = \{\langle p^+, p^-\rangle \mid \mathcal{M}(p^+) = f,\ \mathcal{M}(p^-) = f\}$.

Remark 106. The notations in the first bullet of Proposition 105 are not accidental, as they correspond to the three types of assignments (in, out, undec) of the complete labeling of $\mathcal{AF}$.[56] Moreover, as shown in [13], all the results in this section carry on to labeling semantics.

As an immediate consequence of the last proposition we get a representation of the stable extension of $\mathcal{AF}$. Indeed, as a stable extension is a set $\mathcal{E} \subseteq \mathsf{Arg}$ such that $\mathsf{Arg} = \mathcal{E} \cup \mathcal{E}^+$, by the last proposition we just have to add a requirement that $\mathsf{Undec}(\mathcal{M}) = \emptyset$ for every model $\mathcal{M}$ of a theory. This can be easily done by adding the following 'excluded middle' condition:

$$\mathsf{EM}^{\pm}(\mathcal{AF}) = \bigwedge_{\langle p^+,p^-\rangle \in \mathsf{Arg}} \left(p^+ \vee p^-\right)$$

Corollary 107. *Let* $\mathcal{AF} = \langle \mathsf{Arg}, \mathit{Attack}\rangle$ *be an argumentation framework. Then:*

- *For every* $\mathcal{E} \in \mathsf{Stb}(\mathcal{AF})$ *there is a model* $\mathcal{M}$ *of* $\mathsf{CMP}^{\pm}(\mathcal{AF}) \cup \{\mathsf{EM}^{\pm}(\mathcal{AF})\}$ *such that* $\mathsf{In}(\mathcal{M}) = \mathcal{E}$ *and* $\mathsf{Out}(\mathcal{M}) = \mathcal{E}^+$.

- *For every model* $\mathcal{M}$ *of* $\mathsf{CMP}^{\pm}(\mathcal{AF}) \cup \{\mathsf{EM}^{\pm}(\mathcal{AF})\}$ *there is a stable extension* $\mathcal{E} \in \mathsf{Stb}(\mathcal{AF})$ *such that* $\mathcal{E} = \mathsf{In}(\mathcal{M})$ *and* $\mathcal{E}^+ = \mathsf{Out}(\mathcal{M})$.

When it comes to other types of extensions like grounded or preferred extensions, propositional formulas in classical logic are not sufficient for the representation, since the definitions of such extensions involve qualitative or comparative considerations. One way of dealing with this is to incorporate quantifiers in the language. As is shown in [94; 13; 82; 9], for this purpose first-order languages are not necessary, and it is sufficient to remain in the propositional level, by using *quantified Boolean formulas*. For this, we extend the underlying language with universal and existential quantifiers $\forall, \exists$ over propositional variables.

[56] Labeling semantics for argumentation frameworks is described, e.g., in [23].

Intuitively, the meaning of a quantified Boolean formula (QBF) of the form $\exists p\, \forall q\, \psi$ is that there exists a truth assignment of p such that for every truth assignment of q, ψ is true. Clearly, every QBF is associated with a logically equivalent propositional formula, thus ultimately we are still at the propositional level. This may be formally defined as follows:

Definition 108 (QBF-related notions)**.** Consider a QBF Ψ.

- An occurrence of an atom p in Ψ is called *free* if it is not in the scope of a quantifier $\mathsf{Q}p$, for $\mathsf{Q} \in \{\forall, \exists\}$.
- We denote by $\Psi[\phi_1/p_1, \dots, \phi_n/p_n]$ the uniform substitution of each free occurrence of a variable (atom) p_i in Ψ by a formula ϕ_i, for $i = 1, \dots, n$, and denote by T and F the propositional constants for truth and falsity (respectively).[57]
- Valuations over QBFs are, as usual, functions that assign truth values to the propositional variables (the atomic formulas) in the QBFs, and are extended to complex formulas as follows:
  $\nu(\neg\psi) = \neg\nu(\psi)$,
  $\nu(\psi \circ \phi) = \nu(\psi) \circ \nu(\phi)$ for $\circ \in \{\wedge, \vee, \supset\}$,
  $\nu(\forall p\, \psi) = \nu(\psi[\mathsf{T}/p]) \wedge \nu(\psi[\mathsf{F}/p])$,
  $\nu(\exists p\, \psi) = \nu(\psi[\mathsf{T}/p]) \vee \nu(\psi[\mathsf{F}/p])$.

Preferred extensions of an argumentation framework $\mathcal{AF}$ with n arguments that correspond to the n pairs $\{\langle p_1^+, p_1^- \rangle, \dots, \langle p_n^+, p_n^- \rangle\}$ may now be represented by the following QBF:

$$\begin{aligned} \mathsf{PRF}^{\pm}(\mathcal{AF}) \;=\; & \mathsf{CMP}^{\pm}(\mathcal{AF})(p_1^+, p_1^-, \dots, p_n^+, p_n^-) \wedge \\ & \forall q_1^+, q_1^-, \dots, q_n^+, q_n^- \Big(\mathsf{CMP}^{\pm}(\mathcal{AF})(q_1^+, q_1^-, \dots, q_n^+, q_n^-) \supset \\ & \mathsf{INC}^{\pm}_{\subseteq}(p_1^+, p_1^-, \dots, p_n^+, p_n^-, q_1^+, q_1^-, \dots, q_n^+, q_n^-) \Big). \end{aligned}$$

Here, $\mathsf{CMP}^{\pm}(\mathcal{AF})(p_1^+, p_1^-, \dots, p_n^+, p_n^-)$ is the formula $\mathsf{CMP}^{\pm}(\mathcal{AF})$ considered previously, but with the free variables $p_1^+, p_1^-, \dots p_n^+, p_n^-$, and

$$\begin{aligned} & \mathsf{INC}^{\pm}_{\subseteq}(p_1^+, p_1^-, \dots p_n^+, p_n^-, q_1^+, q_1^-, \dots q_n^+, q_n^-) \;=\; \\ & \qquad \bigwedge\nolimits_i \Big((p_i^+ \wedge \neg p_i^-) \supset (q_i^+ \wedge \neg q_i^-) \Big) \supset \bigwedge\nolimits_i \Big((q_i^+ \wedge \neg q_i^-) \supset (p_i^+ \wedge \neg p_i^-) \Big). \end{aligned}$$

Intuitively, a model $\mathcal{M}$ of $\mathsf{PRF}^{\pm}(\mathcal{AF})$ should satisfy two requirements: the condition in the first line of the formula (i.e., $\mathsf{CMP}^{\pm}(\mathcal{AF})$) assures that the pairs $\langle p^+, p^- \rangle$ that are verified by $\mathcal{M}$ correspond to a complete extension of $\mathcal{AF}$. The condition on the second and the third line ($\mathsf{CMP}^{\pm}(\mathcal{AF}) \supset \mathsf{INC}^{\pm}_{\subseteq}(\mathcal{AF})$) assures that this set of pairs is not strictly $\subset$-included in another set that forms a complete extension of $\mathcal{AF}$. We thus have:

[57] That is, for every valuation ν it holds that $\nu(\mathsf{T}) = t$ and $\nu(\mathsf{F}) = f$.

Proposition 109. ([13]) *Let $\mathcal{AF} = \langle \mathsf{Arg}, \mathit{Attack} \rangle$ be an argumentation framework. Then:*

- *For every preferred extension $\mathcal{E} \in \mathsf{Prf}(\mathcal{AF})$ there is a model $\mathcal{M}$ of $\mathsf{PRF}^{\pm}(\mathcal{AF})$ such that $\mathsf{In}(\mathcal{M}) = \mathcal{E}$, $\mathsf{Out}(\mathcal{M}) = \mathcal{E}^{+}$, and $\mathsf{Undec}(\mathcal{M}) = \mathsf{Arg} \setminus (\mathcal{E} \cup \mathcal{E}^{+})$.*
- *For every model $\mathcal{M}$ of $\mathsf{PRF}^{\pm}(\mathcal{AF})$ there is a preferred extension $\mathcal{E} \in \mathsf{Prf}(\mathcal{AF})$ such that $\mathcal{E} = \mathsf{In}(\mathcal{M})$, $\mathcal{E}^{+} = \mathsf{Out}(\mathcal{M})$, and $\mathsf{Arg} \setminus (\mathcal{E} \cup \mathcal{E}^{+}) = \mathsf{Undec}(\mathcal{M})$.*

In a similar way it is possible to represent the grounded semantics as well as other types of comparative Dung-type extensions, such as semi-stable semantics, eager semantic, ideal semantics, and so forth (see [13]). In [82] similar QBF-based representations are used for representing extensions of abstract dialectical frameworks [54], and in [9] they are used for representing conflict-tolerant semantics. It follows that off-the-shelf SAT-solvers and/or QBF-solvers may be used for computing argumentation-based entailments by Dung semantics.

Another approach based on propositional logic is taken in [169]. Again, arguments are represented by propositional letters in a finite set Atoms. The language of propositional logic is enriched with a connective $\twoheadrightarrow$ characterized by the axiom scheme $(\phi \wedge (\phi \twoheadrightarrow \psi)) \supset \neg\psi$ to express argumentative attack. The fact that an argument ψ (in Atoms) is defeated is then expressed by:

$$\mathsf{def}\psi =_{\mathsf{df}} \bigvee_{\phi \in \mathsf{Atoms}} (\phi \wedge (\phi \twoheadrightarrow \psi)).$$

In order to express admissible semantics, i.e., the idea that the selected arguments have to defend themselves from all attacks, the following axiom is used:

$$(\phi \wedge (\psi \twoheadrightarrow \phi)) \supset \mathsf{def}\psi.$$

The logic $\mathfrak{L}_{\mathsf{A}} = \langle \mathfrak{L}^{\twoheadrightarrow}_{\mathsf{Atoms}}, \vdash_{\mathsf{A}} \rangle$ is axiomatized by classical propositional logic enriched with the three discussed axiom schemes. In order to characterize complete extensions, $\mathfrak{L}_{\mathsf{A}}$ is enriched with

$$\bigwedge_{\phi \in \mathsf{Atoms}} ((\phi \twoheadrightarrow \psi) \supset \mathsf{def}\phi) \supset \psi$$

resulting in $\mathfrak{L}_{\mathsf{C}} = \langle \mathcal{L}^{\twoheadrightarrow}_{\mathsf{Atoms}}, \vdash_{\mathsf{C}} \rangle$, expressing that if an argument is defended then it is selected.[58]

Similar to the approach in QBL, in order to characterize grounded and preferred semantics, more formal machinery needs to be employed. Instead of quantifiers, in [169] the

[58] The presentation of the logics in [169] is slightly simplified in that the original systems also capture argumentative changes, that is, a dynamic proof theory is presented that allows for the addition of new arguments and new argumentative attacks "on-the-fly". For a similar approach see our discussion in Section 3.3.

preferential semantics of adaptive logics is used (recall Section 2.4.2-A). That means, for the grounded [preferred] semantics those $\mathfrak{L}_{\mathsf{C}}$-interpretations are selected in which the least [most] atoms are true. As shown in [173], the selection semantics underlying adaptive logics can also be expressed in terms of maximal consistent subsets.

Given our previous discussion of MCS-based reasoning, we therefore state the following corollary from [169, Theorem 1]: Given a logic $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ and sets $\mathcal{T}$ and $\mathcal{T}'$ of $\mathcal{L}$-sentences, let in the following proposition $\mathsf{MC}^{\mathcal{T}}_{\mathsf{L}}(\mathcal{T}')$ be the set of all maximally $\vdash$-consistent sets $\mathcal{S}$ of $\mathcal{L}$-sentences for which: (a) $\mathcal{T} \subseteq \mathcal{S}$, and (b) there is no $\vdash$-consistent set $\mathcal{S}'$ of $\mathcal{L}$-sentences for which both $(\mathcal{S} \cap \mathcal{T}') \subsetneq (\mathcal{S}' \cap \mathcal{T}')$ and $\mathcal{T} \subseteq \mathcal{S}'$.

Proposition 110. *Let $\mathcal{AF} = \langle \mathsf{Args}, \mathtt{Attack} \rangle$ be an abstract argumentation framework based on a finite set of arguments. Consider the language $\mathcal{L}^{\twoheadrightarrow}_{\mathsf{Args}}$ and let $\Gamma = \{\phi \twoheadrightarrow \psi \mid (\phi, \psi) \in \mathtt{Attacks}\} \cup \{\neg(\phi \twoheadrightarrow \psi) \mid (\phi, \psi) \notin \mathtt{Attacks}\}$. We have:*

- $\mathsf{Adm}(\mathcal{AF}) = \{\mathsf{Atoms}(\mathcal{S}) \mid \mathcal{S} \in \mathsf{MCS}^{\Gamma}_{\mathfrak{L}_{\mathsf{A}}}(\mathcal{L}^{\twoheadrightarrow}_{\mathsf{Args}})\}$
  (In other words, $\mathcal{T} \in \mathsf{Adm}(\mathcal{AF})$ iff there is a maximally $\mathfrak{L}_{\mathcal{A}}$-consistent set of sentences $\mathcal{S}$ for which $\Gamma \subseteq \mathcal{S}$ and $\mathcal{T} = \mathsf{Atoms}(\mathcal{S})$),
- $\mathsf{Cmp}(\mathcal{AF}) = \{\mathsf{Atoms}(\mathcal{S}) \mid \mathcal{S} \in \mathsf{MCS}^{\Gamma}_{\mathfrak{L}_{\mathsf{C}}}(\mathcal{L}^{\twoheadrightarrow}_{\mathsf{Args}})\}$,
- $\mathsf{Grd}(\mathcal{AF}) = \mathsf{Atoms}(\mathcal{S})$ *where* $\{\mathcal{S}\} = \mathsf{MC}^{\Gamma}_{\mathfrak{L}_{\mathcal{C}}}(\{\neg\phi \mid \phi \in \mathsf{Atoms}\})$,
- $\mathsf{Prf}(\mathcal{AF}) = \{\mathsf{Atoms}(\mathcal{S}) \mid \mathcal{S} \in \mathsf{MC}^{\Gamma}_{\mathfrak{L}_{\mathcal{A}}}(\mathsf{Atoms})\} = \{\mathsf{Atoms}(\mathcal{S}) \mid \mathcal{S} \in \mathsf{MC}^{\Gamma}_{\mathfrak{L}_{\mathcal{C}}}(\mathsf{Atoms})\}$,
- $\mathsf{SStb}(\mathcal{AF}) = \{\mathsf{Atoms}(\mathcal{S}) \mid \mathcal{S} \in \mathsf{MC}^{\Gamma}_{\mathfrak{L}_{\mathcal{C}}}(\{\phi \vee \mathsf{def}\,\phi \mid \phi \in \mathsf{Atoms}\})\}$.[59]

We note, finally, that the presentation in this section is by no means exhaustive, but rather meant to illustrate the way logical propositional formulas may be used for encoding the dynamics of argumentation-based reasoning. Among other approaches that are based on a Tarskian logic we recall the ones in [103] and [97] based on intuitionistic logic, in [92] based on Łukasiewicz logic, in [91] based on monadic second order logic, in [101] and [102] based on classical logic, and in [79] based on first-order logic with finite domains. We refer to [32] for a recent comprehensive survey on the subject (see in particular Sections 4–8 therein, which are relevant to the material in this chapter), where also a variety of implementations are described (summarized in [32, Table 4]).

[59] $\mathsf{SStb}(\mathcal{AF})$ is the set of the *semi-stable extensions* of $\mathcal{AF}$, that is: the complete extensions $\mathcal{E}$ such that $\mathcal{E} \cup \mathcal{E}^{+}$ is maximal among all the complete extensions of $\mathcal{AF}$.

3.2 Representation Methods Based on Modal Languages

In this section we consider several systems for reasoning about argumentation in a modal logical context. We distinguish two major purposes these systems serve:

1. The first goal, which is shared among all the presented systems and discussed in Section 3.2.1, is to express underlying notions of abstract argumentation, such as attacks and semantic selections, in the object language via modal operators.

2. The second goal, discussed in Section 3.2.2, is to integrate central notions underlying argumentative reasoning with those expressing argumentation dynamics in Item 1, for instance, propositional attitudes such as belief and endorsement, and justification. In this way, the presented logics offer a comprehensive logical model of (meta)argumentation and its dynamics.

We start with the basic settings of [44; 62; 113; 178; 114], which are concerned with meta-argumentative reasoning, and then move on to some frameworks that include epistemic considerations [115; 161].

3.2.1 Argumentation Logics

Grossi in [113; 114] defines *argumentation models* to reason about argumentative situations. An argumentation model $\mathcal{M}$ based on an argumentation framework $\mathcal{AF} = \langle \mathsf{Args}, \rightarrow \rangle$[60] is a tuple $\langle \mathsf{Args}, \leftarrow, v \rangle$, where $\leftarrow$ is the inverted version of $\rightarrow$ (that is, $A \leftarrow B$ iff $B \rightarrow A$). The pair $\langle \mathsf{Args}, \leftarrow \rangle$ constitutes a Kripkean possible world frame where arguments provide the points connected by the accessibility relation $\leftarrow$. As usual, the assignment v associates each propositional atom with a set of points (arguments) in which they hold.

