Towards Formalising Argumentation about Legal Cases

Adam Z. Wyner, Trevor J.M. Bench-Capon, and Katie M. Atkinson
Department of Computer Science
University of Liverpool, Liverpool, UK
{azwyner,tbc,katie}@liverpool.ac.uk

ABSTRACT

In this paper we offer an account of reasoning with legal cases in terms of argumentation schemes. These schemes, and undercutting attacks associated with them, are expressed as defeasible rules of inference that will lend themselves to formalisation within the ASPIC+ framework. We begin by modelling the style of reasoning with cases developed by Aleven and Ashley in the CATO project, which describes cases using factors, and then extend the account to accommodate the dimensions used in Rissland and Ashley's earlier HYPO project. Some additional scope for argumentation is then identified and formalised.

1. INTRODUCTION

Legal case-based reasoning (LCBR) has long been a topic of interest in AI and Law, which has evolved a variety of approaches. One important stream began with HYPO [2], and subsequently developed into CATO [1] and IBP [8] in one stream, and CABARET [18] and BankXX [17] in another. More theoretically-oriented research appears in [16], [4], and [6]. In these approaches, a current undecided case is decided by comparing and contrasting features in the current case against precedent cases in a case-base that have similar features. The decision in the "best" precedent case is then taken as the decision into the current case following the legal reasoning principle of *stare decisis*.

In [20], a number of novel legal case-based argumentation schemes designed to reflect reasoning with factors as in Aleven and Ashley's CATO [1] were described, where the focus is to determine how and in what way a precedent case does (or does not) argue in support of a determination in the current case. However, the presentation was semi-formal and not set in an analytic framework which supports reasoning. In this paper, we reanalyse and refine the argumentation schemes of [20] in a form which lends itself to being formally defined in ASPIC+, which will be part of our future work. Moreover, we extend our analysis in novel ways to consider additional aspects about reasoning with factors and dimensions as in HYPO.

The paper advances the state-of-the-art in several respects: legal case-based reasoning with factors is clarified and defeasible legal case-based reasoning is represented in argumentation schemes, where the arguments are well-developed and justified instantiations of schemes that approximate a formal argumentation framework. Furthermore, the schemes are grounded in a knowledge base, and so identifies the information required to carry out the reasoning. Finally, the analysis provides a uniform representation language into which various alternative proposals for LCBR can be cast, compared, integrated, and reasoned with, which will facilitate future exploration to integrate approaches such as [6].

We first set out elements of the formal framework for argumentation that serve as our target representation. In section 3, we discuss case-based reasoning as in CATO. We introduce our running example and the elements of the language we need for the argumentation schemes before presenting CATO style argumentation schemes in the formal framework. In section 4, the analysis is extended to reasoning about dimensions, the relationship between facts and factors, and factor incompatibility. Section 5 discusses additional aspects of the proposal.

2. THE FORMAL SETTING

We first briefly summarise the formal frameworks used in this paper which serve as our target representations. An *abstract argument framework*, as introduced by Dung, [11] is a pair $AF = \langle \mathcal{A}, defeat \rangle$, where $\mathcal{A}$ is a set of arguments and *defeat* a binary relation on $\mathcal{A}$. A subset $\mathcal{B}$ of $\mathcal{A}$ is said to be *conflict-free* if no argument in $\mathcal{B}$ defeats an argument in $\mathcal{B}$ and it is said to be *admissible* if it is both conflict-free and also defends itself against any attack, i.e., if an argument A_1 is in $\mathcal{B}$ and some argument A_2 in A but not in B defeats A_1, then some argument in $\mathcal{B}$ defeats A_2. A *preferred extension* is then a maximal (with respect to set inclusion) admissible set. Dung defines several other types of extensions but they are not used in our model.

Dung's arguments are entirely abstract, with no features other than the defeat relation. In order to enable some content to be given to the arguments, a refinement of Dung's abstract approach, which provides some structure for arguments, was developed in the ASPIC framework, most fully as ASPIC+ in [14]. This framework assumes an unspecified logical language and knowledge base, which may include facts, strict rules, and defeasible rules; it defines arguments as inference trees formed by applying inference rules (which may be either strict or defeasible) to a knowledge base. An argument is thus an inference tree with a root node, which is the *claim* of the argument, and leaves, which are the *premises* of the argument. Leaf nodes are facts in the knowledge base; as premises, they are either *assumptions*, which are unjustified and are defeated if challenged, or *axioms*, which cannot be questioned. An argument A_i can have the claims of arguments $\mathrm{A}_n, \ldots, \mathrm{A}_m$ as premises. The arguments $\mathrm{A}_n, \ldots, \mathrm{A}_m$ are referred to as subarguments of A_i. Formulae, literals and arguments are all in an asymmetric *contrariness*

ICAIL '11, June 6-10, 2011, Pittsburgh, PA

relation; where elements are in symmetric contrariness relations, they are *contradictory*. The contrary of a literal p is $\overline{p}$, and the contrary of a formula ϕ is $\overline{\phi}$. A naming convention is assumed for the rules of arguments such that $\overline{Arg}$ indicates the contrary (inapplicability) of the rule of Arg. To satisfy consistency postulates, it should not possible for arguments that have claims that are contrary are in the same extension. Following [14], a strict argument, such as that labeled (A1), with premises P1, P2, and claim C1 is:

$$\frac{P1, P2}{C1}\ (A1)$$

while a defeasible argument (A2) is:

$$\frac{P1, P2}{\overline{\underline{C1}}}\ (A2)$$

and a defeasible argument (A3) with claim C1 and a defeasible subargument (A4) with claim P1 is:

$$\genfrac{}{}{1.2pt}{}{\genfrac{}{}{1.2pt}{}{P3, P4}{P1,}\ (A4) \quad P2}{C1}\ (A3)$$

The notion of an argument as an inference tree leads to three ways of attacking an argument:

- attacking an inference (*undercut*): an argument B undercuts an argument A if the claim of B is $\overline{A}$.
- attacking a conclusion (*rebuttal*): an argument B rebuts an argument A if the claim of A is ϕ and the claim of B is $\overline{\phi}$.
- attacking a premise (*undermining*): an argument B undermines an argument A if p is a premise of A and the claim of B is $\overline{p}$.

All three attacks apply to defeasible rules, but only undermining can be used against strict rules, since they are universally applicable and guarantee their conclusions whenever their premises hold. To resolve underminings and rebuttals, a preference relation $\prec$ on arguments (to be specified as input) is used, which leads to three corresponding kinds of defeat: undercutting, rebutting and undermining defeat. Note that ASPIC+ distinguishes attack from defeat, so that attacks can be unsuccessful. Essentially, A successfully rebuts (undermines) B if A rebuts (undermines) B and $A \not\prec B$. Then A defeats B if A undercuts B or successfully rebuts or successfully undermines B.

