Unpacking arguments

Trevor BENCH-CAPON [a], and Bart VERHEIJ [b]
[a] *Department of Computer Science, University of Liverpool*
[b] *Bernoulli Institute of Mathematics, Computer Science and Artificial Intelligence, University of Groningen*

Abstract. Although argumentation is often studied in AI using abstract frameworks, actual debate often shows a dynamic interaction between argument structure and attack. Often intermediates steps in the reasoning are omitted, but it may be these intermediate steps which are the vulnerable parts of the argument. Inspired by Loui and Norman's work on the rationale of arguments, we study the relation between argument structure and attack in terms of the unpacking of arguments. The paper provides an analysis of two kinds of rationales discussed by Loui and Norman. Example dialogues inspired by Dutch tort law are used for illustration.

Keywords. Computational argumentation, Legal reasoning, Argument attack

1. Introduction

Abstract argumentation [1] has been very influential in AI. It provides a rich formal analysis of the evaluation of the status of a set of arguments and is applicable to many application settings, but takes the set of arguments and the attack relations between them as given. When generating arguments from a knowledge base, as is done in ASPIC+ [2], this presents no problems. But when modelling arguments proposed in actual disputes things may be more difficult. Often some intermediate steps of the argument are unstated, so that while a denial that the intermediate step is valid can attack the argument, this is not clear from what has been explicitly stated. That the vulnerable component is unstated makes the argument harder to attack because the weak point needs to be supplied by the audience. To address such interaction between argument structure and attack, Loui and Norman proposed [3] that arguments should be unpacked to uncover their *rationales*, restoring intermediate steps glossed over in the original presentation, so that vulnerabilities can be identified, attacked and, where possible, defended. The idea can also be recognized in the jurisprudential practice of "rational reconstruction".

Legal arguments are typically multistep, as we move from evidence to facts to factors to issues before finally arriving at a decision [4]. As well as being multi-step, decisions in legal cases often turn on preferences and social values promoted [5], as expressed in precedent cases. The preferences are usually not explicit, but must be recognised if the justification is to be properly understood, and may provide a way to critique the decision.

These two aspects of arguments about legal cases, the omission of intermediate steps and the implicit preferences, correspond to two of the rationales identified in [3], the c-rationale (*compression* rationale) and the r-rationale (*resolution* rationale). We believe that unpacking arguments to restore these aspects is important if the arguments are to be

properly understood, attacked and defended. In this paper, therefore, we revisit Loui and Normans's paper from a contemporary perspective.

Section 2 will summarise their paper, with particular attention to the c-rationales and r-rationales. Section 3 will relate their work to subsequent work on argumentation, in particular work on argument structures such as ASPIC+ [2] and DEFLOG [6]. Section 4 gives some discussion and concluding remarks.[1]

2. Interpreting Loui and Norman's work on unpacking arguments

A key idea in [3] is that the rationales used in an argumentative dialogue can be interpreted as the summaries ('compilations') of extended rationales with more structure. Loui and Norman show how, by unpacking such summary rationales, new argument moves are possible. Thus if someone claims *C because of reason R*, an opponent may identify an immediate step so that the claim becomes *C because of S, and S because of reason R*. Now the opponent can argue that, despite *R*, *S* does not hold for reason *T*, invalidating the conclusion *C*. A defence against this is to provide a different unpacking, claiming that *C* holds because of *U* which is established by *R*, even when *T* is true.
Here a proponent makes an argument, and the opponent unpacks that argument in a certain way, and uses that unpacking to make an attack. The proponent concedes the attack, but disagrees with the unpacking, thereby defending the original position.

Loui and Norman distinguish rationales for rules and rationales for decisions. In the authors' terminology, rule rationales express mechanisms for adopting a rule, while decision rationales express mechanisms for forming an opinion about the outcome of a case. As kinds of rule rationales, the authors distinguish compression, specialization and fit (referred to as c-rationales, s-rationales and f-rationales). The kinds of decision rationales are disputation and resolution (d-rationales and r-rationales). Another distinction used is that between object-level and meta-level disputation, where c-, s-, and d-rationales occur in object level disputes, and r- and f-rationales in meta-level discussion.

We will consider two of these rationale types which are common in legal cases as noted above. For a rule rationale we will look at c-rationales, where the unpacking restores missing intermediate steps. For a decision rationale we look at r-rationales, which identify the use of preferences.

2.1. Unpacking a compression rationale

Here is an example of a small dialogue in which a compression rationale is unpacked, subsequently attacked and then defended, following the basic format given earlier. The unpacking here has the form of adding an intermediate step, thereby interpreting a one step argument as a two step argument. Our example is from Dutch tort law (for legal background, see, e.g., [8]).