In the following, we enrich the propositional language by two unary modalities. Thus, formulas in the language are defined by the following BNF:[61]

$$\phi \quad := \quad \mathsf{Atoms} \mid \neg\phi \mid \phi \wedge \phi \mid \boxbar_{\mathsf{a}}\phi \mid \boxbar_{\mathsf{u}}\phi \mid \mathsf{F}$$

where Atoms is a set of propositional atoms of the language. The diamond-versions of the given modal operators are defined as usual: $\diamondbar_{\mathsf{a}} =_{\mathsf{df}} \neg \boxbar_{\mathsf{a}} \neg$ and $\diamondbar_{\mathsf{u}} =_{\mathsf{df}} \neg \boxbar_{\mathsf{u}} \neg$. Other propositional connectives, such as implication $\supset$, disjunction $\vee$, and the propositional constant T for truth are defined as usual in classical propositional logic.

[60] To keep the original notations, we use in this section the arrow sign for designating the attack relation.

[61] We use the $\boxbar$-notation in our language since we will later on generalize this logic to a product logic where the argumentation-related modalities will provide the vertical axis.

Validity for atoms and propositional connectives is defined in the usual way. Similarly, the modal operators $\boxbar_{\mathsf{a}}$ and $\boxbar_{\mathsf{u}}$ function like a usual necessitation and universal necessitation operator. For a model $\mathcal{M} = \langle \mathsf{Args}, \leftarrow, v \rangle$ and an argument $A \in \mathsf{Args}$, we define:

- $\mathcal{M}, A \models \boxbar_{\mathsf{a}} \phi$ iff for all $B \in \mathsf{Args}$ for which $A \leftarrow B$ we have $\mathcal{M}, B \models \phi$. Since worlds are identified with arguments, this definition is understood as follows: all attackers B of the argument A have the property ϕ.
- $\mathcal{M}, A \models \boxbar_{\mathsf{u}} \phi$ iff for all $B \in \mathsf{Args}$, $\mathcal{M}, B \models \phi$. In words: all the arguments $B \in \mathsf{Args}$ have the property ϕ.[62]
- $\mathcal{M} \models \phi$ iff for all $A \in \mathsf{Args}$ it holds that $\mathcal{M}, A \models \phi$. The set of all formulas ϕ for which $\mathcal{M} \models \phi$ is denoted by $[\![\phi]\!]_{\mathcal{M}}$ (the subscript is removed when the context disambiguates).

In sum, since there are no frame conditions, we are dealing with models of the modal logic K enriched with universal modality.

Example 111. *Consider the argumentation framework and the assignment v presented in Figure 10.*

atom	$v(\cdot)$
p	$\{A, A'\}$
q	$\{A, C\}$

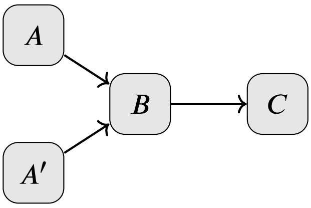


Figure 10: Left: the assignment of Example 111; Right: the argumentation framework of Example 111

In this case, we have:

- $\mathcal{M}, A \models \boxbar_{\mathsf{a}} \mathsf{F}$ *and* $\mathcal{M}, A' \models \boxbar_{\mathsf{a}} \mathsf{F}$, *expressing that A and A' have no attackers.*
- $\mathcal{M}, B \models \diamondbar_{\mathsf{a}} \boxbar_{\mathsf{a}} \mathsf{F}$, *expressing that there is an attacker against which B cannot be defended (since this attacker has no attackers).*
- $\mathcal{M}, C \models \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} \mathsf{T}$ *and* $\mathcal{M}, C \models \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} p$, *expressing that for all attackers of C there is a defender (either A or A')*

More generally, we have for any $x \in \mathsf{Args}$:

[62] Thus, if $\mathcal{M}, A_0 \models \boxbar_{\mathsf{u}} \phi$ for some A_0 then $\mathcal{M}, A \models \boxbar_{\mathsf{u}} \phi$ for every $A \in \mathsf{Args}$.

- $\mathcal{M}, x \models \boxbar_{\mathsf{u}}\big((p \vee q) \supset \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} (p \vee q)\big)$, *expressing that the set* $\{A, A', C\}$ *(consisting of the worlds in which* $p \vee q$ *holds) attacks all its attackers.*

As the following proposition shows, the induced logic is expressive enough to characterize several standard semantics.

Proposition 112. ([113, p. 411]) *Let* $\mathcal{AF} = \langle \mathsf{Args}, \to \rangle$ *and* $\mathcal{M} = \langle \mathsf{Args}, \leftarrow, v \rangle$. *For* $[\![\phi]\!]_{\mathcal{M}} = \mathcal{E} \subseteq \mathsf{Args}$, *it holds that:*

$$\mathcal{M} \models \mathsf{sem}(\phi) \textit{ iff } \mathcal{E} \in \mathsf{Sem}(\mathcal{AF}),$$

where the correspondence between the formula sem *and the semantics* Sem *is the following:*

Sem	sem(ϕ)
Adm	$\boxbar_{\mathsf{u}}(\phi \supset (\boxbar_{\mathsf{a}}\neg\phi \wedge \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} \phi))$
Cmp	$\boxbar_{\mathsf{u}}((\phi \supset \boxbar_{\mathsf{a}}\neg\phi) \wedge (\phi \leftrightarrow \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} \phi))$
Stb	$\boxbar_{\mathsf{u}}(\phi \leftrightarrow \boxbar_{\mathsf{a}}\neg\phi)$

Example 113. *In Example 111 we have, for instance, that:*

- $\mathcal{M} \models \mathsf{adm}(p)$, *since* $\{A, A'\}$ *is admissible, while*
- $\mathcal{M} \models \neg\mathsf{cmp}(p)$ *and* $\mathcal{M} \models \neg\mathsf{stb}(p)$, *since* $\{A, A'\}$ *is neither complete nor stable, and*
- $\mathcal{M} \models \mathsf{cmp}(p \vee q)$ *and* $\mathcal{M} \models \mathsf{stb}(p \vee q)$, *since* $\{A, A', C\}$ *is complete and stable.*

The logic, however, lacks the resources to express argumentation semantics that are based on minimality or maximality assumptions, such as grounded and preferred semantics. We recall (see [85]) that the grounded extension is characterized by the least fixed point of the function

$$\mathsf{defended} : \wp(\mathsf{Args}) \to \wp(\mathsf{Args}),$$

which maps a set $\mathcal{S}$ of arguments to the set of all arguments in Args that are defended by $\mathcal{S}$. Now, recall from our example that $\boxbar_{\mathsf{a}}\diamondbar_{\mathsf{a}}$ expresses argumentative defense in the logic, i.e., $\mathcal{M}, A \models \boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} \phi$ iff $[\![\phi]\!]_{\mathcal{M}}$ defends A. We thus need to characterize the formula ψ for which $[\![\psi]\!]_{\mathcal{M}}$ is minimal such that $[\![\psi]\!]_{\mathcal{M}} = [\![\boxbar_{\mathsf{a}} \diamondbar_{\mathsf{a}} \psi]\!]_{\mathcal{M}}$. For this purpose one can enrich the argumentation logic by a fixpoint μ-operator (see [51] for an introduction to modal μ-calculi), defined as follows: [63]

$$\mathcal{M}, A \models \mu p.\phi(p) \text{ iff } A \in \bigcap\big\{\mathcal{S} \in \wp(\mathsf{Args}) \mid [\![\phi]\!]_{\mathcal{M}[p:=\mathcal{S}]} \subseteq \mathcal{S}\big\},$$

[63] All systems introduced in this section have an adequate axiomatization (see e.g. [113]), which we omit for space reasons.

where $\mathcal{M}[p := \mathcal{S}] = \langle \mathsf{Args}, \leftarrow, v' \rangle$, $v'_{\mathsf{Atoms}\setminus\{p\}} = v_{\mathsf{Atoms}\setminus\{p\}}$, and $v'(p) = \mathcal{S}$.[64]

In [114] Grossi tackles preferred and semi-stable semantics[65] by means of a second-order formalization:

$$\mathcal{M}, A \models \exists p.\phi(p) \text{ iff there is an } \mathcal{S} \subseteq \mathsf{Args} \text{ such that } \mathcal{M}_{[p:=\mathcal{S}]}, A \models \phi(p).$$

The following proposition is shown in [113] for the grounded semantics and in [114] for the preferred and semi-stable semantics:[66]

Proposition 114. *Denote by* $\phi \sqsubseteq_{\mathsf{u}} \psi$ *the formula* $\boxbar_{\mathsf{u}}(\phi \supset \psi)$ *and denote by* $\phi \sqsubset_{\mathsf{u}} \psi$ *the formula* $(\phi \sqsubseteq_{\mathsf{u}} \psi) \wedge \neg(\psi \sqsubseteq_{\mathsf{u}} \phi)$*. Let* ϕ *be a formula such that* $[\![\phi]\!]_{\mathcal{M}} = \mathcal{E} \subseteq \mathsf{Args}$*. It holds that:*

$$\mathcal{M} \models \mathsf{sem}(\phi) \textit{ iff } \mathcal{E} \in \mathsf{Sem}(\mathcal{AF}),$$

where the correspondence between the formula sem *and the semantics* Sem *is the following:*

Sem	$\mathsf{sem}(\phi)$
Grd	$\mathsf{cmpl}(\phi) \wedge \forall q.(\mathsf{cmpl}(q) \supset \phi \sqsubseteq_{\mathsf{u}} q)$
Prf	$\mathsf{cmpl}(\phi) \wedge \neg\exists q.(\mathsf{cmpl}(q) \wedge \phi \sqsubset_{\mathsf{u}} q))$
SStb	$\mathsf{cmpl}(\phi) \wedge \neg\exists q.((\phi \vee \diamondbar_{\mathsf{a}}\phi) \sqsubset_{\mathsf{u}} (q \vee \diamondbar_{\mathsf{a}} q)))$

In [62], Caminada and Gabbay also use argumentation models, but proceed differently when characterizing argumentation semantics. Let p_i, p_o and p_u be three atoms which are intended to represent the three argument labels `in`, `out`, and `undec`. We can now elegantly express the characteristic requirements of complete labelings:[67]

1. $\mathcal{M}, A \models (\boxbar_{\mathsf{a}}\mathsf{F} \vee \boxbar_{\mathsf{a}} p_o) \supset p_i$ expresses that if A is not attacked ($\boxbar_{\mathsf{a}}\mathsf{F}$) or all attackers of A are `out` ($\boxbar_{\mathsf{a}} p_o$), then A is `in`;

2. $\mathcal{M}, A \models \diamondbar_{\mathsf{a}} p_i \supset p_o$ expresses that if A is attacked by an argument that is `in`, then A is `out`;

3. $\mathcal{M}, A \models \boxbar_{\mathsf{a}}(p_o \vee p_u) \wedge \diamondbar_{\mathsf{a}} p_u \supset p_u$ expresses that if A has only attackers that are `out` or `undec` and at least one attacker is `undec`, then A is `undec` as well;

[64] If $\mathcal{A}$ is a set of atoms and v is a valuation, $v_{\mathcal{A}}$ denotes the restriction of v to the atoms in $\mathcal{A}$.

[65] Recall Footnote 59.

[66] See below for the treatment of preferred extensions in [161] in terms of a fixpoint μ-operator.

[67] Recall Remark 106. See [57] and [23] for a characterization of argumentation semantics in terms of labelings.

4. $\mathcal{M}, A \models (p_i \vee p_o \vee p_u) \wedge \neg(p_i \wedge p_o) \wedge \neg(p_i \wedge p_u) \wedge \neg(p_o \wedge p_u)$ expresses that A has exactly one label.

By restricting argumentation models to those that satisfy Items 1–4 (at every argument A), we can, for instance, characterize the grounded extension as follows, where again $\mathcal{AF} = \langle \mathsf{Args}, \rightarrow \rangle$: If for every model $\mathcal{M}$ in the restricted class based on the frame $\langle \mathsf{Args}, \leftarrow \rangle$ we have $\mathcal{M}, B \models p_i$ then $B \in \mathsf{Grd}(\mathcal{AF})$, and vice versa. Other semantics are represented in [62] by techniques from circumscription logic.

A different approach is taken in [44] and [178]. The starting point is again an argumentation framework $\mathcal{AF} = \langle \mathsf{Args}, \rightarrow \rangle$, but instead of treating arguments as possible worlds in a Kripkean frame as in the previous approaches, the set of worlds is now given by $\wp(\mathsf{Args})$. Again, the accessibility relation encodes argumentative attacks.

Denote by $\rightarrow^{\wp}$ the following lifting of $\rightarrow$ to $\wp(\mathsf{Args}) \times \wp(\mathsf{Args})$: we write $\mathcal{S} \rightarrow^{\wp} \mathcal{S}'$ iff there is an $A \in \mathcal{S}$ and a $B \in \mathcal{S}'$ such that $A \rightarrow B$. Let also $\rightarrow^{\wp}_{\mathsf{C}} = (\wp(\mathsf{Args}) \times \wp(\mathsf{Args})) \setminus \rightarrow^{\wp}$ be the complement of $\rightarrow^{\wp}$. Figure 11 shows a simple example.

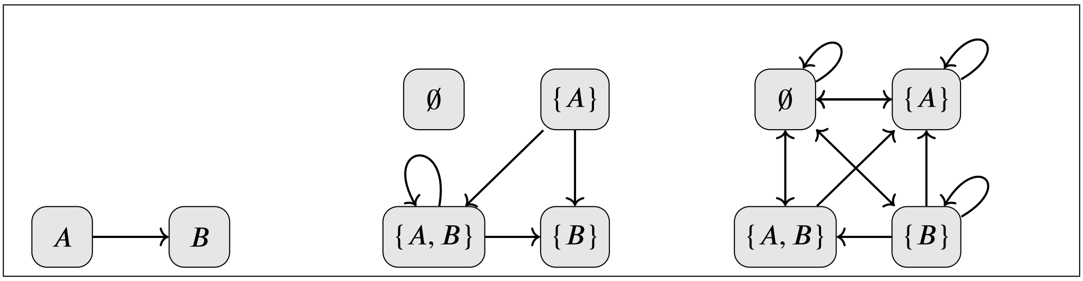


Figure 11: Left: The attack diagram for $\mathcal{AF} = \langle \{A, B\}, \rightarrow \rangle$, where $\rightarrow = \{(a, b)\}$; Middle: Graph for $\rightarrow^{\wp}$; Right: Graph for $\rightarrow^{\wp}_{\mathsf{C}}$.

The formal language is similar to the ones given above, except that now the propositional atoms corresponds directly to the abstract arguments:

$$\phi \quad := \quad \mathsf{Args} \mid \neg\phi \mid \phi \wedge \phi \mid \Box_{\mathsf{u}}\phi \mid \Box_{\mathsf{a}}\phi$$

The truth conditions of propositional connectives are as usual. We define:

- $\mathcal{M}, \mathcal{S} \models A$ iff $A \in \mathcal{S}$. This expresses that a is a member of the currently considered set of arguments;
- $\mathcal{M}, \mathcal{S} \models \Box_{\mathsf{a}}\phi$ iff for all $\mathcal{S}'$ for which $\mathcal{S} \rightarrow^{\wp}_{\mathsf{C}} \mathcal{S}'$, it holds that $\mathcal{M}, \mathcal{S}' \models \phi$. This expresses that ϕ holds for all sets of arguments $\mathcal{S}'$ not attacked by $\mathcal{S}$.

- $\mathcal{M}, \mathcal{S} \models \Box_{\mathsf{u}}\phi$ iff for all $\mathcal{S} \in \wp(\mathsf{Args})$ it holds that $\mathcal{M}, \mathcal{S} \models \phi$. This expresses that all sets of arguments have the property ϕ.

Just like the previous formalisms, at its core also this logic is K enriched with a universal modality. The logic allows us to express core concepts of abstract argumentation such as attack and defense:

- $\mathcal{M} \models \Box_{\mathsf{u}}(A \supset \Box_{\mathsf{a}}\neg B)$ expresses that A attacks B,
- $\mathcal{M} \models \Box_{\mathsf{u}}(\bigwedge\mathcal{S} \supset \Box_{\mathsf{a}}\neg\bigwedge\mathcal{S}')$ expresses that some argument $A \in \mathcal{S}$ attacks some argument $A' \in \mathcal{S}'$,
- $\mathcal{M} \models \Box_{\mathsf{u}} \bigwedge_{\mathcal{S}' \in \wp(\mathsf{Args})} \big(\Box_{\mathsf{u}}(\bigwedge \mathcal{S}' \supset \Box_{\mathsf{a}}\neg A) \supset \Box_{\mathsf{u}}(\bigwedge \mathcal{S} \supset \Box_{\mathsf{a}}\neg \bigwedge\mathcal{S}')\big)$ expresses that the set of arguments $\mathcal{S}$ defends the argument A.[68]

In a series of articles Gabbay and various co-authors investigate logical characterizations of argumentation frameworks. In [102] and [103] the basic idea is similar to the systems presented above: arguments are represented by propositional atoms, and the fact that an argument A attacks argument B is represented by the formula $A \supset {\sim}B$, in which $\supset$ is an implication and $\sim$ is a negation of the underlying logic. Different core logics are considered:

- In [103] the underlying logic is the intuitionistic logic G_3, whose Kripkean models consist of two linearly ordered worlds (also known as Here-and-There logic [149]).
- In [102] the underlying logic is classical and $\sim$ is a strong negation N, for which ${\sim}p \supset \neg p$ but not necessarily vice versa (where $\neg$ is the classical negation).[69] N can be used to express different argument label/statuses: a holds if a is `in`, $\mathsf{N}a$ holds if a is `out` and $\neg a \wedge \neg\mathsf{N}a$ holds if a is `undec`.