3. CATO ARGUMENTATION SCHEMES

In this section, we provide argumentation schemes for CATO style case-based reasoning in a form which lends itself to being formally defined in ASPIC+, which will be part of our future work. We give a brief overview of CBR as in CATO in section 3.1, introduce our running example in section 3.2, present elements of the language in section 3.3, formalise the argumentation schemes in 3.4, and report the results with respect to our example in 3.5.

3.1 Case-based Reasoning as in CATO

CATO [1], which we focus on in this section, analyses cases in terms of *factors*, where a factor is a stereotypical pattern of facts which predispose the decision in favour of one party or the other in the case; for trade secret law, the domain CATO is designed for, the factors concern trade secret misappropriation and are derived from *Restatement of Torts First, Sec. 757* and the *Uniform Trade Secret Act* (see [2, 1]). As different precedents have different distributions of factors, finding and reasoning about precedents with respect to a current case requires one to examine the combinations of and counter-balancing between factors in the cases. In addition to the factors themselves, there is a *factor hierarchy* in which an *abstract factor* has factors as children, which are reasons *for* and *against* the presence of the abstract factor. Reasoning with the abstract factors and the factors of a case, differences between the cases can sometimes be reconciled. The argumentation schemes discussed in this paper make such reasoning patterns explicit and formal.

A case comparison method for LCBR was introduced in [4], where cases are analysed in terms of *partitions* of case factors. Various distributions of factors amongst the partitions can be used to support or undermine the plaintiff's argument that the current case should be decided in the plaintiff's favour. [20] provided some informally expressed *argumentation schemes* for this partition method, where the schemes are *defeasible reasoning patterns* and the partitions are sets of CATO factors and the factor hierarchy is used. This paper formalises, articulates, and extends this line of research on LCBR.

3.2 Running Example

To clarify the discussion, we provide a running example using *Mason v Jack Daniels Distillery* (indicated with *Mason*) and *M. Bryce and Associates v Gladstone* (indicated with *Bryce*) as analysed in CATO, based on the factors and factor hierarchy in [1]. We give the factors for each case with an indication of the side:

- *Mason*
  - *F1 Disclosure-In-Negotiations (d)*
  - *F6 Security-Measures (p)*
  - *F15 Unique-Product (p)*
  - *F16 Info-Reverse-Engineerable (d)*
  - *F21 Knew-Info-Confidential (p)*
- *Bryce*
  - *F1 Disclosure-In-Negotiations (d)*
  - *F4 Agreed-Not-To-Disclose (p)*
  - *F6 Security-Measures (p)*
  - *F18 Identical-Products (p)*
  - *F21 Knew-Info-Confidential (p)*

We illustrate the formalism with this example.

3.3 Elements of a Language

We begin by defining the terms which we shall use to talk about our cases and which will be used in our underlying Knowledge Base (KB). We will not talk about cases directly, but cases as analysed for use by the CATO system, which is the system which provides the paradigm on which our argumentation will be based. For CATO, a case has a name, a set of factors in favour of the plaintiff, a set of factors in favour of the defendant, and an outcome, which is one of plaintiff, defendant, or ?, which indicates as yet undecided cases. Accordingly we describe cases with the following three binary relations:

- *pFactors(case,setOfFactors).*
- *dFactors(case,setOfFactors).*
- *outcome(case, party).*

With respect to our running example, we have:

- *pFactors(Mason, {F6, F15, F21})*
- *dFactors(Mason, {F1, F16})*
- *pFactors(Bryce, {F4, F6, F18, F21})*
- *dFactors(Bryce, {F1})*

The outcomes, where *Mason* is the current (*curr*) case which is undecided and *Bryce* is the precedent (*prec*) which is decided for the plaintiff, are:

- *outcome(Mason, ?)*
- *outcome(Bryce, plaintiff)*

Additionally a feature of CATO is that factors are organised into a factor hierarchy, with factors being the children of more abstract factors. Thus for every factor we have one or more relations of the form:

- *parentFactor(factor,abstractFactor)*.

While CATO has some intermediate layers in the factor hierarchy, we omit some of them for our current purposes as well as the label of these higher level factors. The abstract factors are also associated with a side: *F102 (p), F105 (d), F115 (p)*.

- *parentFactor(F1, F102)*
- *parentFactor(F4, F102)*
- *parentFactor(F4, F115)*
- *parentFactor(F6, F102)*
- *parentFactor(F15, F105)*
- *parentFactor(F16, F105)*
- *parentFactor(F21, F115)*

Cases are compared with one another in terms of their factors. This gives rise to a further six relations. The first case in the comparison will be assumed to be our current case, *curr*, the second some decided case, *prec*.[1]

- *commonPfactors(curr, prec, setOfFactors)*, where pfactors(curr,P1) and pFactors(prec,P2) and setofFactors = $P1 \cap P2$.
- *commonDfactors(curr, prec, setOfFactors)*, where dfactors(curr,D1) and dFactors(prec,D2) and setofFactors = $D1 \cap D2$.
- *currPfactors(curr, prec, setOfFactors)*, where pfactors(curr,P1) and pFactors(prec,P2) and setofFactors = $P1 \setminus (P1 \cap P2)$.
- *currDfactors(curr, prec, setOfFactors)*, where dfactors(curr,D1) and dFactors(prec,D2) and setofFactors = $D1 \setminus (D1 \cap D2)$.
- *precPfactors(curr, prec, setOfFactors)*, where pfactors(curr,P1) and pFactors(prec,P2) and setofFactors = $P2 \setminus (P1 \cap P2)$.
- *precDfactors(curr, prec, setOfFactors)*, where dfactors(curr,D1) and dFactors(prec,D2) and setofFactors = $D2 \setminus (D1 \cap D2)$.

[1]We intend these definitions to be taken as strict axioms.

With respect to our running example, we have:

- *commonPfactors(Mason, Bryce, {F6, F21}*)
- *commonDfactors(Mason, Bryce, {F1}*)
- *currPfactors(Mason, Bryce, {F15}*)
- *currDfactors(Mason, Bryce, {F16}*)
- *precPfactors(Mason, Bryce, {F4, F18}*)
- *precDfactors(Mason, Bryce, {}*)

These relations are the building blocks for our arguments. The first two are the basis for a comparison and represent what is common between the two cases. The remaining four represent differences, and their effect will depend on the outcome of the previous case and the side for which we are arguing. Suppose we are arguing for the plaintiff: then we can only use precedents with the outcome plaintiff. For such cases, *currPfactors* and *precDfactors* will strengthen the plaintiff's position, since they represent, respectively, plaintiff reasons in *curr* not available in the *prec* and defendant reasons in the *prec* which are not available in *curr*. On the other hand *currDfactors* and *precPfactors* weaken the plaintiff's position in *curr*. Similarly, if arguing for the defendant in the *curr*, *currDfactors* and *precPfactors* strengthen the position and *currPfactors* and *precDfactors* weaken it. The precise nature of the strengthening and weakening will be made clear when we consider the argument schemes based on these different partitions.