> A: I claim that there is a duty to pay the damages (`dut`) because of the act that resulted in damages (`act`).
> B: Unpacking your reasoning, you seem to claim `dut` because of `act` using the additional intermediate reason that there is a breach of contract (`bre`). I disagree with `bre`, because there

[1]The present paper builds on and extends the discussion of Loui and Norman's paper [3] in [7]. The second author acknowledges support by the NWO Zwaartekracht Hybrid Intelligence project.

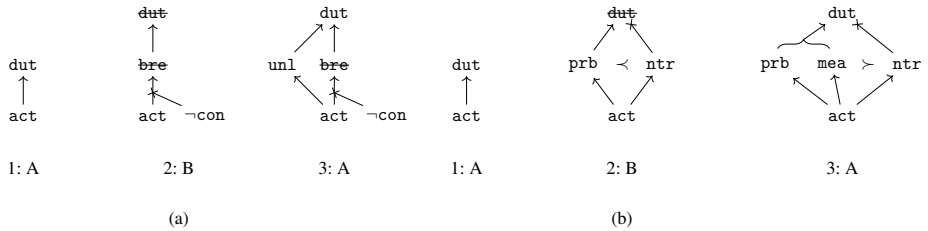

Figure 1. Unpacking (a) a compression rationale and (b) resolution rationale

was no contract (¬con), so there is no support for bre. Hence there is also no support for dut.

A: I agree with your reason ¬con and that hence there is no support for bre. But I was not using bre as an intermediate step supporting dut. Instead I used the intermediate step that the act was unlawful (unl), hence my claim dut because of act.

A graphical summary of the 3-step dialogue is shown in Figure 1. Normal arrows indicate a supporting reason and arrows ending in a cross indicate an attacking reason. All abbreviated statements are considered to be successfully supported, except those that are struck-through. Writing the first argument by A as act $\rightarrow$ dut, B replies in the second move by interpreting the argument as actually having two steps, act $\rightarrow$ bre $\rightarrow$ dut, and then attacks the unpacked argument on the intermediate step using the argument ¬con, so that bre and dut are no longer successfully supported. But then at the third step A concedes that ¬con, while denying the unpacking via bre, instead claiming the unpacking act $\rightarrow$ unl $\rightarrow$ dut, providing an alternative way to support dut, thereby still maintaining act $\rightarrow$ dut. B unpacks A's rationale into two steps, attacking the first step. As a result, for B, dut has no successful support. A accepts everything that B has said, but provides an alternative unpacking of the rationale, via unl, so justifying dut.

2.2. Unpacking a resolution rationale

Again we give a mini-dialogue, illustrating how the idea of resolution rationales and argument attack is approached in [3]. The example unpacks an argument as expressing a preference between two conflicting reasons.

A: I claim that there is a duty to pay the damages (dut) because of the act that resulted in damages (act).

B: Unpacking your reasoning, you seem to claim dut because of act using the weighing of two reasons, one for the duty to pay (the high probability of damages, prb) and one against (the mild nature and low scale of the possible damages, ntr). I disagree with this weighing, because the nature and scale of the possible damages was exceptionally low and so outweighs the high probability of damages. Hence I disagree with your claim dut because of act.

A: I agree with your weighing of the two reasons you mention. But I was using an additional reason for the duty to pay (it was easy to take precautionary measures, mea), and the two reasons for the duty to pay taken together (prb $\wedge$ mea) outweigh the one reason against (ntr). Hence my claim dut because of act.

A graphical summary is shown in Figure 1(b). If we write A's argument in the first dialogue move as act $\rightarrow$ dut, then in the second move B unpacks A's reasoning by claiming that A has weighed prb $\rightarrow$ dut and ntr $\rightarrow$ ¬dut. According to B's weighing

of these two reasons, the conclusion should be $\neg$dut. In the third step, A agrees with B's weighing of the two reasons, but adds a third reason (mea $\rightarrow$ dut) that turns the outcome to the other side, concluding for dut.

2.3. Characteristics of unpacking arguments

From this interpretation we can see that we can use unpacking to attack and defend an argument in a dialogue as follows:

1. *An argument is unpacked.* We have seen two kinds of unpacking: first by interjecting an intermediate step, and second by decomposing the argument as the preference-based resolution of a intermediate conflict of reasons.
2. *A new attack is made.* We have seen that a new intermediate step is attacked by an exceptional circumstance, and that the new interpretation of the argument as the resolution of a conflict of reasons is given an opposite outcome by a reversed preference.
3. *A new defence is made.* We have seen an alternative unpacking of the argument, immune to the new attack, and an extended set of conflicting reasons maintaining the original preference-based resolution.