Remark 115. *The negation* N *in the second item also has an elegant modal characterization in the logic* CNN *[102]. Like* G_3*, there are two worlds in the underlying pointed Kripkean*

[68] To express this, the set Args is supposed to be finite (otherwise a second-order approach is needed). In order to express properties of specific semantics the authors enhance their modal logic by unary non-normal modal operators. We refer to [178] for further details.

[69] An earlier characterization of Dung-style argumentation in classical logic has been presented in [101] for stable semantics (as well as for complete semantics in a 3-valued setting). The only logical connective in the presented system is the "Peirce-Quine-Dung dagger" $\Downarrow$, a generalization of the Peirce-Quine dagger or of NOR: $\Downarrow \Delta$ is true iff $\bigvee\Delta$ is false. The attack relation corresponds in this representation to the direct subformula relation (which is generalized to equivalence classes in order to deal with attack cycles): note that if $\Downarrow\Delta$ is true all members of Δ are false and, vice versa, if some member of Δ is true, $\Downarrow\Delta$ is false. In this context Gabbay also develops a "geometric concept of proof" which concerns inference rules (such as geometrical modus ponens) that operate on patterns of a given attack diagram and which are adequate to a given proof procedure in the Peirce-Quine-Dung-Dagger logic. Similar to the modal systems discussed here, the logic in [101] offers several generalizations, such as quantifiers, higher-order attacks, etc.

models, just now for each world the other world is the only accessible one. The modal truth conditions for N *are then spelled out by:* $\mathsf{N}\phi$ *holds in one world iff* $\neg\phi$ *holds in the other. Similarly to intuistionistic possible worlds models (including those of* G_3*), models of* CNN *are constrained by a "monotony" requirement on* $\models$*: if p holds at the actual world, it necessarily holds at the other world as well. However, if p holds at the non-actual world, it need not hold at the actual world, although the actual world is accessible.*

The translations of a given argumentation framework into the language of G_3 (see Equation (1)) or of CNN (see Equation (2)) are also similar for both systems, where for each $x \in \mathsf{Args}$, $x^- = \{y \in \mathsf{Args} \mid y \to x\}$ and the formula n in Equation (1), introduced to identify the actual world, can be defined by $\bigwedge_{x\in\mathsf{Args}}(x \vee \neg x)$:[70]

$$\bigwedge_{x\in\mathsf{Args}}\left(\overbrace{\big(x\supset(\mathsf{n}\vee\bigwedge_{y\in x^-}\neg y)\big)}^{\text{if }\texttt{in}\text{, all attackers }\texttt{out}}\wedge\overbrace{\big(\bigwedge_{y\in x^-}\neg y\supset(\mathsf{n}\vee x)\big)}^{\text{if all attackers }\texttt{out}\text{, then }\texttt{in}}\wedge\overbrace{\big(\neg x\supset(\mathsf{n}\vee\bigvee_{y\in x^-}y)\big)}^{\text{if }\texttt{out}\text{, some attackers }\texttt{in}}\wedge\overbrace{\big(\bigvee_{y\in x^-}y\supset(\mathsf{n}\vee\neg x)\big)}^{\text{if some attackers }\texttt{in}\text{, then }\texttt{out}}\right) \tag{1}$$

$$\bigwedge_{x\in\mathsf{Args}}\left(\overbrace{\big(\bigwedge_{y\in x^-}\mathsf{N}y\leftrightarrow x\big)}^{x\ \texttt{in}\text{ iff all attackers }\texttt{out}}\wedge\overbrace{\big(\big(\bigwedge_{y\in x^-}\neg y\wedge\bigvee_{y\in x^-}\neg\mathsf{N}y\big)\supset(\neg x\wedge\neg\mathsf{N}x)\big)}^{\text{if all attackers not }\texttt{in}\text{ and some }\texttt{und}\text{, then }\texttt{und}}\wedge\overbrace{\big(\bigwedge_{x\in y^-}x\supset\mathsf{N}y\big)}^{x\text{ attacks }y}\right) \tag{2}$$

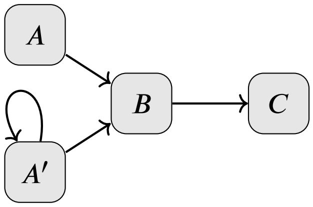
In both systems (i.e., the Kripkean semantics for G_3 and in CNN), we can, for each atom, identify one of the truth-assignment patterns in (the left part of) Table 16 relative to the two worlds in a given model. These patterns correspond to argument labels as indicated in the same table. This means that the models of the translated argumentation frameworks are one-to-one related to the complete labelings of the framework. As a consequence, the entailed atoms characterize the grounded extension. Stable semantics can be characterized by demanding excluded middle $p \vee {\sim}p$ (where again in the case of G_3 $\sim$ is intuitionistic negation and in the case of CNN it is strong negation).

We illustrate this by means of the argumentation framework in Figure 12.

[70] Clearly, like previous encodings, the translations presuppose a finite set of arguments.

	G$_3$ / CNN			LN1			
	in	out	undec	in	out	undec	
w_1	1	0	0	1	0	1	w_1
w_2	1	0	1	1	0	0	w_2
				1	0	1	w_3

Table 16: Overview: truth-value assignment pattern and argument labelings. Note that in G$_3$ and CNN two worlds are used, while in LN1 there are three worlds.



	G$_3$ / CNN		LN1			
	w_1	w_2	w_1	w_2	w_3	label
C	1	1	1	1	1	in
B	0	0	0	0	0	out
A	1	1	1	1	1	in
A'	0	1	1	0	1	undec

Figure 12: Example for the characterizations of the given AF on the left in the logics G$_3$, CNN and LN1

A related approach is introduced in [100] and [62], where argumentation frameworks are characterize in terms of provability logic[71] and argumentation labelings are modeled in terms of fixed points of modal formulas. The underlying logic LN1 is given by K4, enhanced with:

- LÃűb's axiom ($\Diamond\phi \supset \Diamond(\phi \wedge \Box\neg\phi)$),
- an axiom of linearity ($(\Diamond\phi \wedge \Diamond\psi) \supset (\Diamond(\phi \wedge \psi) \vee \Diamond(\phi \wedge \Diamond\psi) \vee \Diamond(\psi \wedge \Diamond\phi))$), and
- some axioms characterizing the behavior of atoms: $(p \supset \Box(\neg p \supset \Box p)$, $\Box(\Box\bot \vee p) \leftrightarrow \Box p$ and $\Box(\Box\bot \vee \neg p) \leftrightarrow \Box\neg p)$.

Pointed LN1 models are such that the accessibility relation $<$ forms finite linear chains starting with the actual world. Additionally, it is required that if all non-endpoints of $<$ agree on the assignment of an atom, then the endpoint takes over the same assignment.

[71] A similar approach was used in [99] for cyclic logic programs.

Let $G\phi = \phi \wedge \Box\phi$. Argumentation frameworks are translated into the language of $\mathsf{LF1}$ as follows:

$$G\left(\Box\bot \vee \bigwedge_{\substack{x\in\mathsf{Args}\\ x^-\neq\emptyset}}\left(x \leftrightarrow \bigwedge_{y\in x^-}\Diamond\neg y\right)\right) \wedge \bigwedge_{\substack{x\in\mathsf{Args}\\ x^-=\emptyset}} Gx \tag{3}$$

In [100] it is shown that there is a one-to-one correspondence between $\mathsf{LP1}$-models of the formula in Equation (3), whose states form chains of length 3, and complete labelings of the given argumentation framework. As was the case for G_3 and CNN, we can again uniquely associate argument labels with valuation patterns at the given possible worlds (see the right-hand side of Table 16). We show how this plays out in our example in Figure 12.

Remark 116. *The logics* $\mathsf{G}_3, \mathsf{CNN}$ *and* $\mathsf{LN1}$ *can readily express higher-order and joint attacks, as well as argument quantifiers. We refer to the original papers for more details.*

3.2.2 Belief, Informativeness and Awareness

One of the advantages of using modal argumentation logics is the possibility to integrate epistemic modalities. In this section we demonstrate this.

Grossi and van der Hoek [115] propose a modal product logic (see [105]) in which the argumentation logic from [113; 114] (see our discussion in the previous section) provides one ingredient and a $\mathsf{KD45}$ epistemic logic provides another. The latter have frames of the form $\langle \mathcal{S}, \mathcal{P}\rangle$, where $\mathcal{S}$ is a set of (epistemic) states and $\mathcal{P} \subseteq \mathcal{S}$ is a non-empty subset of $\mathcal{S}$, namely those that a given agent considers possible. A frame of the product logic is then the product of an epistemic frame $\langle \mathcal{S}, \mathcal{P}\rangle$ and an argumentation frame $\langle \mathcal{A}, \leftarrow\rangle$. The domain of a model $\mathcal{M}$ of the product logic is the Cartesian product between epistemic states and arguments ($\mathcal{S} \times \mathsf{Args}$) and its assignment function v associates propositional atoms with sets of state-argument pairs in its domain. One can picture the workings of such a product logic in terms of a chess-board with epistemic states providing the x-axis and arguments providing the y-axis (see Example 117 below for a concrete illustration). The epistemic modality, $\boxminus_{\mathsf{b}}$, and its universal cousin, $\boxminus_{\mathsf{u}}$, move along the x-axis while keeping arguments fixed. The argumentative modality $\boxbar_{\mathsf{a}}$ and $\boxbar_{\mathsf{u}}$, move along the y-axis while keeping states fixed:

- $\mathcal{M}, (s, A) \models \boxbar_{\mathsf{a}}\phi$ iff for all $B \in \mathsf{Args}$ such that $A \leftarrow B$, we have: $\mathcal{M}, (s, B) \models \phi$
- $\mathcal{M}, (s, A) \models \boxbar_{\mathsf{u}}\phi$ iff for all $B \in \mathsf{Args}$, we have: $\mathcal{M}, (s, B) \models \phi$
- $\mathcal{M}, (s, A) \models \boxminus_{\mathsf{b}}\phi$ iff for all $s' \in \mathcal{P}$, we have: $\mathcal{M}, (s', A) \models \phi$.

- $\mathcal{M}, (s, A) \models \boxminus_{\mathsf{u}} \phi$ iff for all $s' \in \mathcal{S}$, we have: $\mathcal{M}, (s', A) \models \phi$.

Grossi and van der Hoek also introduce a designated symbol/atom σ to signify that an argument A supports an epistemic state s in case $\mathcal{M}, (s, A) \models \sigma$.

To illustrate these definitions, we take a look at an example.

Example 117. *Consider the following argumentative scenario (inspired by [143] and [113]):*

Default (C) *It was sunny yesterday, so it will be sunny today.*

Pete (B) *Currently there are thick clouds, it is going to rain and storm.*

CNN (A) *The weather report of the CNN reports sunny but windy weather.*

FOX (A') *The weather report of FOX news reports sunny and calm weather.*

We use the atoms w for it "being windy", s for it "being sunny", and `CNN`, `FOX`, *and* `Pete` *are atoms that indicate sources of information.*

We consider the epistemic states $\mathcal{S} = \{s_1, s_2, s_3\}$ where the possible epistemic states of our agent are $\mathcal{P} = \{s_1, s_2\}$. Figure 13 illustrates the situation. On the y-axis we find our four arguments where the arrows between them illustrate the inverted(!) attack relation. On the x-axis we find the epistemic state, where the possible epistemic states in $\mathcal{P}$ are highlighted.

- *Highlighted in boxes along the x axis are properties of arguments that are robust under changes of the epistemic state. For instance,*
  - $\mathcal{M}, (s_i, A) \models \mathtt{CNN}$ *for all* $1 \leq i \leq 3$, *which indicates that argument A is based on evidence from CNN.*
  - *Similarly, argument A' is based on evidence from FOX, etc.*
- *Highlighted in boxes along the y-axis are properties of epistemic states that are robust under changes of the considered argument. For instance,*
  - $\mathcal{M}, (s_1, x) \models s \land \neg w$ *for all* $x \in \{A, A', B, C\}$, *which expresses that according to state s_1 we have calm and sunny weather.*
- *The symbol σ indicates which arguments support which epistemic states. For instance,*
  - $\mathcal{M}, (s_2, A') \models \sigma$ *meaning that argument A' supports state s_2.*

In the given system we can express properties that concern information states that involve both beliefs and argumentative properties, such as:

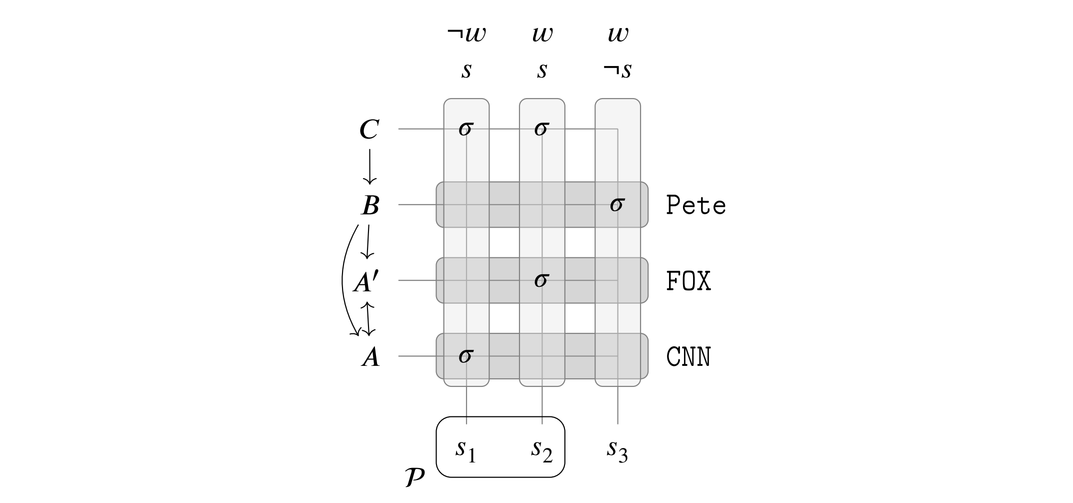


Figure 13: Model $\mathcal{M}$ in for Example 117. The vertical [horizontal] boxes represent properties of states [arguments] that are robust under changes of the considered arguments [states].

- $\mathcal{M} \models (\neg s \wedge \sigma) \supset \boxbar_{\mathsf{a}}(\texttt{CNN} \wedge \texttt{FOX})$ *meaning that if an argument supports "not sunny" then all attackers of it rely on CNN or FOX.*

- $\mathcal{M} \models \boxminus_{\mathsf{b}}(s \wedge ((w \wedge \sigma) \supset (\texttt{FOX} \vee \Diamond\mkern-10mu\mid_{\mathsf{a}}\texttt{Pete})))$ *meaning that our agent believes s and that if an argument supports windy weather then it relies on FOX or it is attacked by an argument that relies on Pete.*

Grossi and van der Hoek enrich this framework further by an *endorsement operator* $\boxbar_{\mathsf{e}}$ that works similar to $\boxminus_{\mathsf{b}}$ except that it operates on the y-axis and therefore concerns arguments rather than epistemic states: instead of fixing a set of possible belief states we now fix a set of endorsed arguments $\mathcal{E} \subseteq \mathsf{Args}$ and define:

- $\mathcal{M}, (s, A) \models \boxbar_{\mathsf{e}}\phi$ iff for all $a \in \mathcal{E}$, $\mathcal{M}, (s, a) \models \phi$.

This way it is possible to formally characterize several types of argumentation-based beliefs:

- $\mathsf{SB}\phi = \boxminus_{\mathsf{b}}(\boxbar_{\mathsf{u}}\phi \wedge \Diamond\mkern-10mu\mid_{\mathsf{u}}\sigma)$ expressing an (argumentatively) supported belief in ϕ,

- $\mathsf{EB}\phi = \boxminus_{\mathsf{b}}(\boxbar_{\mathsf{u}}\phi \wedge \Diamond\mkern-10mu\mid_{\mathsf{e}}\sigma)$ expressing an endorsed supported belief in ϕ, and

- $\mathsf{JB}(\phi, \psi) = \boxminus_{\mathsf{b}}(\boxbar_{\mathsf{u}}\phi \wedge \Diamond\mkern-10mu\mid_{\mathsf{e}}(\sigma \wedge \boxminus_{\mathsf{u}}\psi))$ expressing a belief in ϕ, justified by a belief in ψ.[72]

[72] In this definition also a universal belief modality is used, which is defined as usual.