Next we need to express that one set of factors, *factorSet1*, is preferred over another, *factorSet2*.

- *preferred(factorSet1,factorSet2)*.

In our analysis, the preference is the claim of a defeasible argumentation scheme *CS2*, which only appears later. We cannot, then, straigthforwardly provide the preference in our running example until section 3.4.

Finally, we need another relation between factors. If factors for a given party share the same parent, then both factors get their force from the fact that the same abstract factor is present in the case. This means that it may be possible to substitute one for another. Similarly if they favour different parties, they may cancel each other out so as to remove the abstract factor from the case. Therefore we have two additional predicates:

- *substitutes(factor1, factor2)*
- *cancels(factor1, factor2)*

We define substitution and cancellation of factors which would benefit the plaintiff as follows, where substitutions apply *between* cases and cancellations apply *within* cases. Substitutions and cancellations for the defendant would be similar, though switching the predicates (and factor sets):

- *substitutes(factor1, factor2)*, where currDfactors (curr, prec, D_c) and factor1 $\in D_c$ and dFactors(prec, D_p) and factor2 $\in D_p$ and parentFactor(factor1, abstract1) and parentFactor(factor2, abstract1).
- *substitutes(factor1, factor2)*, where precPfactors (curr, prec, P_p) and factor1 $\in P_p$ and pFactors (curr, P_c) and factor2 in P_c and parentFactor (factor1, abstract1) and parentFactor (factor2, abstract1).

- *cancels(factor2, factor1)*, where currDfactors (curr, prec, D_c) and factor1 $\in D_c$ and pFactors (curr, P_c) and factor2 $\in P_c$ and parentFactor (factor1, abstract1) and parentFactor (factor2, abstract1).

- *cancels(factor2, factor1)*, where precPfactors (curr, prec, P_p) and factor1 $\in P_p$ and dFactors (precCase, D_p) and factor2 $\in D_p$ and parentFactor (factor1, abstract1) and parentFactor (factor2, abstract1).

In our running example, we have:

- *substitutes(F4, F6)* since precPfactors(Mason, Bryce, {F4, F18}) and F4 $\in$ {F4, F18} and pFactors(Mason, {F6, F15, F21}) and F6 $\in$ {F6, F15, F21} and parentFactor(F4, F102) and parentFactor(F6, F102).

- *cancels(F15, F16)* since currPfactors (Mason, Bryce, {F15}) and F15 $\in$ {F15} and dFactors(Mason, {F1, F16}) and F16 $\in$ {F1, F16} and parentFactor(F15, F105) and parentFactor(F16, F105).

Intuitively, we want to argue that: we should decide *Mason* on the basis of *Bryce* since *Mason* and *Bryce* share several factors (both for plaintiff and defendant); since *Bryce* was decided for the plaintiff so too should *Mason* be decided; and, any differences between them can be argued away by substitution and cancellation.

3.4 CATO style Argument Schemes

In this section, we present the argument schemes built from this language. We will always suppose that we wish to argue the *curr* for the plaintiff.[2] Arguments for the defendant are similar, except that the strengthening and weakening factor partitions are reversed as discussed above. The argument is that the *curr* should be decided for the plaintiff because the common P factors were preferred to the common D factors in the *prec*.[3]

CS1 *Given curr and prec:*

$$\frac{\textit{commonPfactors(curr, prec, P) ,} \\ \textit{commonDfactors(curr, prec, D) , preferred(P, D)}}{\textit{outcome(curr, plaintiff)}}$$

Instantiating *CS1*, where our *curr* is *Mason* and our *prec* is *Bryce*, we have the following argument, indicated with **Mason(Bryce)A1**:

Mason(Bryce)A1

$$\frac{\textit{commonPfactors(Mason, Bryce, \{F6, F21\}) ,} \\ \textit{commonDfactors(Mason, Bryce, \{F1\}) ,} \\ \textit{preferred(\{F6,F21\}, \{F1\})}}{\textit{outcome(Mason, plaintiff)}}$$

We will assume at this point that the information about cases in our KB is correct, or at least beyond dispute; this is relaxed in section 4.2. In ASPIC+ terms this makes them *axioms* and so the first two premises cannot be questioned. The third, however, needs to be established, and this will be done using $CS2$, which we will describe after considering undercutters to $CS1$. There may also be rebuttals, using a variety of argument schemes. We need also to recognise that even if such a preference has been established in the *prec*, it may not be applicable to the *curr*, because the defendant has arguments in the *curr* that were not available in the *prec*. We therefore have the undercutting attack for arguments using $CS1$.

U1.1 *Let A be an argument instantiating $CS1$ given curr and prec, where currDfactors(curr, prec, D):*

$$\frac{\textit{currDfactors(curr, prec, D) , } d \in D}{\overline{A}}$$

Instantiating *U1.1* with *Mason* and *Bryce*, we have an undercutter argument:

Mason(Bryce)A2

$$\frac{\textit{currDfactors(Mason, Bryce, \{F16\}) , F16} \in \textit{\{F16\}}}{\overline{\textit{\textbf{Mason(Bryce)A1}}}}$$

While this presents a challenge to the plaintiff, the argument for the plaintiff can be defended if the distinctions between the cases can be *downplayed*. The undercutting move of $U1.1$ is one way of distinguishing the two cases, and in CATO the abstract factor hierarchy allows us to downplay distinctions. This downplaying can be done in two ways, *substitution* or *cancellation*, corresponding to the two different kinds of extra strength the *curr* may have.

U1.1.1 *Let A be an argument instantiating U1.1 given curr and prec where currDfactors(curr, prec, D_c) and dFactors(prec, D_p) and factor1 $\in D_c$ and factor2 $\in D_p$:*

$$\frac{\textit{substitutes(factor1, factor2)}}{\overline{A}}$$

U1.1.2: *Let A be an argument instantiating U1.1 given curr and prec where currDfactors(curr, prec, D_c) and pFactors(curr, P_c) and factor1 $\in D_c$ and factor2 $\in P_c$:*

$$\frac{\textit{cancels(factor2, factor1)}}{\overline{A}}$$

The idea here is that as the undercutting factor in the *curr* has the same parent as a factor in *prec*, we can substitute for the undercutting factor, where the point is that the abstract factor can be seen to have been applied also in the *prec*; alternatively, the undercutting factor in the *curr* is cancelled out by some other factor in *curr*, so that the abstract factor does not apply. Instantiating *U1.1.2* with our running example and given that we previously determined that *cancels(F15, F16)*, we can form the following argument:

Mason(Bryce)A3

$$\frac{\textit{cancels(F15, F16)}}{\overline{\textit{\textbf{Mason(Bryce)A2}}}}$$

To establish the preference between two sets of factors that is required to justify the third premise of $CS1$, we have $CS2$.