3. Applying later developments to unpacking arguments

Loui and Norman's paper [3] appeared in the same year as Dung's paper [1], which significantly influenced the subsequent formal and computational study of argumentation. In this section, we study the unpacking of arguments as discussed in Section 2, in terms of subsequent developments.

3.1. Abstract and Structured Argumentation

Following the publication of Dung's formal study of the semantic evaluation of argument attack relations [1], his abstract argumentation frameworks became the standard reference for the semantics of argumentation. In that paper, directed graphs are used to represent argument attack, and the key evaluative principle is that an argument is accepted if there is no accepted attacking argument, and rejected otherwise. The approach is referred to as 'abstract argumentation', because the arguments in Dung's framework have no properties other than the attack relation.

Often, however, it is necessary to consider the structure of the arguments. This is particularly so in law where a claim is not useful without its justification. One approach to representing structure is ASPIC+ [2], in which the arguments can be seen as comprising subarguments, and the attacks distinguished according to the element of the argument (conclusion, premise or inference rule) that is attacked.

3.1.1. Compression

Figure 2 illustrates the above compression rationale dialogue. On the left the developing abstract framework is shown using the notation of [1]. On the right the structured arguments are shown using a representation inspired by the ASPIC+ framework [2]. extended

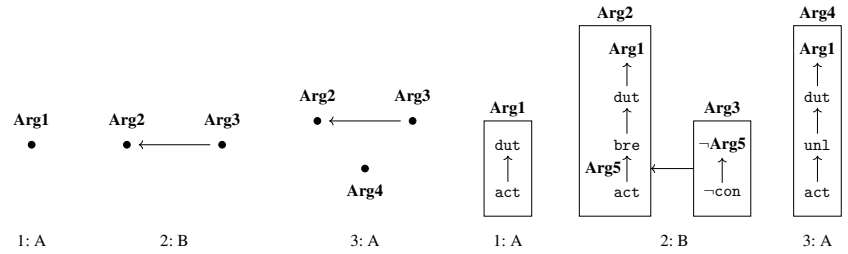

Figure 2. Unpacking a compression rationale using abstract argumentation

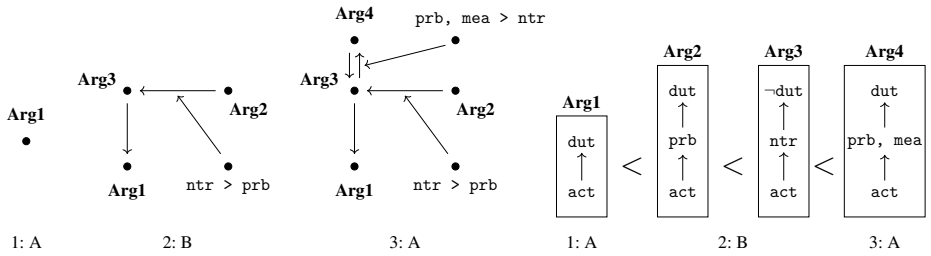

Figure 3. Unpacking a resolution rationale using abstract argumentation

to allow arguments to be conclusions as well as premises. At the first move, there is one argument, Arg1, and there is no attack. After B's move there are two arguments, Arg2 and Arg3. Arg2 is introduced as the justification for Arg1, and so Arg1 moves *inside* the node. Arg3 attacks Arg2 on a subargument (Arg5). so only Arg3 is accepted. In the third move a different justification of Arg1, Arg4, is given. This is not attacked. Now Arg1 is accepted, but only as the claim of the accepted Arg4, rather than as an argument in its own right. At the level of abstraction of Arg1, where all intermediate structure is ignored, Arg2 and Arg4 are indistinguishable. But it matters whether A was originally putting forward Arg1 as an abstraction of Arg3 or Arg4: in law it is important not only that the correct claims are accepted, but that the correct justification for them is given. Note also that in the abstract framework the relationship between Arg1 and Args 2 and 4 is lost.

3.1.2. Resolution

ASPIC+ also allows for preferences between arguments, allowing us to model the resolution rationale. Here B shows that Arg1 requires the resolution of a preference between Arg2 and Arg3, and resolves it so as to deny the claim of Arg1. A now responds by finding a second reason for the original claim, Arg4, so that the *combination* of `prb` and `mea` can be preferred to `ntr`, defeating Arg3 and reinstating Arg1. The situation is shown graphically in Figure 3, with the abstract version on the right shown as an Extended Argumentation Framework [9], which expresses preferences as attacks on attacks.