Example 118. *Suppose that in Example 117 we have six agents, Anne, Bill, Chris, Dan, Eli, and Fay that endorse different arguments and have different beliefs. We have, for instance:*

	Anne	*Bill*	*Chris*	*Dan*	*Eli*	*Fay*
Endorsed arguments	$\{A', C\}$	$\{C\}$	$\{A\}$	$\{B\}$	$\{A', C\}$	$\{B\}$
Possible belief states	$\{s_2\}$	$\{s_1, s_2\}$	$\{s_1\}$	$\{s_1, s_2\}$	$\{s_3\}$	$\{s_3\}$
$\mathsf{SB}s$	*Yes*	*Yes*	*Yes*	*Yes*	*No*	*No*
$\mathsf{EB}s$	*Yes*	*Yes*	*Yes*	*No*	*No*	*No*
$\mathsf{JB}(s, \mathtt{FOX})$	*Yes*	*No*	*No*	*No*	*No*	*No*
$\mathsf{JB}(s, \mathtt{CNN})$	*No*	*No*	*Yes*	*No*	*No*	*No*
$\mathsf{JB}(\neg s, \mathtt{Pete})$	*No*	*No*	*No*	*No*	*No*	*Yes*

While in the framework of Grossi and van der Hoek belief and argumentative considerations are treated by independent modalities, in [161] beliefs are dependent on the underlying argumentative structure. For this they consider *argumentation-support models* which are defined as product modal logics similar to the models discussed above. Let us highlight some differences. First, the language in [161] does not allow for arbitrary nesting of modalities. The underlying grammar is defined as follows:

$$\alpha := \top \mid p \mid \neg\alpha \mid \alpha \wedge \alpha \mid \boxminus_{\mathsf{u}}\alpha \mid \boxbar_{\mathsf{u}}\beta \qquad \beta := \top \mid \Box_{\mathsf{a}}\alpha \mid \neg\beta \mid \beta \wedge \beta \mid \boxbar^{\alpha}_{\mathsf{a}}\beta \mid \mathsf{Gfp}^{\alpha}$$

While α-formulas express facts about possible worlds, β-formulas describe arguments. To explain the meaning of the different modal operators, let us take a look at the semantics.

For this we take a closer look at the argumentation-support models introduced. An argumentation-support model is given by a tuple $\langle \mathcal{S}, \mathsf{Args}, \{\leftarrow_X \mid X \subseteq \mathcal{S}\}, v_{\mathsf{s}}, v_{\mathsf{a}}\rangle$, where $\mathcal{S}$ is a (non-empty) set of (factual) states, Args is a set of arguments, for each $X \subseteq \mathcal{S}$, $\leftarrow_X$ is a contextualized (inverted) attack relation, and v_{s} [respectively, v_{a}] associates propositional atoms [respectively, arguments] with [non-empty] sets of states.[73] [74] Just like in [115], formulas are evaluated at state-argument pairs. For all classical connectives this works as expected (e.g., $M, (s, A) \models p$ iff $s \in v_s(p)$, and, $M, (s, A) \models \phi_1 \wedge \phi_2$ iff $M, (s, A) \models \phi_1$ and $M, (s, A) \models \phi_2$, etc.). Let us therefore take a look at the modal operators.

First, we notice that the attack modality $\boxbar^{\alpha}_{\mathsf{a}}$ is contextualized to formulas α expressing claims that are disputed in the respective attacks.

[73] Note the difference of this approach to the models of [115], in which there is only one assignment function $v : \mathsf{Atoms} \to \wp(\mathcal{S} \times \mathsf{Args})$.

[74] In [160] and in a similar setting the same authors propose a topological semantics to model evidence supporting arguments.

- $\mathcal{M}, (s, A) \models \boxbar_{\mathsf{a}}^{\phi}\psi$ iff for all B for which $A \leftarrow_{[\![\phi]\!]_{\mathcal{M}}} B$, it holds that $\mathcal{M}, (s, B) \models \psi$ (where $[\![\phi]\!]_{\mathcal{M}} = \{s' \in \mathcal{S} \mid M, (s, C) \models \phi$ for any $C \in \mathsf{Args}\}$). In words: all attackers B of the argument A in a dispute about the claim ϕ satisfy ψ (where, just like in the product logics of [115] discussed above, we keep the given state fixed).

The authors consider several constraints on this relation:

1. $A \leftarrow_X B$ iff $A \leftarrow_{W \setminus X} B$. Clearly, if the attack concerns the question whether X is the case, it will equally concern the question whether $W \setminus X$ is the case.
2. If $A \leftarrow_X B$ then
   (a) $v_{\mathsf{a}}(A) \subseteq X$ or $v_{\mathsf{a}}(A) \subseteq W \setminus X$, and
   (b) $v_{\mathsf{a}}(A) \subseteq X$ implies $v_{\mathsf{a}}(B) \subseteq W \setminus X$.

   The attacked argument will either support X or $W \setminus X$ and the attacking argument should have an opposite stance.
3. If $A \leftarrow_X B$ and $v_{\mathsf{a}}(A) \subseteq Y \subset X$, then $A \leftarrow_Y B$. If B attacks A concerning the claim X and A supports the stronger claim Y, then B also attacks A on the stronger claim.

The universal vertical and horizontal modalities $\boxbar_{\mathsf{u}}$ and $\boxminus_{\mathsf{u}}$ are analogous to those in [115] discussed above. For the $\Box_{\mathsf{a}}$ modality we have:

- $\mathcal{M}, (s, A) \models \Box_{\mathsf{a}}\alpha$ iff $v_{\mathsf{a}}(A) \subseteq [\![\alpha]\!]_{\mathcal{M}}$, meaning that the considered argument A supports the claim α.

Also, Shi et al. enhance the logic with a μ-operator Gfp^{α} (similar to [113], see the discussion in the previous section) to express membership in admissible extensions:[75]

- $\mathcal{M}, (s, A) \models \mathsf{Gfp}^{\phi}$ iff A is in an admissible set of arguments in the argumentation framework $\langle \mathsf{Args}, \rightarrow_{[\![\phi]\!]_{\mathcal{M}}} \rangle$.

An agent believes in α in case there is an admissible argument for α and there is no admissible argument for $\neg\alpha$. This can be expressed by putting

$$\mathsf{B}\alpha \;:=\; \overbrace{\diamondbar_{\mathsf{u}}(\underbrace{\Box_{\mathsf{a}}\alpha}_{\text{it supports } \alpha} \wedge \underbrace{\mathsf{Gfp}^{\alpha}}_{\text{it is admissible}})}^{\text{there is an argument s.t. \dots}} \wedge \overbrace{\neg\diamondbar_{\mathsf{u}}(\underbrace{\Box_{\mathsf{a}}\neg\alpha}_{\text{it supports } \neg\alpha} \wedge \underbrace{\mathsf{Gfp}^{\neg\alpha}}_{\text{it is admissible}})}^{\text{there is no argument s.t. \dots}}.$$

Example 119. *Consider again the scenario in Example 117. Given a set of states* $\mathcal{S} = \{s_1, s_2, s_3\}$ *we let our assignments be as in Table 17.*

We then get, for instance, where $1 \leq i \leq 3$,

[75] Gfp^{α} is the greatest postfix point of $\boxbar_{\mathsf{a}}^{\alpha}\diamondbar_{\mathsf{a}}^{\alpha}$. See [161] for an axiomatization. Note also that the discussion in [161] is restricted to uncontroversial argumentation frameworks (see also [85] for a definition).

atom	$v_{\mathsf{s}}(\cdot)$
s	$\{s_1, s_2\}$
w	$\{s_2, s_3\}$

arg.	$v_{\mathsf{a}}(\cdot)$
A	$\{s_1\}$
A'	$\{s_2\}$
B	$\{s_3\}$
C	$\{s_1, s_2\}$

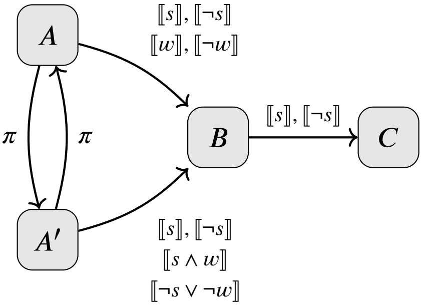


Table 17: Left and Middle: Assignments for Example 119; Right: The attack-diagrams for the contextualized attack relations. Arrows exist for each of the listed labels (e.g., $B \rightarrow_{[\![s]\!]} C$ and $B \rightarrow_{[\![\neg s]\!]} C$), where π is a placeholder for $[\![s \wedge w]\!]$, $[\![\neg s \vee \neg w]\!]$, $[\![w]\!]$ and $[\![\neg w]\!]$.

- $\mathcal{M}, (s_i, x) \models \mathsf{Gfp}^s \wedge \Box_{\mathsf{a}} s$ *for* $x \in \{A, A', C\}$*, while* $\mathcal{M}, (s_i, B) \not\models \mathsf{Gfp}^s$ *and* $\mathcal{M}, (s_i, B) \not\models \Box_{\mathsf{a}} s$
- $\mathcal{M}, (s_i, A') \models \mathsf{Gfp}^{s \wedge w} \wedge \Box_{\mathsf{a}}(s \wedge w)$ *and* $\mathcal{M}, (s_i, A) \models \mathsf{Gfp}^{\neg(s \wedge w)} \wedge \Box_{\mathsf{a}}\neg(s \wedge w)$
- $\mathcal{M} \models \mathsf{B}s$ *while* $\mathcal{M} \not\models \mathsf{B}(s \wedge w)$*.*

The systems presented above have the merit of allowing for argumentation-based approaches to belief and justification, which allow for new and interesting insights. E.g., for all of Grossi's and van der Hoek's belief types (SB, EB and JB) negative introspection fails for beliefs that are not supported by arguments, but succeeds otherwise. That is (where $\mathsf{XB} \in \{\mathsf{SB}, \mathsf{EB}\}$), while:

$\not\models \neg\mathsf{XB}\phi \supset \mathsf{XB}\neg\mathsf{XB}\phi$, and

$\not\models \neg\mathsf{JB}(\phi, \psi) \supset \mathsf{JB}(\neg\mathsf{JB}(\phi, \psi), \psi)$

we have (see [115, Proposition 6])

$\models (\neg\mathsf{XB}\phi \wedge \boxminus_{\mathsf{b}} \diamondminus_{\mathsf{e}} \sigma) \supset \mathsf{XB}\neg\mathsf{XB}\phi$, and

$\models (\neg\mathsf{JB}(\phi, \psi) \wedge \boxminus_{\mathsf{b}} \diamondminus_{\mathsf{e}} (\sigma \wedge \boxminus_{\mathsf{u}}\psi)) \supset \mathsf{JB}(\neg\mathsf{JB}(\phi, \psi), \psi)$

Similarly, in Shi et al.'s system the aggregation of beliefs fails, i.e., $\not\models (\mathsf{B}\alpha \wedge \mathsf{B}\alpha') \supset \mathsf{B}(\alpha \wedge \alpha')$, which may give rise to applications to paradoxes, respectively difficult scenarios, such as the lottery or the preface paradox.

3.3 Reasoning with Dynamic Derivations

Although the satisfiability methods described in the previous sections are logic-based, from a pure logical perspective they have some drawbacks:

- In many of the described formalisms, the encoding of the arguments are by propositional variables, thus arguments are treated as abstract entities. As such, these methods are more adequate to abstract argumentation [23] than to structured argumentation. Put differently, if these methods are applied to argumentation frameworks such as the ones considered in Section 2, the construction of the frameworks and the reasoning methods are distinguished: first the arguments and the attacks among them are produced, and only then the satisfiability-based methods can be applied on them.
- Even more serious is the fact that many of these methods are applicable only to *finite* argumentation frameworks, as for the encoding of the formulas a finite set of arguments is assumed. As such, these methods are suitable only for some logical instantiations (assumption-based frameworks, for instance), but not for all of them (e.g., logic-based argumentation frameworks which are infinite since so are the transitive closures of sets of assertions).

In this section we describe an alternative method to reasoning with logic-based argumentation, which overcomes the two shortcomings of the other approach described above: it is applicable to infinite frameworks and is affected by the logical content of the arguments and the attack rules.

Let $\mathcal{AF}_{\mathfrak{L},\mathcal{A}}(\mathcal{S}) = \langle \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}), Attack(\mathcal{A})\rangle$ be a logical argumentation framework (Definition 8) and let $\mathcal{P}$ be a sound and complete proof system for $\mathfrak{L}$.[76] The idea is to use (inference) rules in $\mathcal{P}$ for deriving new arguments from already derived ones, and to use (attack) rules in $\mathcal{A}$ for excluding derived arguments, when opposing arguments are also derived. This gives rise to the notion of *dynamic proofs* (or dynamic derivations), which are intended for explicating the actual non-monotonic flavor of reasoning processes in a logical argumentation framework. The main idea behind these formalisms is that, unlike 'standard' proof methods, an argument can be challenged (and possibly withdrawn) by a counter-argument, and so a certain argument may be considered as not accepted at a certain stage of the proof, even if it were considered accepted in an earlier stage of the proof. It is only when an argument is

[76] $\mathcal{P}$ may be a Hilbert-type proof system, a Gentzen-type sequent calculus, a natural deduction system, a semantic tableaux system, or any other proof method that is based on finite sequences (or trees) of finite syntactical expressions which are based on the underlying language (see e.g. Section 1.3 of [20] for a general definition of such proof systems). Here we concentrate on sequent calculi, since a sequent is in fact a multiple-conclusion argument. For the other kinds of proof systems some simple modifications of the definitions in what follows are needed.

'finally derived' (in the sense that will be explained later on) that it can be safely concluded by the dynamic proof. In the rest of this section we elaborate on this idea (full details and formal definitions can be found in [16]).

A proof system in our case is determined by a *proof setting* $\mathfrak{S} = \langle \mathfrak{L}, \mathcal{P}, \mathcal{A} \rangle$ consisting of a logic $\mathfrak{L}$, a corresponding sound and complete proof calculus $\mathcal{P}$ for producing $\mathfrak{L}$-arguments, and a set $\mathcal{A}$ of attack rules for eliminating (undefended) attacked arguments. An argument $\langle \mathcal{S}, \psi \rangle$ that is eliminated (i.e., is attacked by an application of a rule in $\mathcal{A}$) will be denoted in what follows by ~~$\langle \mathcal{S}, \psi \rangle$~~.

Definition 120 (proof tuple)**.** A (proof) *tuple* is a triple $T = \langle i, A, \mathsf{J} \rangle$, where i (the tuple's index) is a natural number, $A \in \{\langle \Gamma, \Delta \rangle, \cancel{\langle \Gamma, \Delta \rangle}\}$ (the tuple's argument) is a (possibly attacked) multiple-conclusion argument,[77, 78] and J (the tuple's justification) is a string, consisting of a rule name followed by a sequence of numbers.[79] In the sequel we shall sometimes identify a proof tuple with its argument.

Definition 121 (simple derivation)**.** Let $\mathfrak{S} = \langle \mathfrak{L}, \mathcal{P}, \mathcal{A} \rangle$ be a proof setting. A *simple $\mathfrak{S}$-derivation* based on a set $\mathcal{S}$ of formulas in $\mathcal{L}$, is a finite sequence $\mathcal{D}_{\mathfrak{S}}(\mathcal{S}) = \langle T_1, \dots T_m \rangle$ of proof tuples, where each $T_i \in \mathcal{D}$ is of either of the following forms:

- $T_i = \langle i, A, \mathsf{J} \rangle$, where J ="$\mathcal{R}$ $i_1, \dots, i_n$" and A is obtained by applying the inference rule $\mathcal{R} \in \mathcal{P}$ on the arguments of the tuples $T_{i_1}, \dots T_{i_k}$ $(i_1, \dots, i_n < i)$.
- $T_i = \langle i, A, \mathsf{J} \rangle$, where J ="$\mathcal{R}$ $i_1, \dots, i_n$" and A is obtained by applying the elimination rule $\mathcal{R} \in \mathcal{A}$ on the arguments of the tuples $T_{i_1}, \dots T_{i_k}$ $(i_1, \dots, i_n < i)$. In this case both the attacked argument A and the attacking argument A_{i_1} should be elements of $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$.[80]

Tuples of the first form are called *introducing tuples* and those of the second form are called *eliminating tuples*.

Example 122. Let $\mathcal{P}$ be Gentzen's proof system LK for classical logic. Table 18 presents this system in terms of (multiple-conclusion) arguments.

Consider now the set of assumptions $\mathcal{S} = \{\neg p, p, q\}$ (see also Example 37). Figure 14 presents a simple derivation with respect to LK and Ucut as the sole attack rule. To simplify the reading, in this and other derivations in the rest of the paper we shall sometimes use abbreviations or omit some details, e.g. the tuple signs in proof steps.

[77] Thus Δ, the conclusion of A, is a finite set of formulas and not just a formula. (In classical logic, Δ may be replaced by its disjunction $\bigvee \Delta$.) When Δ is a singleton we shall omit the parentheses and identify A with a standard argument in the sense of Definition 5.