CS2 *Given curr and prec:*

$$\frac{\textit{commonPfactors(curr, prec, P) ,} \\ \textit{commonDfactors(curr, prec, D) ,} \\ \textit{outcome(prec, plaintiff)}}{\textit{preferred(P, D)}}$$

[2] To comply with ASPIC+, the informal conditions given in the schemes ought to be explicit premises of the schemes. Though we continue to use informal conditions in this paper, we will fully comply with ASPIC+ formalism in future work. Furthermore, strictly speaking, schemes should explicitly express the rules they use. For clarity, we leave them implicit.

[3] Note that $CS1$ uses only a subset of the factors from the prec: this is because $CS2$ also encapsulates the rule broadening move as discussed in [18], which is necessary to adapt the *prec* so as to match the *curr*.

Instantiating *CS2*, we have an argument for the preference, as mentioned above:

Mason(Bryce)A4

$$\frac{\textit{commonPfactors(Mason, Bryce, \{F6, F21\})}\ ,\ \textit{commonDfactors(Mason, Bryce, \{F16\})}\ ,\ \textit{outcome(Bryce, plaintiff)}}{\textit{preferred(\{F6,F21\}, \{F1\})}}$$

All of the premises of *CS2* are taken from our database, or straightforward set operations on such data and so represent ASPIC+ axioms which cannot be questioned. It is, however, possible to both rebut and to undercut the argument. We assume that set-theoretic notations and axioms are part of the logical language.

R2.1 *Where preferred(P,D) is the claim of an argument instantiating CS2, given curr from CS2, and given prec2, a precedent with outcome for the defendant:*

$$\frac{\textit{commonPfactors(curr, prec2, P1)}\ ,\ P \subseteq P1\ ,\ \textit{commonDfactors(curr, prec2, D1)}\ ,\ D \subseteq D1\ ,\ \textit{outcome(prec2, defendant)}}{preferred(P, D)}$$

Attacks made using $R2.1$ offer counter examples in which the same comparison was available in a case decided for the defendant, suggesting that the preference is opposite, providing a rebuttal. We do not consider such rebuttals in this paper although they will be open to objections similar to those made against arguments using CS2.

U2.1 *Let* A *be an argument instantiating* $CS2$ *with curr and prec:*

$$\frac{\textit{precPfactors(curr, prec, P)}\ ,\ p \in P}{\overline{A}}$$

Instantiating *U2.1* with *Mason* and *Bryce*, we have two arguments, one for each factor in precPfactors:

Mason(Bryce)A5

$$\frac{\textit{precPfactors(Mason, Bryce, \{F4, F18\})}\ ,\ F4 \in \{F4, F18\}}{\overline{\textbf{\textit{Mason(Bryce)A4}}}}$$

Mason(Bryce)A5′

$$\frac{\textit{precPfactors(Mason, Bryce, \{F4, F18\})}\ ,\ F18 \in \{F4, F18\}}{\overline{\textbf{\textit{Mason(Bryce)A4}}}}$$

$U2.1$, however, undercuts the argument by suggesting that it may have been the additional plaintiff factors available in the prec that tipped the balance, and so distinguishing the *curr* and the *prec*. Like $U1.1$, $U2.1$ can be undercut if we can downplay the distinction.

U2.1.1 *Let A be an argument instantiating U2.1, and given curr and prec, where precPfactors(curr, prec,* P_p*) and pFactors(curr,* P_c*) and factor1* $\in P_p$ *and factor2* $\in P_c$*:*

$$\frac{\textit{substitutes(factor1, factor2)}}{\overline{A}}$$

U2.1.2*: Let A be an argument instantiating U2.1, and given curr and prec, where precPfactors (curr, prec,* P_p*) and dFactors(prec,* D_p*) and factor1* $\in P_p$ *and factor2* $\in D_p$*:*

$$\frac{\textit{cancels(factor1, factor2)}}{\overline{A}}$$

Given that in *Mason* and *Bryce*, substitutes(F4,F6), we instantiate *U2.1.1*, which undercuts *U2.1*:

Mason(Bryce)A6

$$\frac{\textit{substitutes(F4, F6)}}{\overline{\textbf{\textit{Mason(Bryce)A5}}}}$$

At this point we have: the main argument for the plaintiff based on a particular *prec*, comprising an application of a preference and an argument for the preference; undercutters of these two subarguments; and undercutters of some of these undercutting arguments. We may still, however, have some strengths of the *curr* unused, and so we can add some supplementary arguments. The schemes are intended to ensure that the factors have not been used to substitute or cancel an earlier argument.[4]

CS3 *Given curr and prec and commonPfactors (curr, prec, P) and commonDfactors (curr, prec, D) and currPfactors (curr, prec, P2) and dFactors (curr, D2) and pFactors (prec, P3):*

$$\frac{\textit{preferred(P, D)},\ \exists p\ [p \in P2 \mid ((\neg \exists d \in D2,\ \textit{cancels(d,p)}),\ (\neg \exists p3 \in P3,\ \textit{substitutes (p,p3)}))]}{\textit{outcome (curr,plaintiff)}}$$

CS4 *Given curr and prec and commonPfactors (curr, prec, P) and commonDfactors (curr, prec, D) and precDfactors (curr, prec, D2) and pFactors (prec, P2) and dFactors (curr, D3):*

$$\frac{\textit{preferred(P, D)},\ \exists d\ [d \in D2 \mid ((\neg \exists p \in P2,\ \textit{cancels(p,d)}),\ (\neg \exists d3 \in D3,\ \textit{substitutes(d,d3)}))]}{\textit{outcome(curr,plaintiff)}}$$

These arguments make use of the factors not used to substitute or cancel factors cited to undercut the arguments for the plaintiff based on the *prec*. Thus $CS3$ points to additional plaintiff factors in the *curr*, that were not used to cancel or substitute for factors otherwise used. $CS4$ does the same thing in terms of factors that made the defendant's case stronger in the *prec*. Note that both require the *preferred(P, D)* as a premise, and so must use $CS2$ to establish this.

3.5 Running Example Result

We have the following defeat relations between arguments, which are represented in Figure 1, where we indicate that **Mason (Bryce) A4** is a subargument of **Mason (Bryce) A2**:

- attack(**Mason(Bryce)A2**, **Mason(Bryce)A1**)
- attack(**Mason(Bryce)A3**, **Mason(Bryce)A2**)
- attack(**Mason(Bryce)A5**, **Mason(Bryce)A4**)
- attack(**Mason(Bryce)A5′**, **Mason(Bryce)A4**)
- attack(**Mason(Bryce)A6**, **Mason(Bryce)A5**)

Following [11] and the assumption in ASPIC+ that an attack on a subargument is an attack on the argument, the preferred extension is {**Mason(Bryce)A6**, **Mason(Bryce)A5′**, **Mason(Bryce)A3**}. This

[4]There are some issues relating to unification and set construction that will be addressed in the future work. These issues also apply to several later schemes.