3.2. Sentence-based argument structure

Second we discuss a sentence-based approach to argument structure. In such an approach, argument support and attack are not treated separately by first determining supporting arguments and then abstracting from them by focusing on attack (as in the ap-

proach in the previous subsection). Instead, all argument structure is expressed by explicit sentences, using dedicated sentences for both support and attack.

Concretely, the argument structure and its evaluation as suggested in Figure 2 can be reconstructed in the DefLog formalism [6], as follows. Three sets of sentences represent the assumptions made by A and B in the three moves:

A_1: $\mathtt{act}$; $\mathtt{act} \rightsquigarrow \mathtt{dut}$ (justifying $\mathtt{dut}$)
B_2: $\mathtt{act}$; $\mathtt{act} \rightsquigarrow \mathtt{bre}$; $\mathtt{bre} \rightsquigarrow \mathtt{dut}$; $\neg\mathtt{con}$; $\neg\mathtt{con} \rightsquigarrow \times(\mathtt{act} \rightsquigarrow \mathtt{bre})$
(neither justifying $\mathtt{bre}$ nor $\mathtt{dut}$ since it follows that $\times(\mathtt{act} \rightsquigarrow \mathtt{bre})$, hence $\mathtt{act} \rightsquigarrow \mathtt{bre}$ is defeated)
A_3: $\mathtt{act}$; $\mathtt{act} \rightsquigarrow \mathtt{bre}$; $\mathtt{bre} \rightsquigarrow \mathtt{dut}$; $\neg\mathtt{con}$; $\neg\mathtt{con} \rightsquigarrow \times(\mathtt{act} \rightsquigarrow \mathtt{bre})$; $\mathtt{act} \rightsquigarrow \mathtt{unl}$; $\mathtt{unl} \rightsquigarrow \mathtt{dut}$
(justifying $\mathtt{unl}$ and $\mathtt{dut}$, but not $\mathtt{bre}$)

Note that in this reconstruction in the second move B revises the commitments made by A in the first move (in the sense that B does not assume A's assumption $\mathtt{act} \rightsquigarrow \mathtt{dut}$), while A in the third move commits to all assumptions made by B in the second (in the sense that all B's assumptions are also assumed by A).

4. Concluding Remarks

In Section 3, we discussed unpacking arguments using developments in computational argument that appeared after [3]. Both abstract argumentation and a sentence based approach can make the unpackings explicit, but neither retains the connection between the unpacked and the unpacking arguments. This could be studied using case models [10].

Finally, unpacking arguments should not be confused with identifying enthymemes. Unpacking identifies additional, possibly dubious, arguments, rather than assumptions. In terms of the knowledge base, what is made explicit is a *rule*, rather than a *statement*.

References

[1] Dung PM. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. Artificial Intelligence. 1995;77:321-57.
[2] Prakken H. An Abstract Framework for Argumentation with Structured Arguments. Argument and Computation. 2010;1(2):93-124.
[3] Loui RP, Norman J. Rationales and Argument Moves. Artificial Intelligence and Law. 1995;3:159-89.
[4] Bench-Capon T, Atkinson K. Using Argumentation Schemes to Model Legal Reasoning. arXiv preprint arXiv:221000315: Presented at 4th European Conference on Argumentation. 2022.
[5] Bench-Capon T, Sartor G. A Model of Legal Reasoning with Cases Incorporating Theories and Values. Artificial Intelligence. 2003;150(1):97-143.
[6] Verheij B. DefLog: on the Logical Interpretation of Prima Facie Justified Assumptions. Journal of Logic and Computation. 2003;13(3):319-46.
[7] Governatori G, , Verheij B, Araszkiewicz M, Bench-Capon T, Francesconi E, et al. Thirty Years of AI and Law: The first decade. AI and Law. 2022 In Press;30(4).
[8] Verheij B, Hage JC, Lodder AR. Logical Tools for Legal Argument: a Practical Assessment in the Domain of Tort. In: Proceedings of the 6th ICAIL. New York (New York): ACM Press; 1997. p. 243-9.
[9] Modgil S. Reasoning about Preferences in Argumentation Frameworks. Artificial Intelligence. 2009;173(9):901-34.
[10] Verheij B. Formalizing Arguments, Rules and Cases. In: Proceedings of the 16th ICAIL. New York (New York): ACM Press; 2017. p. 199-208.