[78] When the underlying calculus is Hilbert-type or based on a natural deduction system, A may be just a formula (corresponding to the rule conclusions is those proof systems) rather than an argument.

[79] This string indicates what rule has to be applied, and on what tuples, in order to derive T.

[80] This prevents situations in which, e.g., $\langle \neg p, \neg p \rangle$ Ucut-attacks $\langle p, p \rangle$, although $\mathcal{S} = \{p\}$.

Rule Name	Acronym	Rule's conditions	Rule's conclusion
Axiom			$\langle\psi, \psi\rangle$
Weakening		$\langle\mathcal{S}, \mathcal{T}\rangle$	$\langle\mathcal{S} \cup \mathcal{S}', \mathcal{T} \cup \mathcal{T}'\rangle$
Cut		$\langle\mathcal{S}_1, \mathcal{T}_1 \cup \{\psi\}\rangle$, $\langle\mathcal{S}_2 \cup \{\psi\}, \mathcal{T}_2\rangle$	$\langle\mathcal{S}_1 \cup \mathcal{S}_2, \mathcal{T}_1 \cup \mathcal{T}_2\rangle$
Left-$\wedge$	[$\wedge$L]	$\langle\mathcal{S} \cup \{\psi\} \cup \{\phi\}, \mathcal{T}\rangle$	$\langle\mathcal{S} \cup \{\psi \wedge \phi\}, \mathcal{T}\rangle$
Right-$\wedge$	[$\wedge$R]	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi\}\rangle$, $\langle\mathcal{S}, \mathcal{T} \cup \{\phi\}\rangle$	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi \wedge \phi\}\rangle$
Left-$\vee$	[$\vee$L]	$\langle\mathcal{S} \cup \{\psi\}, \mathcal{T}\rangle$, $\langle\mathcal{S} \cup \{\phi\}, \mathcal{T}\rangle$	$\langle\mathcal{S} \cup \{\psi \vee \phi\}, \mathcal{T}\rangle$
Right-$\vee$	[$\vee$R]	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi\} \cup \{\phi\}\rangle$	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi \vee \phi\}\rangle$
Left-$\supset$	[$\supset$L]	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi\}\rangle$, $\langle\mathcal{S} \cup \{\phi\}, \mathcal{T}\rangle$	$\langle\mathcal{S} \cup \{\psi \supset \phi\}, \mathcal{T}\rangle$
Right-$\supset$	[$\supset$R]	$\langle\mathcal{S} \cup \{\psi\}, \mathcal{T} \cup \{\phi\}\rangle$	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi \supset \phi\}\rangle$
Left-$\neg$	[$\neg$L]	$\langle\mathcal{S}, \mathcal{T} \cup \{\psi\}\rangle$	$\langle\mathcal{S} \cup \{\neg\psi\}, \mathcal{T}\rangle$
Right-$\neg$	[$\neg$R]	$\langle\mathcal{S} \cup \{\psi\}, \mathcal{T}\rangle$	$\langle\mathcal{S}, \mathcal{T} \cup \{\neg\psi\}\rangle$

Table 18: Arguments construction rules according to LK.

Note that in this derivation Tuple 8 represents a Ucut-attack of the argument in Tuple 7 on the argument in Tuple 1 (where the former serves also as the justification of the attack), and Tuple 11 represents a Ucut-attack of the argument in Tuple 1 on the argument in Tuple 7, justified by the arguments in Tuples 9 and 10. Thus, Tuples 8 and 11 are eliminating while the other tuples are introducing.

Not all the attacks in a simple derivation should be successful, since if the attacking argument is itself attacked by another argument (i.e., it appears in an eliminating tuple) the attack may not be validated. The iterative process in Figure 15 checks this, and evaluates each tuple's argument: Elim is the status of an eliminated argument whose attacker is not already eliminated, Attack means that the argument attacks an argument whose status is Elim, and Accept is the status of a derived argument whose status is not Elim.

Definition 123 ((strongly) coherent derivation)**.** *A simple derivation $\mathcal{D}$ is* coherent*, if there is no argument that eliminates another argument and that is eliminated itself. Formally:*

1.	$\langle p, p \rangle$	Axiom
2.	$\langle \emptyset, \{p, \neg p\} \rangle$	Right-$\neg$, 1
3.	$\langle \emptyset, p \vee \neg p \rangle$	Right-$\vee$, 2
4.	$\langle p \vee \neg p, \neg(p \wedge \neg p) \rangle$	$\cdots$
5.	$\langle \neg(p \wedge \neg p), p \vee \neg p \rangle$	$\cdots$
6.	$\langle q, q \rangle$	Axiom
7.	$\langle \neg p, \neg p \rangle$	Axiom
8.	~~$\langle p, p \rangle$~~	Ucut, 7, 7, 7, 1
9.	$\langle p, \neg\neg p \rangle$	$\cdots$
10.	$\langle \neg\neg p, p \rangle$	$\cdots$
11.	~~$\langle \neg p, \neg p \rangle$~~	Ucut, 1, 9, 10, 7

Figure 14: A derivation with respect to LK and Ucut, based on $\mathcal{S} = \{\neg p, p, q\}$

```
Input: a simple derivation $\mathcal{D}$.
let $\mathsf{Attack} := \mathsf{Elim} := \mathsf{Derived} := \emptyset$;
while ($\mathcal{D}$ is not empty) do {
    if the last element in $\mathcal{D}$ introduces an argument $A$, then
        add $A$ to the set $\mathsf{Derived}$;
    if the last element in $\mathcal{D}$ is an attack of $A_1 \notin \mathsf{Elim}$ on $A_2$, then
        add $A_1$ to $\mathsf{Attack}$ and $A_2$ to $\mathsf{Elim}$;
    remove the last element from $\mathcal{D}$ }
let $\mathsf{Accept} := \mathsf{Derived} - \mathsf{Elim}$;
Output: $\mathsf{Attack}$, $\mathsf{Elim}$, $\mathsf{Accept}$.
```

Figure 15: Evaluation of a simple derivation.

$\mathsf{Attack}(\mathcal{D}) \cap \mathsf{Elim}(\mathcal{D}) = \emptyset$. *We say that $\mathcal{D}$ is* strongly coherent*, if*

$$\mathsf{Sup}(\mathsf{Attack}(\mathcal{D})) = \bigcup_{A \in \mathsf{Attack}(\mathcal{D})} \mathsf{Sup}(A)$$

is consistent.[81]

Example 124 (Example 122 continued)**.** *Consider the simple derivation $\mathcal{D}$ of Example 122.*

- *When considering only the simple derivation consisting of lines 1–8 we have that* $\langle q, q\rangle, \langle\neg p, \neg p\rangle \in \mathsf{Accept}$, $\mathsf{Attack} = \{\langle\neg p, \neg p\rangle\}$ *and* $\mathsf{Elim} = \{\langle p, p\rangle\}$.
- *When considering the simple derivation consisting of lines 1–11 we have that* $\langle q, q\rangle$, $\langle p, p\rangle \in \mathsf{Accept}$, $\mathsf{Attack} = \{\langle p, p\rangle\}$ *and* $\mathsf{Elim} = \{\langle\neg p, \neg p\rangle\}$. *Note that when the algorithm in Figure 15 reaches line 8,* $\langle p, p\rangle$ *is not added to* Elim *since its attacking argument* $\langle\neg p, \neg p\rangle$ *is already in* Elim *at that point.*[82]

In particular, in each step the derivation that is obtained is both coherent and strongly coherent.

Now we can define what dynamic derivations are.

Definition 125 (dynamic derivation)**.** Let $\mathfrak{S} = \langle \mathfrak{L}, \mathcal{P}, \mathcal{A}\rangle$ be a proof setting. A *dynamic $\mathfrak{S}$-derivation* based on a set $\mathcal{S}$ of formulas in $\mathcal{L}$, is an $\mathcal{S}$-based simple $\mathfrak{S}$-derivation $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$ which is of one of the following forms:

a) $\mathcal{D}_{\mathfrak{S}}(\mathcal{S}) = \langle T\rangle$, where $T = \langle 1, A, \mathsf{J}\rangle$ is a proof tuple.

b) $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$ is obtained by adding to a dynamic derivation a sequence of introducing tuples whose arguments are not in $\mathsf{Elim}(\mathcal{D}_{\mathfrak{S}}(\mathcal{S}))$.

c) $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$ is obtained by adding to a dynamic derivation a sequence of eliminating tuples where the attacking arguments are in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$ and are not attacked by arguments in $\mathsf{Accept}(\mathcal{D}_{\mathfrak{S}}(\mathcal{S})) \cap \mathsf{Arg}_{\mathfrak{L}}(\mathcal{S})$. The attacks must be based on arguments that are proved in $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$.[83]

One may think of a dynamic derivation as a proof that progresses over derivation steps. At each step the current derivation is extended by a 'block' of introduced arguments or eliminated arguments. As a result, the statuses of the arguments in the derivation are updated. In particular, a derived argument may be eliminated in light of new derived arguments, but also the other way around is possible: an eliminated argument may be 'restored' if its attacking argument is counter-attacked. It follows that previously accepted data may not be accepted anymore (and vice-versa) until and unless new derived information revises the state of affairs.

[81] As shown in [11], in the proof setting $\mathfrak{S} = \langle \mathsf{CL}, LK, \{\mathsf{Ucut}\}\rangle$, strong coherence implies coherence (but not vice-versa).

[82] This is so, since the evaluation process progresses backwards, from the last tuple to the first one, so $\langle\neg p, \neg p\rangle$ is already eliminated in the first evaluation step, following line 11.

[83] This condition assures that the attacks are 'sound': the attacking arguments are not counter-attacked by an accepted $\mathcal{S}$-based argument.

Example 126 (Examples 122 and 124, continued)**.** *The simple derivation of Example 122 is also a dynamic derivation. Example 124 demonstrates the dynamic nature of this derivation. For instance, although the argument $\langle \neg p, \neg p\rangle$ is derived in Step 7 of the derivation, it is eliminated in Step 11 of the derivation as a consequence of an Undercut attack, initiated by $\langle p, p\rangle$.*

The next definition, of the outcomes of a dynamic derivation, indicates when it is 'safe' to conclude that a derived argument must hold under any circumstances.

Definition 127 (final derivability)**.** *Let $\mathfrak{S} = \langle \mathfrak{L}, \mathcal{P}, \mathcal{A}\rangle$ be a proof setting and let $\mathcal{S}$ be a set of $\mathcal{L}$-formulas.*

- *A formula ψ is* finally derived *in a coherent dynamic $\mathfrak{S}$-derivation $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$, if for some $\Gamma \subseteq \mathcal{S}$ the argument $A = \langle \Gamma, \psi\rangle$ is in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}) \cap \mathsf{Accept}(\mathcal{D}_{\mathfrak{S}}(\mathcal{S}))$, and for every coherent dynamic derivation $\mathcal{D}'_{\mathfrak{S}}(\mathcal{S})$ extending $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$ (i.e., any dynamic derivation whose prefix is $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$), still $A \in \mathsf{Accept}(\mathcal{D}'_{\mathfrak{S}}(\mathcal{S}))$.*

- *A formula ψ is* sparsely finally derived *in a strongly coherent dynamic $\mathfrak{S}$-derivation $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$, if for some $\Gamma \subseteq \mathcal{S}$ the argument $A = \langle \Gamma, \psi\rangle$ is in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}) \cap \mathsf{Accept}(\mathcal{D}_{\mathfrak{S}}(\mathcal{S}))$, and for every strongly coherent dynamic derivation $\mathcal{D}'_{\mathfrak{S}}(\mathcal{S})$ that extends $\mathcal{D}_{\mathfrak{S}}(\mathcal{S})$ there is some $\Gamma' \subseteq \mathcal{S}$ such that the argument $A' = \langle \Gamma', \psi\rangle$ is in $\mathsf{Arg}_{\mathfrak{L}}(\mathcal{S}) \cap \mathsf{Accept}(\mathcal{D}'_{\mathfrak{S}}(\mathcal{S}))$.*

Thus, final derivability means that an argument is derived and accepted in a valid dynamic derivation and remains in this status in every extension of the derivation. Sparse final derivability is a weaker notion, meaning that if an argument A is derived and accepted in a valid dynamic derivation, in every extension of that derivation the conclusion of A is a conclusion of a derived and accepted argument.

Definition 128 ($\mid\!\sim^{\mathfrak{S}}_{\cap}$, $\mid\!\sim^{\mathfrak{S}}_{\Cap}$)**.** *Let $\mathfrak{S} = \langle \mathfrak{L}, \mathfrak{C}, \mathcal{A}\rangle$ be a proof setting, $\mathcal{S}$ a set of $\mathcal{L}$-formulas, and ψ an $\mathcal{L}$-formula.*

- *$\mathcal{S} \mid\!\sim^{\mathfrak{S}}_{\cap} \psi$ iff there is a $\mathfrak{S}$-derivation based on $\mathcal{S}$, in which ψ is finally derived.*

- *$\mathcal{S} \mid\!\sim^{\mathfrak{S}}_{\Cap} \psi$ iff there is a $\mathfrak{S}$-derivation based on $\mathcal{S}$, in which ψ is sparsely finally derived.*

Example 129.

a) q is finally derived (and so also sparsely finally derived) in the derivation of Figure 14 where $\mathfrak{S} = \langle \mathsf{CL}, LK, \{Ucut\}\rangle$ and $\mathcal{S} = \{p, \neg p, q\}$. Indeed, the only arguments in $\mathsf{Arg}_{\mathsf{CL}}(\mathcal{S})$ that can potentially Ucut-attack $\langle q, q\rangle$ are of the form $\langle \{p, \neg p\}, \psi\rangle$ or $\langle \{p, \neg p, q\}, \psi\rangle$, where ψ is logically equivalent to $\neg q$. However, such arguments are counter-attacked by the argument $\langle \emptyset, p \vee \neg p\rangle$, obtained in Tuple 3 of the derivation. It *follows, by the conditions in Item (c) of Definition 125, that no eliminating tuple in which $\langle q, q\rangle$ is attacked can be derived in any extension of the derivation above, thus q is finally derived in this derivation.*

We have, then, that $\{p, \neg p, q\} \mathrel{|\!\sim}^{\mathfrak{S}}_{\star} q$, while $\{p, \neg p, q\} \not\mathrel{|\!\sim}^{\mathfrak{S}}_{\star} p$ and $\{p, \neg p, q\} \not\mathrel{|\!\sim}^{\mathfrak{S}}_{\star} \neg p$, for any $\star \in \{\cap, \Cap\}$.

b) *To see the need for sparse final derivability, let again $\mathfrak{S} = \langle \mathsf{CL}, LK, \{Ucut\}\rangle$ and consider the set $\mathcal{S}' = \{p \wedge q, \neg p \wedge q\}$. Note that both $A_1 = \langle p \wedge q, q\rangle$ and $A_2 = \langle \neg p \wedge q, q\rangle$ are LK-derivable in this case, but neither of them is finally derivable, since any $\mathfrak{S}$-derivation that includes them can be extended with derivations of $A_3 = \langle \neg p \wedge q, \neg(p \wedge q)\rangle$ and $A_4 = \langle p \wedge q, \neg(\neg p \wedge q)\rangle$ that respectively Ucut-attack A_1 and A_2. Note, however, that these attacks cannot be applied* simultaneously*, since the attackers A_3 and A_4 counter-attack each other. It follows that in each extension of the derivation either A_1 or A_2 is accepted, and so q is sparsely finally derived from $\mathcal{S}'$.*

We have, then, that $\{p\wedge q, \neg p\wedge q\} \mathrel{|\!\sim}^{\mathfrak{S}}_{\Cap} q$ (and it is easy to verify that $\{p\wedge q, \neg p\wedge q\} \not\mathrel{|\!\sim}^{\mathfrak{S}}_{\Cap} p$ and $\{p \wedge q, \neg p \wedge q\} \not\mathrel{|\!\sim}^{\mathfrak{S}}_{\Cap} \neg p$).

The next proposition, introduced in [11], provides some soundness and completeness results for entailments by dynamic proofs (Definition 128) and entailments induced by Dung-semantics (Definition 12), and relates both of these entailments to reasoning with maximal consistency (Definition 44).

Proposition 130. Let $\mathfrak{S} = \langle \mathsf{CL}, LK, \{\text{Ucut}\}\rangle$ be a proof setting. Then for every finite set $\mathcal{S}$ of formulas and formula ψ, it holds that:

- $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{S}}_{\cap} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL}}_{\cap\mathsf{mcs}} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL},\{\mathsf{Ucut}\}}_{\mathsf{Grd}} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL},\{\mathsf{Ucut}\}}_{\cap\mathsf{Prf}} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL},\{\mathsf{Ucut}\}}_{\cap\mathsf{Stb}} \psi$.
- $\mathcal{S} \mathrel{|\!\sim}^{\mathfrak{S}}_{\Cap} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL}}_{\Cap\mathsf{mcs}} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL},\{\mathsf{Ucut}\}}_{\Cap\mathsf{Prf}} \psi$ iff $\mathcal{S} \mathrel{|\!\sim}^{\mathsf{CL},\{\mathsf{Ucut}\}}_{\Cap\mathsf{Stb}} \psi$.