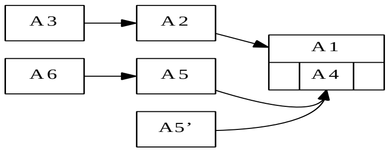

Figure 1: Mason(Bryce) Argument Graph

means that we cannot use *Bryce* as a precedent to argue for the plaintiff in *Mason* since while they have common factors, *Bryce* was decided in favour of the plaintiff, and the preference for the decision holds, we have not succeeded in eliminating all significant distinctions; in particular, we have not found a substitution for F18. Were we to have found such a substitution, then we would have an attack on **Mason(Bryce)A5′**, in which case, the extension would be {**Mason(Bryce)A6**, **Mason(Bryce)A4**, **Mason(Bryce)A3**, **Mason(Bryce)A1**} and *Bryce* would have been a good precedent for *Mason* as informally discussed previously.

Though this might appear to be a negative result, a positive result rests on finding an argument against **Mason(Bryce)A5′**, which would require that we substitute or cancel F18 based on comparable factors in the factor hierarchy. Indeed, it is arguable that *F18 Identical-Products* holds in both cases, but was too obvious to be explicitly mentioned in *Mason*. Alternatively, F18 may be seen as providing too weak a factor to distinguish the cases. Moreover, **Mason (Bryce) A6** may rest on resolving the relative strength of F4 and F6, if that becomes an issue. In all three instances, we would need to argue about factors themselves, which is the subject of the next section.

4. FURTHER ARGUMENT SCHEMES FOR REASONING ABOUT FACTORS AND DIMENSIONS

In section 3, we took as given that it is clear what factors of a case hold; in addition, the only way to moderate the effects of factors was with reference to the factor hierarchy and arguments of cancellation or substitution. However, there are additional considerations in reasoning about factors: as argued in [5], what factors hold of a case or which side favoured by a particular fact may be the whole point. In this section, we outline these considerations and present them in our ASPIC+ style analysis. In section 4.1, we outline the difference between factors and dimensions, as well as the role dimensions play in legal reasoning. In section 4.1, we review some of the main motivations to focus attention on reasoning about factors. Argumentation schemes for reasoning about what factors hold in a case relative to the facts of a case are introduced in section 4.2. Further schemes for reasoning about exclusory relations between factors are discussed in section 4.3. Finally, section 4.4 presents schemes for reasoning about factors along dimensions.

4.1 Dimensions in Legal Case Based Reasoning

Dimensions, rather than discrete factors, were used in Rissland and Ashley's HYPO [2], the system from which CATO was developed. Since factors as in CATO predominate in the literature [7, 16, 6], some background discussion on and justification for dimensions is warranted. Dimensions have an *extent* and values along the extent. In contrast to factors, which are either simply present or absent, a dimension, if present, may favour the plaintiff or defendant *to a particular degree*. Dimensions encompass a range of values, with the extreme pro-plaintiff value at one end and the extreme pro-defendant value at the other. Thus, at some unspecified point along the range the dimension will cease to favour the plaintiff and start to favour the defendant. Dimensions and factors are, however, related.

In one relationship, while a dimension may be continuous, factors are intervals along the dimension and ordered with respect to one another; in other words as in [6], factors can be taken as the *values* positioned along a dimension. For example, one dimension in HYPO is *Secrets-Voluntarily-Disclosed*, and ranges from 0 to 10,000,000 disclosees, 0 being the pro-plaintiff direction. In CATO, this dimension is expressed as factors that are ranked in strength. There is a pro-defendant factor *Secrets-Disclosed- Outsiders*, which is present if any disclosure at all had been made, effectively stating that the dimension favours the defendant rather than the plaintiff if a single person is disclosed to, and after that no further force is given to the defendant if there are a million disclosures. In this respect it is a relatively weak factor for the defendant. In addition, there is a *Disclosure-In-Public-Forum* factor, which is intended to cover extensive non-specific disclosure. This is a stronger factor for the defendant. If the latter, stronger, factor applies, then the former does not. Thus, we must reason not only with respect to the factors that hold of a case, but as well as with the *relative strength* of the factors one to the other. A number of HYPO dimensions are Boolean and counted as present only for one end of the range (e.g. *Common-Employee-Sole-Developer*), and these map straightforwardly to a single CATO factor.

Alternatively, HYPO dimensions and CATO factors are not related by a strength ordering relative to some measurable parameter. Most interesting is the HYPO dimension *Security-Measures-Adopted* which has a range (from pro-defendant to pro-plaintiff):

- Minimal measures, Access to premises controlled, Restrictions on Entry by Visitors, Restrictions on Entry by Employees, Product Marked Confidential, Employee Trade Secrets Program Exists, Restrictions on hardcopy release, Employee non-disclosure agreements.

In CATO, this translates into several factors without reference to relative strength or position along some continuous parameter: *No-Security-Measures*, *Security- Measures*, *Outsider-Disclosures- Restricted*, and *Agreed-Not-To-Disclose*. The increasing support for the plaintiff's cause is indicated by additional factors, in effect, a cumulative reading, rather than by an ordering of the factors according to strength, and which associates them with a point along a HYPO dimension.

Although factors dominated thinking about this style of reasoning in AI and Law for some time (e.g. [7, 16, 6]), the need for dimensions was argued for in [5]. Chief amongst the reasons was that the key issue of the case may be about where along the dimension a factor falls and, having situated it, whether the factor favours the plaintiff or defendant. The classic *Pierson v Post* is an example: the dispute turns on when pursuit can be counted as justifying possession, for which different degrees of progress towards bodily possession need to be recognised.[5] Contrast this with the representation based on factors in, for example [7], where the case is assigned the factor *caught*, and Post is then left without an argument.

We can further illustrate the issues using *Pierson v Post* as basis

[5] In brief the facts were these. Post was chasing a fox with horse and hounds and had cornered it when Pierson intervened and killed it with a fence pole. Post sued Pierson for taking his fox. On appeal, Pierson won on the grounds that only by mortally wounding or seizing the animal can one acquire possession of it, not simply by pursuing it.

for further reasoning about the factors and schemes that apply in cases. We will take Post as the plaintiff, as in the original action. In [7], the only factors present are *notCaught* and *OpenLand*, both of which are pro-defendant. Thus any case found for the defendant where the incident had taken place on open land and the plaintiff had not caught the animal would serve as a precedent; the plaintiff had nothing on which to base the plaintiff's case, all additional factors in the chosen precedent strengthening the defendant's case. In fact the argument put forward for the plaintiff was that Post was sufficiently close to, and sufficiently certain of, taking bodily possession of the fox that it should be *counted as* caught.

Essentially this is an argument against the presence of a factor favouring the defendant, and an argument in favour of the presence of a factor favouring the plaintiff. What this means in ASPIC+ terms is that the factors attributed to the case are not regarded as axioms, but require justification. What form might this justification take?