We refer to [11] for further related results, where e.g. the base logic is not necessarily classical logic and the attack is not necessarily Undercut.

Example 131. *The first item of Example 129 demonstrates the first two items of the last proposition for $\mathcal{S} = \{p, \neg p, q\}$ (Examples 122 and 126), as $\bigcap \mathsf{MCS}_{\mathsf{CL}}(\mathcal{S}) = \{q\}$. The second item of Example 129 exemplifies the second item of Proposition 130, where $\mathcal{S}' = \{p \wedge q, \neg p \wedge q\}$ is the set of assertions.*

Some other approaches for reasoning with logic-based (structured) argumentation frameworks are the following:[84]

[84] As indicated before, description of algorithms for reasoning with argumentation frameworks which are not logic-based, including those for abstract argumentation frameworks, are not in the scope of the current chapter. For the latter, see e.g. the surveys in [144] and [68].

- For logic-based methods whose arguments are classical (Definition 4), already the construction of arguments poses serious computational challenges, since the finding of a *minimal* subset of a set of formulas that implies the consequent is in the second level of the polynomial hierarchy [96]. Algorithms for constructing classical arguments and counter-arguments can be found e.g. in [93].

- Common computational machineries of logic-based argumentation frameworks are based on dispute trees and dispute derivations [86; 88], both of which can be represented as games between proponent and opponent players. For some illustrations and an overview of their use in ABA frameworks, see [87, Section 5] and [73, Section 5].

- Illustrations of reasoning with ASPIC$^+$ can be found, e.g., in [146, Section 4.5]; Inference engines for APSIC$^+$ are surveyed (with relevant further references) in [147, Section 6].

In [169] a similar dynamic proof theory to the one discussed above has been presented, but for abstract argumentation instead of structured argumentation. It allows for the addition of new arguments and new argumentative attacks in an ongoing open-ended proof of an adaptive logic. The finally derivable propositional atoms are those that are in the intersection of a given semantics. The latter are characterized in terms of different adaptive proof strategies.

4 Concluding Remarks

Formal argumentation theory is by now a well-established and still extensively growing research area, even when restricted to its applications in Artificial Intelligence. There is no wonder, then, that it has many branches with different disciplines, some of them went as far as pure graph-theoretical approaches, treating argumentation frameworks as directed graphs, and so viewing their nodes (that is, the arguments) as totally abstract entities. In this chapter, we have taken to some extent the opposite approach, arguing that a meaningful and solid argumentation-based system must have a *logic* behind it, which takes a primary role not only in the construction of argumentation frameworks, but is also essential for the specification of their dynamics and deductive methods of reasoning. In Sections 2 and 3 we demonstrated, respectively, the fundamental role that logic may (and should) have in relation to these two aspects of formal argumentation systems. Indeed, the common ground of all the approaches surveyed in this chapter is that they are logically developed methodologies towards formal argumentation systems. We believe that this is crucial for justifying the outcomes of such systems in a logical and rational way.

Acknowledgments

We thank Marcos Cramer and the reviewers of this chapter for their helpful comments. Ofer Arieli is supported by the Israel Science Foundation (Grant No. 550/19). Jesse Heyninck is supported by the German National Science Foundation under the DFG-project CAR (Conditional Argumentative Reasoning) KE-1413/11-1.

References

[1] Gianvincenzo Alfano, Sergio Greco, Francesco Parisi, Gerardo I. Simari, and Guillermo R. Simari. On the incremental computation of semantics in dynamic argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 11. College Publications, 2021.

[2] Leila Amgoud. Postulates for logic-based argumentation systems. *International Journal of Approximate Reasoning*, 55(9):2028–2048, 2014.

[3] Leila Amgoud and Philippe Besnard. A formal analysis of logic-based argumentation systems. In Amol Deshpande and Anthony Hunter, editors, *Proceedings of the 4th International Conference Scalable Uncertainty Management (SUM'10)*, volume 6379 of *Lecture Notes in Computer Science*, pages 42–55. Springer, 2010.

[4] Leila Amgoud and Philippe Besnard. Logical limits of abstract argumentation frameworks. *Journal of Applied Non-Classical Logics*, 23(3):229–267, 2013.

[5] Leila Amgoud, Philippe Besnard, and Srdjan Vesic. Identifying the core of logic-based argumentation systems. In *Proceedings of the IEEE 23rd International Conference on Tools with Artificial Intelligence (ICTAI'11)*, pages 633–636. IEEE Computer Society, 2011.

[6] G. Aldo Antonelli. A directly cautious theory of defeasible consequence for default logic via the notion of general extension. *Artificial Intelligence*, 109(1–2):71–109, 1999.

[7] Grigoris Antoniou. Studying properties of classes of default logics. *Journal of Experimental & Theoretical Artificial Intelligence*, 10(4):495–505, 1998.

[8] Krzysztof R. Apt. Logic programming. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*, pages 493–574. Elsevier and MIT Press, 1990.

[9] Ofer Arieli. On the acceptance of loops in argumentation frameworks. *Journal of Logics and Computation*, 26(4):1203–1234, 2016.

[10] Ofer Arieli, AnneMarie Borg, and Jesse Heyninck. A review of the relations between logical argumentation and reasoning with maximal consistency. *Annals of Mathematics and Artificial Intelligence*, 87(3):187–226, 2019.

[11] Ofer Arieli, AnneMarie Borg, and Christian Straßer. Reasoning with maximal consistency by argumentative approaches. *Journal of Logic and Computation*, 28(7):1523–1563, 2018.

[12] Ofer Arieli, AnneMarie Borg, and Christian Straßer. Tuning logical argumentation frameworks: A postulate-derived approach. In Roman Barták and Eric Bell, editors, *Proceed-*

ings of the 33rd International Florida Artificial Intelligence Research Society Conference (FLAIRS'20), pages 557–562. AAAI Press, 2020.

[13] Ofer Arieli and Martin Caminada. A QBF-based formalization of abstract argumentation semantics. *Journal of Applied Logic*, 11(2):229–252, 2013.

[14] Ofer Arieli and Christian Straßer. Sequent-based logical argumentation. *Argument & Computation*, 6(1):73–99, 2015.

[15] Ofer Arieli and Christian Straßer. Deductive argumentation by enhanced sequent calculi and dynamic derivations. In Mario Benevides and René Thiemann, editors, *Proceedings of the 10th Workshop on Logical and Semantic Frameworks with Applications, (LSFA'15)*, volume 323 of *Electronic Notes in Theoretical Computer Science*, pages 21–37, 2016.

[16] Ofer Arieli and Christian Straßer. Logical argumentation by dynamic proof systems. *Theoretical Computer Science*, 781:63–91, 2019.

[17] Ofer Arieli and Christian Straßer. On minimality and consistency tolerance in logical argumentation frameworks. In Henry Prakken, Stefano Bistarelli, Francesco Santini, and Carlo Taticchi, editors, *Proceedings of the 8th International Conference on Computational Models of Argument (COMMA'20)*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 91–102. IOS Press, 2020.

[18] Arnon Avron. What is relevance logic? *Annals of Pure and Applied Logic*, 165(1):26–48, 2014.

[19] Arnon Avron. RM and its nice properties. In Katalin Bimbó, editor, *J. Michael Dunn on Information Based Logics*, volume 8 of *Outstanding Contributions to Logic*, pages 15–43. Springer, 2016.

[20] Arnon Avron, Ofer Arieli, and Anna Zamansky. *Theory of Effective Propositional Paraconsistent Logics*, volume 75 of *Studies in Logic*. College Publications, 2018.

[21] Chitta Baral, Sarit Kraus, and Jack Minker. Combining multiple knowledge bases. *IEEE Transactions on Knowledge and Data Engineering*, 3(2):208–220, 1991.

[22] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26(4):365–410, 2011.

[23] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. Abstract argumentation frameworks and their semantics. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 159–236. College Publications, 2018.

[24] Pietro Baroni and Massimiliano Giacomin. Semantics of abstract argument systems. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 25–44. Springer, 2009.

[25] Diderik Batens. On a logic of induction. *L&PS – Logic & Philosophy of Science*, IV(1):3–32, 2006.

[26] Diderik Batens. A universal logic approach to adaptive logics. *Logica Universalis*, 1(1):221–242, 2007.

[27] Diderik Batens and Lieven Haesaert. On classical adaptive logics of induction. *Logique et Analyse*, 44(173–175):255–290, 2003.

[28] Ringo Baumann, Sylvie Doutre, Jean-Guy Mailly, and Johannes P. Wallner. Enforcement in formal argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 8. College Publications, 2021.

[29] Mathieu Beirlaen and Christian Straßer. Non-monotonic reasoning with normative conflicts in multi-agent deontic logic. *Journal of Logic and Computation*, 24(6):1179–1207, 2014.

[30] Salem Benferhat, Claudette Cayrol, Didier Dubois, Jérôme Lang, and Henri Prade. Inconsistency management and prioritized syntax-based entailment. In Ruzena Bajcsy, editor, *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI'93)*, pages 640–647. Morgan Kaufmann, 1993.

[31] Salem Benferhat, Didier Dubois, and Henri Prade. Some syntactic approaches to the handling of inconsistent knowledge bases: A comparative study part 1: The flat case. *Studia Logica*, 58(1):17–45, 1997.

[32] Philippe Besnard, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. Logical theories and abstract argumentation: A survey of existing works. *Journal of Argument and Computation*, 11(1-2):41–102, 2020.

[33] Philippe Besnard and Sylvie Doutre. Checking the acceptability of a set of arguments. In James P. Delgrande and Torsten Schaub, editors, *Proceedings of the 10th International Workshop on Non-Monotonic Reasoning (NMR'04)*, pages 59–64, 2004.

[34] Philippe Besnard, Alejandro García, Antony Hunter, Sanjay Modgil, Henry Prakken, Guillermo Simari, and Francesca Toni. Introduction to structured argumentation. *Argument & Computation*, 5(1):1–4, 2014.

[35] Philippe Besnard and Anthony Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128(1–2):203–235, 2001.

[36] Philippe Besnard and Anthony Hunter. Argumentation based on classical logic. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 133–152. Springer, 2009.

[37] Philippe Besnard and Anthony Hunter. Constructing argument graphs with deductive arguments: a tutorial. *Argument & Computation*, 5(1):5–30, 2014.

[38] Philippe Besnard and Anthony Hunter. A review of argumentation based on deductive arguments. In Pietro Baroni, Dov Gabay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 437–484. College Publications, 2018.

[39] Philippe Besnard and Torsten Schaub. Signed systems for paraconsistent reasoning. *Journal of Automated Reasoning*, 20:191–213, 1998.

[40] Patrick Blackburn, Johan van Benthem, and Frank Wolter. *Handbook of Modal Logic*. Studies in Logic and Practical Reasoning. Elsevier Science, 2006.

[41] Alexander Bochman. *Explanatory Nonmonotonic Reasoning*, volume 4 of *Advances in Logic*. World Scientific, 2005.

[42] Alexander Bochman. Two paradigms of nonmonotonic reasoning. In *Proceedings of the International Symposium on Artificial Intelligence and Mathematics (ISAIM'06)*, 2006.

[43] Alexander Bochman. *A logical theory of nonmonotonic inference and belief change*. Springer

Science & Business Media, 2013.

[44] Guido Boella, Joris Hulstijn, and Leendert Van Der Torre. A logic of abstract argumentation. In Simon Parsons, Nicolas Maudet, Pavlos Moraitis, and Iyad Rahwan, editors, *Proceedings of the 2nd International Workshop on Argumentation in Multi-Agent Systems (ArgMAS'05)*, volume 4049 of *Lecture Notes in Computer Science*, pages 29–41. Springer, 2005.

[45] Bart Bogaerts. *Groundedness in logics with a fixpoint semantics*. PhD thesis, Informatics Section, Department of Computer Science, Katholieke Universiteit Leuven, 2015.

[46] Andrei Bondarenko, Phan Minh Dung, Robert Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93(1):63–101, 1997.

[47] AnneMarie Borg. *Modeling Defeasible Reasoning from an Argumentative Angle, With Emphasis on Sequent-Based Argumentation*. PhD thesis, Ruhr Universität Bochum, 2019.

[48] AnneMarie Borg. Assumptive sequent-based argumentation. *Journal of Applied Logics-IfCoLog Journal of Logics and their Applications*, 7(3):227–294, 2020.

[49] AnneMarie Borg and Christian Straßer. Relevance in structured argumentation. In Jérôme Lang, editor, *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI'18)*, pages 1753–1759. ijcai.org, 2018.

[50] AnneMarie Borg, Christian Straßer, and Ofer Arieli. A generalized proof-theoretic approach to structured argumentation by hypersequent calculi. *Studia Logica*, 109(1):167–238, 2021.

[51] Julian C. Bradfield and Colin Stirling. Modal logics and mu-calculi: An introduction. In Jan A. Bergstra, Alban Ponse, and Scott A. Smolka, editors, *Handbook of Process Algebra*, pages 293–330. North-Holland / Elsevier, 2001.

[52] Gerhard Brewka. Preferred subtheories: An extended logical framework for default reasoning. In Natesa Sridharan, editor, *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI'89)*, pages 1043–1048. Morgan Kaufmann, 1989.

[53] Gerhard Brewka. Adding priorities and specificity to default logic. In Craig MacNish, David Pearce, and Luís Moniz Pereira, editors, *Proceedings of the European Workshop on Logics in Artificial Intelligence, European Workshop, (JELIA'94)*, volume 838 of *Lecture Notes in Computer Science*, pages 247–260. Springer, 1994.

[54] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes Peter Wallner, and Stefan Woltran. Abstract dialectical frameworks. an overview. *Journal of Applied Logics-IfCoLog Journal of Logics and their Applications*, 4(8):2263–2317, 2017.

[55] Gerhard Brewka and Georg Gottlob. Well-founded semantics for default logic. *Fundamenta Informaticae*, 31(3, 4):221–236, 1997.

[56] Katabzyna Budzynska and Serena Villata. Processing natural language argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 577–627. College Publications, 2018.

[57] Martin Caminada. On the issue of reinstatement in argumentation. In Michael Fisher, Wiebe van der Hoek, Boris Konev, and Alexei Lisitsa, editors, *Proceedings of the 10th European Conference on Logics in Artificial Intelligence (JELIA'06)*, volume 4160 of *Lecture Notes in Computer Science*, pages 111–123. Springer, 2006.

[58] Martin Caminada. Argumentation semantics as formal discussion. In Pietro Baroni, Dov Gabay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 487–518. College Publications, 2018.

[59] Martin Caminada. Rationality postulates: Applying argumentation theory for non-monotonic reasoning. In Pietro Baroni, Dov Gabay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 771–795. College Publications, 2018.

[60] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171(5):286–310, 2007.

[61] Martin Caminada, Walter Carnielli, and Paul Dunne. Semi-stable semantics. *Journal of Logic and Computation*, 22(5):1207–1254, 2011.

[62] Martin Caminada and Dov Gabbay. A logical account of formal argumentation. *Studia Logica*, 93(2):109–145, 2009.

[63] Martin Caminada, Sanjay Modgil, and Nir Oren. Preferences and unrestricted rebut. In Simon Parsons, Nir Oren, Chris Reed, and Federico Cerutti, editors, *Proceedings of the 5th International Conference on Computation Models of Argument (COMMA'14)*, Frontiers in Artificial Intelligence and Applications 266, pages 209–220. IOS Press, 2014.

[64] Martin Caminada, Samy Sá, João Alcântara, and Wolfgang Dvořák. On the equivalence between logic programming semantics and argumentation semantics. *International Journal of Approximate Reasoning*, 58:87–111, 2015.

[65] Martin Caminada and Claudia Schulz. On the equivalence between assumption-based argumentation and logic programming. *Journal of Artificial Intelligence Research*, 60:779–825, 2017.

[66] Martin Caminada and Yining Wu. On the limitations of abstract argumentation. In *Proceedings of the 23rd Benelux Conference on Artificial Intelligence (BNAIC'11)*, 2011.

[67] Claudette Cayrol. On the relation between argumentation and non-monotonic coherence-based entailment. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95)*, pages 1443–1448. Morgan Kaufmann, 1995.

[68] Federico Cerutti, Sarah Alice Gaggl, Matthias Thimm, and Johannes Peter Wallner. Foundations of implementations for formal argumentation. *Journal of Applied Logics-IfCoLog Journal of Logics and their Applications*, 4(8):2623–2706, 2017.

[69] Federico Cerutti, Sarah Alice Gaggl, Matthias Thimm, and Johannes Peter Wallner. Foundations of implementations for formal argumentation. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 689–767. College Publications, 2018.

[70] Esther Anna Corsi and Christian G. Fermüller. Logical argumentation principles, sequents, and nondeterministic matrices. In *Proceedings of the 6th International Workshop on Logic, Rationality, and Interaction (LORI'17)*, volume 10455 of *Lecture Notes in Computer Science*, pages 422–437. Springer, 2017.

[71] Esther Anna Corsi and Christian G. Fermüller. Connecting fuzzy logic and argumentation frames via logical attack principles. *Soft Computing*, 23(7):2255–2270, 2019.

[72] Kristijonas Čyras, Xiuyi Fan, Claudia Schulz, and Francesca Toni. Assumption-based argu-

mentation: Disputes, explanations, preferences. *Journal of Applied Logics-IfCoLog Journal of Logics and their Applications*, 4(8):2407–2455, 2017.

[73] Kristijonas Čyras, Xiuyi Fan, Claudia Schulz, and Francesca Toni. Assumption-based argumentation: Disputes, explanations, preferences. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 2407–2456. College Publications, 2018.

[74] Kristijonas Čyras and Francesca Toni. Non-monotonic inference properties for assumption-based argumentation. In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Proceedings of the 3rd International Workshop on Theory and Applications of Formal Argument (TAFA'15)*, volume 9524 of *Lecture Notes in Computer Science*, pages 92–111. Springer, 2015.

[75] Kristijonas Čyras and Francesca Toni. ABA+: assumption-based argumentation with preferences. In Chitta Baral, James Delgrande, and Frank Wolter, editors, *Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning (KR'16)*, pages 553–556. AAAI Press, 2016.

[76] Kristijonas Cyras and Francesca Toni. Properties of ABA+ for non-monotonic reasoning. *arXiv preprint arXiv:1603.08714*, 2016.

[77] Marcello D'Agostino and Sanjay Modgil. Classical logic, argumentation and dialectic. *Artificial Intelligence*, 262:15–51, 2018.

[78] Marcello D'Agostino and Sanjay Modgil. A fully rational account of structured argumentation under resource bounds. In Christian Bessiere, editor, *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI'20)*, pages 1841–1847. ijcai.org, 2020.

[79] Florence Dupin de Saint-Cyr, Pierre Bisquert, Claudette Cayrol, and Marie-Christine Lagasquie-Schiex. Argumentation update in YALLA (yet another logic language for argumentation). *International Journal of Approximate Reasoning*, 75:57–92, 2016.

[80] Marc Denecker, Victor Marek, and Mirosław Truszczyński. Approximations, stable operators, well-founded fixpoints and applications in nonmonotonic reasoning. In *Logic-based Artificial Intelligence*, volume 597 of *The Springer International Series in Engineering and Computer Science*, pages 127–144. Springer, 2000.

[81] Marc Denecker, Victor W Marek, and Mirosław Truszczyński. Uniform semantic treatment of default and autoepistemic logics. *Artificial Intelligence*, 143(1):79–122, 2003.

[82] Martin Diller, Johannes Peter Wallner, and Stefan Woltran. Reasoning in abstract dialectical frameworks using quantified boolean formulas. *Argument & Computation*, 6(2):149–177, 2015.

[83] Yannis Dimopoulos, Bernhard Nebel, and Francesca Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141(1-2):57–78, 2002.

[84] Hans van Ditmarsch, Joseph Halpern, Wiebe van der Hoek, and Barteld Kooi. *Handbook of Epistemic Logic*. College Publications, 2015.

[85] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357,

1995.

[86] Phan Minh Dung, Robert Kowalski, and Francesca Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170(2):114–159, 2006.

[87] Phan Minh Dung, Robert Kowalski, and Francesca Toni. Assumption-based argumentation. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 199–218. Springer, 2009.

[88] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10–15):642–674, 2007.

[89] Phan Minh Dung, Tran Cao Son, and Phan Minh Thang. Argumentation-based semantics for logic programs with first-order formulae. In Matteo Baldoni, Amit Chopra, Tran Cao Son, Katsutoshi Hirayama, and Paolo Torroni, editors, *Proceedings of the 19th International Conference on Principles and Practice of Multi-Agent Systems (PRIMA'16)*, volume 9862 of *Lecture Notes in Computer Science*, pages 43–60. Springer, 2016.

[90] Phan Minh Dung and Phan Minh Thang. Closure and consistency in logic-associated argumentation. *Journal of Artificial Inteligence Research*, 49:79–109, 2014.

[91] Wolfgang Dvořák, Stefan Szeider, and Stefan Woltran. Abstract argumentation via monadic second order logic. In Eyke Hüllermeier, Sebastian Link, Thomas Fober, and Bernhard Seeger, editors, *Proceedings of the 6th International Conference on Scalable Uncertainty Management (SUM'12)*, volume 7520 of *Lecture Notes in Computer Science*, pages 85–98. Springer, 2012.

[92] Sjur Dyrkolbotn. On a formal connection between truth, argumentation and belief. In Margot Colinet, Sophia Katrenko, and Rasmus K. Rendsvig, editors, *Pristine Perspectives on Logic, Language, and Computation - (ESSLLI '12) and (ESSLLI'13) Student Sessions. Selected Papers*, volume 8607 of *Lecture Notes in Computer Science*, pages 69–90. Springer, 2014.

[93] Vasiliki Efstathiou and Anthony Hunter. Algorithms for generating arguments and counterarguments in propositional logic. *International Journal of Approximate Reasoning*, 52(6):672–704, 2011.

[94] Uwe Egly and Stefan Woltran. Reasoning in argumentation frameworks using quantified boolean formulas. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Proceedings of the 1st International Conference on Computational Models of Argument (COMMA'06)*, volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 133–144. IOS Press, 2006.

[95] Christian Eichhorn. *Rational Reasoning with Finite Conditional Knowledge Bases*. Springer, 2018.

[96] Thomas Eiter and Georg Gottlob. The complexity of logic-based abduction. *Journal of the ACM*, 42(1):3–42, 1995.

[97] Jorge Fandinno and Luis Fariñas del Cerro. Constructive logic covers argumentation and logic programming. In Michael Thielscher, Francesca Toni, and Frank Wolter, editors, *Proceedings of the 16th International Conference on Principles of Knowledge Representation and Reasoning (KR'18)*, pages 128–137. AAAI Press, 2018.

[98] Dov Gabbay. Theoretical foundations for non-monotonic reasoning in expert systems. In

Krzysztof R. Apt, editor, *Proceedings of the Conference on Logics and models of concurrent systems*, volume 13 of *NATO ASI Series*, pages 439–457. Springer, 1985.

[99] Dov Gabbay. Modal provability foundations for negation by failure. In Peter Schroeder-Heister, editor, *Proceedings of the International Workshop on Extensions of Logic Programming*, volume 475 of *Lecture Notes in Computer Science*, pages 179–222. Springer, 1990.

[100] Dov Gabbay. Modal provability foundations for argumentation networks. *Studia Logica*, 93(2–3):181–198, 2009.

[101] Dov Gabbay. Dung's argumentation is essentially equivalent to classical propositional logic with the Peirce–Quine dagger. *Logica Universalis*, 5(2):255–318, 2011.

[102] Dov Gabbay and Murdoch Gabbay. The attack as strong negation, Part I. *Logic Journal of the IGPL*, 23(6):881–941, 2015.

[103] Dov Gabbay and Murdoch Gabbay. The attack as intuitionistic negation. *Logic Journal of the IGPL*, 24(5):807–837, 2016.

[104] Dov Gabbay, John Horty, and Xavier Parent. *Handbook of Deontic Logic and Normative Systems*. College Publications, 2013.

[105] Dov Gabbay and Valentin Shehtman. Products of modal logics, Part 1. *Logic Journal of the IGPL*, 6(1):73–146, 1998.

[106] Alejandro García and Guillermo Simari. Defeasible logic programming: an argumentative approach. *Theory and Practice of Logic Programming*, 4(1–2):95–138, 2004.

[107] Tjerk Gauderis and Frederik Van De Putte. Abduction of generalizations. *Theoria. An International Journal for Theory, History and Foundations of Science*, 27(3):345–363, 2012.

[108] Hector Geffner and Judea Pearl. Conditional entailment: Bridging two approaches to default reasoning. *Artificial Intelligence*, 53(2-3):209–244, 1992.

[109] Michael Gelfond and Vladimir Lifschitz. Classical negation in logic programs and disjunctive databases. *New generation computing*, 9(3-4):365–385, 1991.

[110] Gerhard Gentzen. Untersuchungen über das logische Schließen I, II. *Mathematische Zeitschrift*, 39:176–210, 405–431, 1934.

[111] Nikos Gorogiannis and Anthony Hunter. Instantiating abstract argumentation with classical logic arguments: Postulates and properties. *Artificial Intelligence*, 175(9–10):1479–1497, 2011.

[112] Diana Grooters and Henry Prakken. Two aspects of relevance in structured argumentation: Minimality and paraconsistency. *Journal of Artificial Inteligence Research*, 56:197–245, 2016.

[113] Davide Grossi. On the logic of argumentation theory. In Wiebe van der Hoek, Gal Kaminka, Yves Lespérance, Michael Luck, and Sandip Sen, editors, *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'10)*, pages 409–416. IFAAMAS, 2010.

[114] Davide Grossi. Abstract argument games via modal logic. *Synthese*, 190:5–29, 2013.

[115] Davide Grossi and Wiebe van der Hoek. Justified beliefs by justified arguments. In Chitta Baral, Giuseppe De Giacomo, and Thomas Eiter, editors, *Proceedings of the 14th International Conference on Principles of Knowledge Representation and Reasoning (KR'14)*. AAAI

Press, 2014.

[116] James Hawthorne and David Makinson. The quantitative/qualitative watershed for rules of uncertain inference. *Studia Logica*, 86(2):247–297, 2007.

[117] Jesse Heyninck and Ofer Arieli. On the semantics of simple contrapositive assumption-based argumentation frameworks. In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *Proceedings of the 7th International Conference on Computation Models of Argument (COMMA'18)*, volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 9–20. IOS Press, 2018.

[118] Jesse Heyninck and Ofer Arieli. An argumentative characterization of disjunctive logic programming. In Paulo Moura Oliveira, Paulo Novais, and Luís Paulo Reis, editors, *Proceedings of the 19the Conference on Progress in Artificial Intelligenc (EPIA'19)*, volume 11805 of *Lecture Notes in Computer Science*, pages 526–538. Springer, 2019.

[119] Jesse Heyninck and Ofer Arieli. Simple contrapositive assumption-based frameworks. In Marcello Balduccini, Yuliya Lierler, and Stefan Woltran, editors, *Proceedings of the 15th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'19)*, volume 11481 of *Lecture Notes in Computer Science*, pages 75–88. Springer, 2019.

[120] Jesse Heyninck and Ofer Arieli. Argumentative reflections of approximation fixpoint theory. *Computational Models of Argument: Proceedings of COMMA 2020*, 326:215, 2020.

[121] Jesse Heyninck and Ofer Arieli. Simple contrapositive assumption-based frameworks. *Journal of Approximate Reasoning*, 121:103–124, 2020.

[122] Jesse Heyninck, Gabriele Kern-Isberner, and Matthias Thimm. On the correspondence between abstract dialectical frameworks and nonmonotonic conditional logics. In *Proceedings of the 33rd International Florida Artificial Intelligence Research Society Conference (FLAIRS'20)*, 2020.

[123] Jesse Heyninck and Christian Straßer. Relations between assumption-based approaches in nonmonotonic logic and formal argumentation. In *Proceedings of the 16th International Workshop on Non-Monotonic Reasoning (NMR'16)*, 2016.

[124] Jesse Heyninck and Christian Straßer. Revisiting unrestricted rebut and preferences in structured argumentation. In Carles Sierra, editor, *Proceedings of the 26th International Joint Conference on Artificial Intelligence, (IJCAI'17)*, pages 1088–1092. ijcai.org, 2017.

[125] Jesse Heyninck and Christian Straßer. A fully rational argumentation system for preordered defeasible rules. In Edith Elkind, Manuela Veloso, Noa Agmon, and Matthew Taylor, editors, *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS'19)*, pages 1704–1712. International Foundation for Autonomous Agents and Multiagent Systems, 2019.

[126] Jesse Heyninck and Christian Straßer. A comparative study of assumption-based approaches to reasoning with priorities. *Journal of Applied Logic – IfCoLog Journal of Logics and their Applications*, 8(3):737–808, 2021.

[127] Jesse Heyninck and Christian Straßer. Rationality and maximal consistent sets for a fragment of ASPIC$^+$ without undercut. *Argument and Computation*, 12(1):3–47, 2021.

[128] Jaakko Hintikka. *Knowledge and Belief: An Introduction to the Logic of the Two Notions.*

Texts in philosophy. King's College Publications, 2005. Reprint.

[129] John Horty. Moral dilemmas and nonmonotonic logic. *Journal of Philosophical Logic*, 23:35–65, 1994.

[130] Tomi Janhunen. On the intertranslatability of autoepistemic, default and priority logics, and parallel circumscription. In Jürgen Dix, Luis Fariñas del Cerro, and Ulrich Furbach, editors, *Proceedings of the European Workshop on Logics in Artificial Intelligence (JELIA'98)*, volume 1489 of *Lecture Notes in Computer Science*, pages 216–232. Springer, 1998.

[131] Sébastien Konieczny and Ramón Pino Pérez. Merging information under constraints: a logical framework. *Journal of Logic and Computation*, 12(5):773–808, 2002.

[132] Kurt Konolige. On the relation between default and autoepistemic logic. 35(3):343–382, 1988.

[133] Sarit Kraus, Daniel Lehmann, and Menachem Magidor. Nonmonotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence*, 44(1):167 – 207, 1990.

[134] Daniel Lehmann and Menachem Magidor. What does a conditional knowledge base entail? *Artificial Intelligence*, 55:1–60, 1992.

[135] Zimi Li, Nir Oren, and Simon Parsons. On the links between argumentation-based reasoning and nonmonotonic reasoning. In Elizabeth Black, Sanjay Modgil, and Nir Oren, editors, *Proceedings of the 4th International Workshop on Theory and Applications of Formal Argument (TAFA'17)*, volume 10757 of *Lecture Notes in Computer Science*, pages 67–85. Springer, 2018.

[136] Jerzy Łos and Roman Suzsko. Remarks on sentential logics. *Indagationes Mathematicae*, 20:177–183, 1958.

[137] David Makinson. General patterns in nonmonotonic reasoning. In D. Gabbay, C. Hogger, and J. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 35–110. Oxford Science Publications, 1994.

[138] David Makinson. Bridges between classical and nonmonotonic logic. *Logic Journal of the IGPL*, 11(1):69–96, 2003.

[139] David Makinson and Leendert Van Der Torre. Constraints for Input/Output logics. *Journal of Philosophical Logic*, 30:155–185, 2001.

[140] Rob Malouf. Maximal consistent subsets. *Computational Linguistics*, 33(2):153–160, 2007.

[141] Wiktor Marek and Mirosław Truszczyński. Reflexive autoepistemic logic and logic programming. In Luís Moniz Pereira and Anil Nerode, editors, *Proceedings of the 2nd International Workshop on Logic Programming and Nonmonotonic Reasoning (LPNMR'93)*, pages 115–131. MIT Press, 1993.

[142] Joke Meheus and Diderik Batens. A formal logic for abductive reasoning. *Logic Journal of IGPL*, 14(2):221–236, 2006.

[143] Sanjay Modgil. Reasoning about preference in argumentation frameworks. *Artificial Intelligence*, 173(9-10):901–934, 2009.

[144] Sanjay Modgil and Martin Caminada. Proof theories and algorithms for abstract argumentation frameworks. In Guillermo Simari and Iyad Rahwan, editors, *Argumentation in Artificial Intelligence*, pages 105–132. Springer, 2009.

[145] Sanjay Modgil and Henry Prakken. A general account of argumentation with preferences. *Artificial Intelligence*, 195:361–397, 2013.

[146] Sanjay Modgil and Henry Prakken. The ASPIC+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.

[147] Sanjay Modgil and Henry Prakken. Abstract rule-based argumentation. In Pietro Baroni, Dov Gabay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 287–364. College Publications, 2018.

[148] Robert Moore. Semantic considerations on nonmonotonic logic. *Artificial Intelligence*, 25(1):75–94, 1985.

[149] David Pearce. Equilibrium logic. *Annals of Mathematics and Artificial Intelligence*, 47(1–2):3–41, 2006.

[150] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument & Computation*, 1(2):93–124, 2010.

[151] Henry Prakken. Historical overview of formal argumentation. In Pietro Baroni, Dov Gabay, Massimiliano Giacomin, and Leon van der Torre, editors, *Handbook of Formal Argumentation*, pages 75–143. College Publications, 2018.

[152] Henry Prakken and Gerard Vreeswijk. Logics for defeasible argumentation. In Dov Gabbay and Franz Guenthner, editors, *Handbook of Philosophical Logic*, pages 219–318. Springer, 2002.

[153] Henry Prakken and Michiel de Winter. Abstraction in argumentation: Necessary but dangerous. In Sanjay Modgil, Katarzyna Budzynska, and John Lawrence, editors, *Proceedings of the 7th International Conference on Computation Models of Argument (COMMA'18)*, volume 305 of *Frontiers in Artificial Intelligence and Applications*, pages 85–96. IOS Press, 2018.

[154] Raymond Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1-2):81–132, 1980.