In *Pierson v Post*, the defendant's argument was in terms of particular authorities.[6] Tompkins, arguing for the defendant, cites Justinian, Fleca, and Bracton, all of whom seem to say that actual bodily possession is required, and Puttendorf and Barbeyrac, who seem to allow some latitude, but still require mortal wounding. Livingston, arguing for the plaintiff, claims that certain capture would also be enough for Barbeyrac, but also says that it should be so found in this case for the teleological purpose of encouraging the destruction of vermin. Neither of these lines of argument are case based or reasoning on the basis of the relationship of facts and factors per se, but make use of generic argumentation schemes such as *Argument from Authority* and *Sufficient Condition Scheme for Practical Reasoning* found in works such as [19].

4.2 Reasoning with Facts about Factors

There is, however, one argument scheme for case based reasoning from facts to factors which can be articulated. Suppose an undecided case concerning capturing a wild animal is being argued where the plaintiff claims that the animal *was caught on the basis of hot pursuit and inevitable capture*. Moreover, we take *Pierson v Post* as a *precedent*. Yet, though the argument was put forward in the precedent that it was as if the plaintiff caught the animal, this was not sustained, but rather the precedent establishes that the factor *not-Caught* applies to the benefit of the defendant. To argue thus, even if unsuccessfully, requires that cases are not only represented in terms of summarising factors (e.g. *not-Caught*) but also to the underlying *facts* which, in effect, support the factor (e.g. *hot pursuit* and *inevitable capture*). In addition, it indicates that we must reason about *schemes* for relationships between case facts and factors; that is, in *Pierson v Post*, it is argued that *hot pursuit* and *inevitable capture* do not imply *caught*, whereas in our hypothetical case, it is argued that they do.

In HYPO there are some procedures which determine whether a dimension applies in terms of facts stored about the case, in effect, providing schemes to reason about the factor category in the case. These schemes are not provided with any justification and no source is given, but are simply hard-coded into the dimension frames. We could therefore expect to have schemes relating factors to facts. To our previous notation, we add a binary relation *hasFact* between a case and a fact:

- *hasFact(case,fact)*; e.g. hasFact(piersonVPost, hotPursuit)

We also introduce a four-place rule relation:

- *ruleName(fact, factor, justifier, justification type)*, where fact is a *fact*, a justifier is some (or none) *judicial authority or case decision*, and the justification type is from *authority*, *definition*, or *contention*.

[6]For a detailed reconstruction of the arguments in this case see [12].

We have a sample of five interpretation rules, like those used by lawyers to ascribe factors to cases, which are discussed further below.

- **fRule1**(no-BodilyPossession, not-Caught, Justinian, authority)
- **fRule2**(mortallyWounded, caught, Puttendorf, authority)
- **fRule3**(noPursuit, not-Caught, none, definition)
- **fRule4**(hotPursuit, not-Caught, none, contention)
- **fRule5**(hotPursuit, caught, none, contention)

In ASPIC+, there are axioms that cannot be challenged and assumptions that can be challenged. **fRule3** is an axiom, true simply in virtue of the standard English meaning of the words. Other rules, however, need justification. In [15] three types of justification were suggested: authority, legal practice and precedent cases. Of the above **fRule1** and **fRule2** are justified by authorities, Justinian and Puttendorf respectively; if we accept Justinian (Puttendorf) as an authority we accept **fRule1** (**fRule2**). Rules **fRule4** and **fRule5** are not justified by authority or legal practice. At the time of *Pierson*, there were no precedent cases to provide justification, and the dispute was whether **fRule4** or **fRule5** should hold. Following the decision of *Pierson v Post*, we assume that **fRule4** holds, though Livingston argued for **fRule5**. This, however, remains an interpretation of the analyst: all talk of factors, and the attribution of factors to cases is with the analyst, not the judge. As **fRule4** is an assumption, it can, therefore, be challenged.[7]

The analyst would therefore need to record this interpretation in the knowledge base based on a family of argumentation schemes:

CS5 *Given ruleName(fact, factor, justifier, justification type), prec with pFactors(prec, P_p) and dFactors(prec, D_p), and curr with pFactors(curr, P_c) and dFactors(curr, D_c):*

$$\frac{\textit{hasFact(curr, fact), factor} \in P_p}{\textit{factor} \in P_c}$$

$$\frac{\textit{hasFact(curr, fact), factor} \in D_p}{\textit{factor} \in D_c}$$

Reasoning about facts, factor sets, and factors requires this additional level of representation, which is reflected in differences between HYPO and CATO.[8] HYPO stores case facts and calculates the applicability of dimensions on the basis of these. HYPO deterministically assigns dimensions of the basis of facts and does not support argument about them. CATO does not store case facts, and provides a descriptive rather than a computational characterisation of factors, relying on an analyst to assign them to cases; it does not support either argument or justification. We need make no further assumptions here about what knowledge is contained, but there may be a list of possible facts along particular dimensions as in, e.g.

[7]This is similar to the implicit nature of some discussions of values in legal cases [6].

[8]A reviewer notes that there are logical issues about the reification of propositions as "facts", "factors", and set constructs, for which we can presume standard AI solutions.

[5] and [6], or, perhaps, some more complicated ontology representing the domain. For arguments from authority and teleological arguments for the presence of a factor, we rely on the standard argument schemes. As noted above, there are some similarities with the validity rules of [15], but whereas there the validity in question was a legal rule, here it is the qualification of facts as factors used in rules, which are used by the analyst and are not themselves legal rules. Note that these arguments for and against the presence of factors conflict through rebuttal, and so preferences, among sources, purposes, precedents, or a combination of these, may be required to determine which argument is accepted, in the light of a specific audience [6].

4.3 Reasoning about Factor Incompatibility

In section 3.4, we provided rules for arguing about substituting or cancelling factors in relation to the factor hierarchy. In this section, we discuss other ways to reason about factors. In CATO, an alternative treatment of factors has that some pairs of factors are incompatible, so that the presence of one factor in a case provides an argument against the presence of another factor in that case. This is obvious in the case of clearly dichotomous factors such as *caught* and *notCaught*, but it is much more widespread than this in CATO. In CATO each factor has a textual explanation of when it does and does not apply. Often the latter includes circumstances where some other factor does apply. In [1] we have:

> **F20 InfoKnownToCompetitors (d)**
>
> **Description:** Plaintiff's information was known to competitors.
>
> This factor shows that plaintiff's information was known in the industry or available from sources outside plaintiff's business.
>
> **The factor applies if:** The information plaintiff claims as its trade secret is general knowledge in the industry or trade.
>
> **The factor does not apply if:** Competitor's knowledge of plaintiff's information results solely from disclosures made by plaintiff. (In this situation, F10 applies.) Or if the information could be compiled from publicly available sources, but there was no evidence that competitors had actually done so. (In this situation, F24 applies.)