[155] Nicholas Rescher and Ruth Manor. On inference from inconsistent premises. *Theory and Decision*, 1:179–217, 1970.

[156] Claudia Schulz, Ken Satoh, and Francesca Toni. Characterising and explaining inconsistency in logic programs. In Francesco Calimeri, Giovambattista Ianni, and Miroslaw Truszczynski, editors, *Proceedings of the 13th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR'15)*, volume 9345 of *Lecture Notes in Computer Science*, pages 467–479. Springer, 2015.

[157] Claudia Schulz and Francesca Toni. Logic programming in assumption-based argumentation revisited-semantics and graphical representation. In Blai Bonet and Sven Koenig, editors, *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*, pages 1569–1575. AAAI Press, 2015.

[158] Claudia Schulz and Francesca Toni. Justifying answer sets using argumentation. *Theory and Practice of Logic Programming*, 16(1):59–110, 2016.

[159] Yi-Dong Shen, Kewen Wang, Thomas Eiter, Michael Fink, Christoph Redl, Thomas Krennwallner, and Jun Deng. FLP answer set semantics without circular justifications for general logic programs. *Artificial Intelligence*, 213:1–41, 2014.

[160] Chenwei Shi, Sonja Smets, and Fernando R. Velázquez-Quesada. Argument-based belief

in topological structures. In Jérôme Lang, editor, *Proceedings of the 16th Conference on Theoretical Aspects of Rationality and Knowledge (TARK'17)*, volume 251 of *EPTCS*, pages 489–503, 2017.

[161] Chenwei Shi, Sonja Smets, and Fernando R. Velázquez-Quesada. Beliefs supported by binary arguments. *Journal of Applied Non-Classical Logics*, 28(2–3):165–188, 2018.

[162] Yoav Shoham. *Reasoning About Change: Time and Causation from the Standpoint of Artificial Intelligence*. MIT Press, 1988.

[163] Robert Stalnaker. What is a nonmonotonic consequence relation? *Fundamenta Informaticae*, 21(1, 2):7–21, 1994.

[164] Hannes Strass. Approximating operators and semantics for abstract dialectical frameworks. *Artificial Intelligence*, 205:39–70, 2013.

[165] Hannes Strass. On the relative expressiveness of argumentation frameworks, normal logic programs and abstract dialectical frameworks. *arXiv preprint arXiv:1405.0805*, 2014.

[166] Christian Straßer. Adaptively applying modus ponens in conditional logics of normality. *Journal of Applied Non-Classical Logics*, 22(1–2):125–148, 2012.

[167] Christian Straßer. *Adaptive Logic and Defeasible Reasoning. Applications in Argumentation, Normative Reasoning and Default Reasoning.* Springer, 2014.

[168] Christian Straßer and Ofer Arieli. Normative reasoning by sequent-based argumentation. *Journal of Logic and Computation*, 29(3):387–415, 2019.

[169] Christian Straßer and Dunja Šešelja. Towards the proof-theoretic unification of Dung's argumentation framework: an adaptive logic approach. *Journal of Logic and Computation*, 21(2):133–156, 2010.

[170] Alfred Tarski. *Introduction to Logic*. Oxford University Press, 1941.

[171] Francesca Toni. A tutorial on assumption-based argumentation. *Argument & Computation*, 5(1):89–117, 2014.

[172] Alasdair Urquhart. Many-valued logic. In Dov Gabbay and Franz Guenthner, editors, *Handbook of Philosophical Logic*, volume II, pages 249–295. Kluwer, 2001. Second edition.

[173] Frederik Van De Putte. Default assumptions and selection functions: A generic framework for non-monotonic logics. In Félix Castro, Alexander Gelbukh, and Miguel González, editors, *Proceedings of the 12th Mexican International Conference on Advances in Artificial Intelligence and Its Applications (MICAI'13)*, volume 8265 of *Lecture Notes in Computer Science*. Springer, 2013.

[174] Frederik Van De Putte and Christian Straßer. A logic for prioritized normative reasoning. *Journal of Logic and Computation*, 23(3):563–583, 2013.

[175] Maarten Van Dyck. Causal discovery using adaptive logics. Towards a more realistic heuristics for human causal learning. *Logique et Analyse*, 185–188:5–32, 2004.

[176] Bart Van Kerckhove and Guido Vanackere. Vagueness-adaptive logic: A pragmatical approach to sorites paradoxes. *Studia Logica*, 75(3):383–411, 2003.

[177] Srdjan Vesic. Identifying the class of maxi-consistent operators in argumentation. *Journal of Artificial Intelligence Research*, 47:71–93, 2013.

[178] Serena Villata, Guido Boella, Dov M Gabbay, Leendert van der Torre, and Joris Hulstijn.

A logic of argumentation for specification and verification of abstract argumentation frameworks. *Annals of Mathematics and Artificial Intelligence*, 66(1-4):199–230, 2012.

[179] Georg von Wright. Deontic logic. *Mind*, 60(237):1–15, 1951.

[180] Toshiko Wakaki. Assumption-based argumentation equipped with preferences and its application to decision making, practical reasoning, and epistemic reasoning. *Computational Intelligence*, 33(4):706–736, 2017.

[181] Yisong Wang, Fangzhen Lin, Mingyi Zhang, and Jia-Huai You. A well-founded semantics for basic logic programs with arbitrary abstract constraint atoms. In Jörg Hoffmann and Bart Selman, editors, *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI'2012)*, pages 835–841. AAAI Press, 2012.

[182] Erik Weber and Dagmar Provijn. A formal analysis of diagnosis and diagnostic reasoning. *Logique et Analyse*, 42(165–166):161–180, 1999.

[183] Yining Wu and Mikołaj Podlaszewski. Implementing crash-resistance and non-interference in logic-based argumentation. *Journal of Logic and Computation*, 25(2):303–333, 2014.

A Proofs

Below we provide proofs to propositions that appear in the chapter and to the best of our knowledge have not been fully proven yet in the literature.

Proposition 88. *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a propositional logic. The entailments $\mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}}$ and $\mathrel{|\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ are $\Cup$-cautiously cumulative and $\uplus$-cumulative.*

Proof. The properties $\sqcup$-**(C)REF**, **RW**, $\sqcup$-**LLE** follow directly from Definition 44. Note that for $\Cup$, full reflexivity does not hold since for an $\vdash$-inconsistent formula ϕ, $\mathsf{MCS}^{\emptyset}(\{\phi\}) = \{\emptyset\}$. The properties $\sqcup$-**CC** and $\sqcup$-**CM** follow for $\mathrel{|\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ and $\mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}}$ by Lemma 132 and Corollary 133. We paradigmatically show the case for $\mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}}$ and $\sqcup = \Cup$: Suppose that $\mathcal{S}', \mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} \psi$. Then the following equivalences hold: $\mathcal{S}', \mathcal{S} \mathrel{|\!\sim}^{\mathfrak{L}}_{\cap\mathsf{mcs}} \phi$, iff $\bigcap \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S}) \vdash \phi$, iff (by Corollary 133 and since $\bigcap \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S}) \vdash \psi$ by the supposition) $\bigcap \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\psi\}) \vdash \phi$, iff $\mathcal{S}', \mathcal{S} \cup \{\psi\} \mathrel{|\!\sim}_{\cap\mathsf{mcs}} \phi$. □

Lemma 132. *If $\langle \mathcal{S}', \mathcal{S} \rangle \mathrel{|\!\sim}_{\Cap\mathsf{mcs}} \psi$. Then:*

1. $\mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\psi\}) = \{\mathcal{T} \cup \{\psi\} \mid \mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})\}$, *and*
2. $\mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S}) = \mathsf{MCS}^{\mathcal{S}' \cup \{\psi\}}_{\mathfrak{L}}(\mathcal{S})$.

Proof. Item 1, $\subseteq$: Suppose that $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\psi\})$. Thus, $\mathcal{T} \cap \mathcal{S}$ is a $\vdash$-consistent subset of $\mathcal{S}$, given $\mathcal{S}'$. Assume that there is a $\mathcal{T}' \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})$ such that $\mathcal{T} \cap \mathcal{S} \subsetneq \mathcal{T}'$. By the supposition, $\mathcal{T}' \vdash \psi$. Thus, $\mathcal{T}' \cup \{\psi\}$ is a $\vdash$-consistent subset of $\mathcal{S} \cup \{\psi\}$, given $\mathcal{S}'$. But since $\mathcal{T} \subsetneq \mathcal{T}' \cup \{\psi\}$, this is a contradiction to the $\subseteq$-maximal consistency of $\mathcal{T}$. Thus,

$\mathcal{T} \cap \mathcal{S} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})$. By the assumption again, $\mathcal{T} \vdash \psi$, and so $\mathcal{T}' = (\mathcal{T} \cap \mathcal{S}) \cup \{\psi\}$ is an element of the set in the right-hand side of the equation of Item 1.

Item 1, $\supseteq$: Suppose that $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})$. Thus, $\mathcal{T}$ is a $\vdash$-consistent subset of $\mathcal{S}$, given $\mathcal{S}'$. Since $\langle \mathcal{S}', \mathcal{S} \rangle \mid\!\sim_{\Cap\mathsf{mcs}} \psi$, we have that $\mathcal{T}, \mathcal{S}' \vdash \psi$ and so $\mathcal{T} \cup \{\psi\}$ is a $\vdash$-consistent subset of $\mathcal{S} \cup \{\psi\}$, given $\mathcal{S}'$. Assume for a contradiction that there is a proper superset $\mathcal{T}' \supsetneq (\mathcal{T} \cup \{\psi\}$ such that $\mathcal{T}' \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\psi\})$. Then, $\mathcal{T} \subsetneq (\mathcal{T}' \cap \mathcal{S})$ and $\mathcal{T}' \cap \mathcal{S}$ is a $\vdash$-consistent subset of $\mathcal{S}$ given $\mathcal{S}'$, which contradicts the $\subseteq$-maximal consistency of $\mathcal{T}$.

Item 2, $\supseteq$: Suppose that $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\psi\}}(\mathcal{S})$. Thus, $\mathcal{T}$ is a $\vdash$-consistent subset of $\mathcal{S}$ given $\mathcal{S}' \cup \{\psi\}$, and so also given $\mathcal{S}'$. Assume that there is a set $\mathcal{T}' \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})$ such that $\mathcal{T} \subsetneq \mathcal{T}'$. Thus, $\mathcal{T}'$ is $\vdash$-inconsistent with ψ (given $\mathcal{S}'$) since otherwise $\mathcal{T}'$ is $\vdash$-consistent with $\mathcal{S}$ given $\mathcal{S}' \cup \{\psi\}$ in contrast to $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\psi\}}_{\mathfrak{L}}(\mathcal{S})$. Thus, $\mathcal{T}', \mathcal{S}', \psi \vdash \mathsf{F}$. By the main supposition also $\mathcal{T}', \mathcal{S}' \vdash \psi$. Thus, by transitivity, $\mathcal{T}', \mathcal{S}' \vdash \mathsf{F}$ which is a contradiction to the choice of $\mathcal{T}'$. Thus, $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}(\mathcal{S})$.

Item 2, $\subseteq$: The proof is similar to that of the previous item. Briefly, suppose that $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S})$. Since $\langle \mathcal{S}', \mathcal{S} \rangle \mid\!\sim_{\Cap\mathsf{mcs}} \psi$, necessarily $\mathcal{T}$ is a $\vdash$-consistent subset of $\mathcal{S}$, given $\mathcal{S}' \cup \{\psi\}$, and trivially then $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\psi\}}_{\mathfrak{L}}(\mathcal{S})$. □

The following corollary follows immediately in view of the fact that $\mid\!\sim^{\mathfrak{L}}_{\cap\mathsf{mcs}}$ is contained in $\mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$.

Corollary 133. *If $\langle \mathcal{S}', \mathcal{S} \rangle \mid\!\sim_{\cap\mathsf{mcs}} \psi$ then Items 1 and 2 of Lemma 132 hold.*

Proposition 89. *Let $\mathfrak{L} = \langle \mathcal{L}, \vdash \rangle$ be a propositional logic and let $\sqcup \in \{\Cup, \uplus\}$. The entailment $\mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ is $\sqcup$-preferential.*

Proof. The proposition follows by Proposition 88 and Lemma 134. □

Lemma 134. *$\mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}}$ satisfies $\sqcup$-**OR**.*

Proof. We first consider the case $\sqcup = \Cup$. Suppose that $\langle \mathcal{S}', \mathcal{S} \cup \{\phi_1\} \rangle \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$ and $\langle \mathcal{S}', \mathcal{S} \cup \{\phi_2\} \rangle \mid\!\sim^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$. Let $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_1 \vee \phi_2\})$ and $\mathcal{T}' = \mathcal{T} \cap \mathcal{S}$. If $\mathcal{T}'$ is $\vdash$-inconsistent with $\phi_1 \vee \phi_2$, then $\mathcal{T}' \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_1\}) \cap \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_2\})$ and $\mathcal{T} = \mathcal{T}'$. By the supposition $\mathcal{T}', \mathcal{S}' \vdash \psi$ and so $\mathcal{T}, \mathcal{S}' \vdash \psi$.

If $\mathcal{T}'$ is $\vdash$-consistent with both ϕ_1 and ϕ_2, then $\mathcal{T}' \cup \{\phi_1\} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_1\})$, $\mathcal{T}' \cup \{\phi_2\} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_2\})$, and $\mathcal{T} = \mathcal{T}' \cup \{\phi_1 \vee \phi_2\}$. By the supposition $\mathcal{T}', \phi_1, \mathcal{S}' \vdash \psi$ and $\mathcal{T}', \phi_2, \mathcal{S}' \vdash \psi$. Hence, $\mathcal{T}', \phi_1 \vee \phi_2, \mathcal{S}' \vdash \psi$ and so $\mathcal{T}, \mathcal{S}' \vdash \psi$.

If $\mathcal{T}'$ is $\vdash$-consistent with ϕ_1 but is not $\vdash$-consistent with ϕ_2, then $\mathcal{T}' \cup \{\phi_1\} \in \mathsf{MCS}^{\mathcal{S}'}_{\mathfrak{L}}(\mathcal{S} \cup \{\phi_1\})$, $\mathcal{T} = \mathcal{T}' \cup \{\phi_1 \vee \phi_2\}$, and $\mathcal{S}', \mathcal{T}', \phi_2 \vdash \mathsf{F}$. Thus $\mathcal{S}', \mathcal{T}', \phi_2 \vdash \psi$. By the supposition also $\mathcal{T}', \phi_1, \mathcal{S}' \vdash \psi$ and thus $\mathcal{T}', \phi_1 \vee \phi_2, \mathcal{S}' \vdash \psi$. Hence, $\mathcal{T}, \mathcal{S}' \vdash \psi$.

The case that $\mathcal{T}'$ is $\vdash$-consistent with ϕ_2 but $\vdash$-inconsistent with ϕ_1 is analogous.

Since our case distinction is exhaustive and in every case that $\mathcal{T}, \mathcal{S}' \vdash \psi$, we have $\langle \mathcal{S}', \mathcal{S} \cup \{\phi_1 \vee \phi_2\}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$.

We now consider the case $\sqcup = \uplus$. Suppose that $\langle \mathcal{S}' \cup \{\phi_1\}, \mathcal{S}\rangle \mathrel{|\!\sim}_{\Cap\mathsf{mcs}} \psi$ and also $\langle \mathcal{S}' \cup \{\phi_2\}, \mathcal{S}\rangle \mathrel{|\!\sim}_{\Cap\mathsf{mcs}} \psi$. Let $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\phi_1 \vee \phi_2\}}_{\mathfrak{L}}(\mathcal{S})$. Thus, $\mathcal{T}$ is $\vdash$-consistent with $\phi_1 \vee \phi_2$ in the context of $\mathcal{S}'$. Then, $\mathcal{T}$ is $\vdash$-consistent with ϕ_1 or with ϕ_2. Without loss of generality suppose the former. Hence, $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\phi_1\}}_{\mathfrak{L}}(\mathcal{S})$. By the supposition, $\mathcal{T}, \mathcal{S}', \phi_1 \vdash \psi$. If $\mathcal{T}$ is $\vdash$-consistent with ϕ_2 in the context of $\mathcal{S}'$, also $\mathcal{T} \in \mathsf{MCS}^{\mathcal{S}' \cup \{\phi_2\}}_{\mathfrak{L}}(\mathcal{S})$, and so $\mathcal{T}, \mathcal{S}', \phi_2 \vdash \psi$. Otherwise, $\mathcal{T}, \mathcal{S}', \phi_2 \vdash \mathsf{F}$ and thus $\mathcal{T}, \mathcal{S}', \phi_2 \vdash \psi$. In any case, since $\vee$ is a disjunction with respect to $\vdash$, it holds that $\mathcal{T}, \mathcal{S}', \phi_1 \vee \phi_2 \vdash \psi$. Thus, $\langle \mathcal{S}' \cup \{\phi_1 \vee \phi_2\}, \mathcal{S}\rangle \mathrel{|\!\sim}^{\mathfrak{L}}_{\Cap\mathsf{mcs}} \psi$. □

Received May 2021