F10 is *Secrets-Disclosed-Outsiders* and *F24 is Info- Obtainable-Elsewhere*. Thus F10, and F24 are incompatible with F20, and with one another.

To express the relationship of exclusion, we need an additional predicate:

- *excludes(factor1,factor2)*

Note that *excludes* is a symmetric relation, so that *excludes(factor1, factor2)* if and only if *excludes(factor2, factor1)*. Now if one side argues that F24 is present, but the other believes that F20 more appropriately describes the facts, it is important to ensure that both are not taken as present. This gives rise to an additional family of argumentation schemes:

CS6 *Given curr with pfactors(curr,P) and dfactors(curr, D):*

$$\frac{p2 \in P,\ excludes(p2,p)}{p \notin P}$$

$$\frac{p2 \in P,\ excludes(p2,d)}{d \notin P}$$

$$\frac{d2 \in D,\ excludes(d2,p)}{p \notin P}$$

$$\frac{d2 \in D,\ excludes(d2,d)}{d \notin P}$$

There are other argumentation schemes which can be used to establish that a factor is or is not in a case, such as those based on authority or purpose. Also note that if there is an argument based on $CS6$ for *factor1* excluding *factor2*, there will also be a rebutting argument based on $CS6$ for *factor2* excluding *factor1*. Which will be accepted will depend on which is supported by the stronger arguments.

In section 4.1, we recognised that factors may favour their party to different extents. In light of this, we need to reconsider our notions of cancellation and substitution. Arguments based on substitution and cancellation were used to undercut arguments distinguishing cases. Given that undercutters always win, these are powerful arguments. But suppose the factors in question were F20 and F24, as defined above. It is clear from the description that F20 is intended to be more pro-defendant than F24, for F20 represents an actual rather than a merely possible state of affairs. Thus if we have F24 in a *curr* and F20 in a *prec*, there is no problem in substitution: when considering the two under the common abstract factor the plaintiff's case is stronger, because the factor for the defendant is weaker in the *curr*. But if plaintiff attempts to argue that F24 in a *prec* substitutes for F20 in a *curr*, undercutting an instance of $U1.1$, the issue is less clear. The defendant can at least argue that F24 is not strong enough to be substituted for F20.

This has been handled in different ways in different applications: CATO indicated different degrees of influence by distinguishing *thin* and *fat* links in the factor hierarchy. IBP [8], which developed from CATO, introduced the idea of knock-out factors, which could be neither substituted for nor cancelled - indeed were entirely decisive with regards to a particular issue. The most quantitative approach can be found in [10], in which a limited number of dimensions, essentially corresponding to IBP issues, were each considered to have twenty slots, ten pro-plaintiff and ten pro-defendant, and every factor was assigned one of these slots. This enabled the difference in importance as well as the ordering to be considered. The positions were then mapped into weights in several ways, so that differences in the relative importance of the dimensions could also be considered. These issues require us to reason with dimensions, which is the topic of the next section.

4.4 Reasoning along Dimensions

Turning to dimensions, we want to introduce arguments about *substitution* and *cancellation*. As substitution and cancellation were defined using strict rules and always held of factors sharing the same abstract factor, we cannot rebut or undercut them.[9] While we might recast the original rules as defeasible, we prefer to leave the language established originally untouched as far as possible, instead proposing undercutters of arguments $U1.1.1$, $U1.1.2$, $U2.1.1$ and $U2.1.2$.

[9] We refer to the *rules* in section 3.3, not the *arguments* in U1.1.1-2, or U2.1.1-2.

First we introduce some additional language to relate to dimensions. We have a predicate *hasDimension* which takes three arguments: the dimension name, the factor, and a number between ten and minus ten representing the position of that factor on the dimension (see [10]). 0 indicates that the point is in the centre of the dimension, and so could be used to indicate the absence of any factor on the dimension, while ten is the extreme pro-plaintiff point, and minus ten is the extreme pro-defendant point.

- *hasDimension(dimension, factor, position),*
  e.g. dimension(outsideDisclosure, F20, -6) and
  dimension(outsideDisclosure, F24, -4)

Using the *hasDimension* predicate we now introduce the notion of one factor being stronger than another. For cancellation, we are interested in comparing the pro-plaintiff degree of a factor with the pro-defendant degree of another factor. Therefore we will be interested in the *absolute* value of the positions along the dimension.

- *stronger(factor1, factor2, degree),*
  where dimension(d, factor1, pos1), dimension(d, factor2, pos2), degree = |pos1| - |pos2|, and degree > 0.

Now we can undercut arguments made using $U1.1.1$, $U1.1.2$, $U2.1.1$ and $U2.1.2$.

U7 *Let A be an argument instantiating $U1.1.1$ or $U2.1.1$ with claim that factor can be substituted for factor2, or an argument instantiating $U1.1.2$ or $U2.1.2$ claiming that factor cancels factor2:*

$$\frac{\textit{stronger(factor2,factor,degree)}}{\overline{A}}$$

This, however, means that any amount of additional strength, however small, is sufficient to prevent substitution and cancellation: that is, only *a fortiori* arguments will succeed. This rather conservative point of view maybe what we want, but we may equally wish to allow the possibility of a small difference being discounted. We therefore need a notion of a significant difference. We design the predicate to allow that this may vary according to the dimension under consideration.

- *significantDifference(dimension, threshold)*

This allows for an undercutter to arguments made using $U7$.

U7.1 *Let A be an argument instantiating $U7$, with premise stronger(factor2, factor, degree) and dimension(factor, dim, pos1) and dimension(factor2, dim, pos2), and significantDifference(dim, threshold):*

$$\frac{\textit{degree < threshold}}{\overline{A}}$$

Where do the thresholds come from? In most cases they will need to be simply assumed or be determined by some feel for legal practice or even rely on personal preference [13]. For example, we can try to rescue *Bryce* as a precedent for *Mason* by setting the threshold above the degree of F18, so undercutting $A5'$. Disagreement here is quite natural. It would be possible to use a case based argument to set bounds on the threshold, provided one could find a pair of precedent cases in which, using the argument schemes defined here, it was possible to show that a substitution for, or cancellation of, a stronger factor had been made successfully. This is quite a complex argument, and we will not attempt to reduce it to an argumentation scheme here, but rather take the threshold as given.

5. DISCUSSION

By articulating the process of reasoning with precedent cases in this way, we can see it as a sequence of stages in a dialogue between **P**laintiff and **D**efendant, which are as follows, where every option is available:

1. **P**: Assert that the decision should be in favour of the plaintiff since factors favouring the plaintiff are preferred to factors favouring the defendant;
   (a) **D:** Cite additional points in favour of the defendant
   (b) **P:** Substitute for, dismiss, or cancel these additional strengths
   (c) **D:** Dispute strength of substituting or cancelling factors
2. **P:** Identify a precedent case which justifies a preference applicable to the current case;
   (a) **D:** Cite additional points in favouring the plaintiff in the precedent
   (b) **P:** Substitute for, dismiss, or cancel these additional strengths
   (c) **D:** Dispute strength of substituting or cancelling factors
   (d) **D:** Identify a precedent case which justifies a preference for the defendant applicable to the current case
      i. **P:** Cite additional points in favouring the defendant in the precedent:
      ii. **D:** Substitute for, dismiss, or cancel these additional strengths
      iii. **P:** Dispute strength of substituting or cancelling factors
3. **D:** Dispute which factors are present in the current case
   (a) **P:** Defend original factors

Different systems will support more or fewer of these stages. At one extreme we have a neural network style system such as that described in [3] in which the system acts as a black box taking factors (or facts) as an input and expressing a preference based on its internalisation of the set of precedents. Such a system supports only step 1. CATO, from which our discussion began, supports the identification of the preference in 2, the distinguishing moves in 1a and 2a (although it does not discriminate between them), and the counter example move of 2d. CATO also supports the downplaying of 1b and 2b, but does not distinguish between substitution and cancellation. HYPO links facts and dimensions, and so can explain 3, but not support argument about it. Hypothetical arguments in HYPO were intended to explore the issues raised in 1c, 2c, and 2d(iii), but this aspect of HYPO was never fully developed in [2]. These considerations are also used internally in the most advanced version of Chorley's AGATHA [9], although the resulting arguments are not transparent to the user.

Our analysis identifies the knowledge base required by each stage. Given such a KB, the specification of the argumentation schemes in this paper, suitably fully formalised in ASPIC+, would permit straightforward implementation, using a defeasible reasoner to instantiate the schemes from the KB, identifying the attack relations, and then evaluating them as in a Dungian argumentation framework.

Another benefit which will result from representing these arguments in terms of ASPIC+ will be that we can regard cases described under factors as but one source of arguments. At the top level, stage 1, there may be arguments for the defendant rebutting our case based argument for the plaintiff and these arguments may be based on cases, authority, purpose, or whatever other kind of argument our opponent wishes to advance. Similarly, the premises of our arguments often require other, generic, argument schemes, such as authority and purpose, to justify them. By providing a framework in which all kinds of argument can be represented equally, we can readily provide a framework in which reasoning of many different kinds can be deployed. Note that this is done without recasting the various distinctive case based aspects of CATO style arguments uniformly as ordinary rules, as was the case in e.g. [16].

A final important insight is gained by recognising that the above indicates at which points choice is possible, and at which points the judgement is constrained. Let us relate this to the steps above. At step 1 we may get arguments, constructed with a variety of schemes, for and against deciding for the plaintiff, which conflict through rebuttal and so can be decided through preferences. The attack of 1a, however, cannot be rejected on the grounds of preference, but can only be defeated by 1b, which in turn can only be defeated by an argument from 1c. Arguments in stage 1c itself, however, may be resolved on grounds of preference. Similarly although the rebuttals arising at 2d may be decided by preferences, 2a can only be defended by 2b, and 2b by 2c, at which stage preferences may be used to resolve competing arguments. When considering 2d, only at 2d(iii) do preferences play a role. Thus although we may think of case based reasoning as involving a choice between the plaintiff and the defendant arguments, in fact, choice operates at a number of quite specific, fine-grained points in the debate.

6. CONCLUSION

In this paper, we have clarified a range of aspects of legal case-based reasoning with factors using semi-formal defeasible arguments. In future work, we look to fully formalise these schemes in ASPIC+, extend this approach, integrate further aspects and examples of legal case based reasoning such as the issues of IBP and values of [6]. In particular, we will consider how to argue comparatively about precedents to find the most on-point cases using the claim lattice of HYPO and CATO, which we would reconstruct as a tree of arguments in attack relations.

7. ACKNOWLEDGMENTS

This work was partially supported by the European IMPACT Project (***247228***). The views expressed are, however, those of one or more of the authors, and should not be taken as necessarily representative of the project. The authors are grateful for the comments of one of the reviewers, though we could not address them all.

8. REFERENCES

[1] V. Aleven. *Teaching case-based argumentation through a model and examples*. PhD thesis, University of Pittsburgh, Pittsburgh, PA, USA, 1997.

[2] K. Ashley. *Modelling Legal Argument: Reasoning with Cases and Hypotheticals*. Bradford Books/MIT Press, Cambridge, MA, 1990.

[3] T. Bench-Capon. Neural networks and open texture. In *ICAIL 1993*, pages 292–297, 1993.

[4] T. Bench-Capon. Arguing with cases. In A. Oskamp et al., editors, *JURIX 1997*, pages 85–100, Nijmegen, 1997. Gerard Noodt Instituut.

[5] T. Bench-Capon and E. Rissland. Back to the future: dimensions revisited. In B. Verheij and et al., editors, *JURIX 2001*, pages 41–52, Amsterdman, The Netherlands, 2001. IOS Press.

[6] T. Bench-Capon and G. Sartor. A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence*, 150(1-2):97–143, 2003.

[7] D. Berman and C. Hafner. Representing teleological structure in case-based legal reasoning: the missing link. In *ICAIL 1993*, pages 50–59. ACM Press, 1993.

[8] S. Brüninghaus and K. Ashley. Predicting the outcome of case-based legal arguments. In *ICAIL 2003*, pages 233–242, Edinburgh, United Kingdom, 2003. ACM Press: New York, NY.

[9] A. Chorley and T. Bench-Capon. AGATHA: Using heuristic search to automate the construction of case law theories. *Artifical Intelligence and Law*, 13(1):9–51, 2005.

[10] A. Chorley and T. Bench-Capon. An empirical investigation of reasoning with legal cases through theory construction and application. *AI and Law*, 13(3-4):323–371, 2005.

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

[12] T. Gordon and D. Walton. Pierson v. Post revisited - a reconstruction using the Carneades argumentation framework. In *COMMA 2006*, pages 208–219, 2006.

[13] M. Gramair, T. Gordon, and D. Walton. Probablistic semantics for the Carneades argument model using Bayesian networks. In P. Baroni and et al., editors, *Computational Models of Argument: Proceedings of COMMA 2010*, pages 255–266. IOS Press, 2010.

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

[15] H. Prakken. Reconstructing Popov v. Hayashi in a framework for argumentation with structured arguments and Dungean semantics. *Knowledge Engineering Review*, In Press.

[16] H. Prakken and G. Sartor. Modelling reasoning with precedents in a formal dialogue game. *AI and Law*, 6(2-4):231–287, 1998.

[17] E. Rissland, D. Skalak, and M. Friedman. BankXX: Supporting legal arguments through heuristic retrieval. *AI and Law*, 4(1):1–71, 1996.

[18] D. Skalak and E. Rissland. Arguments and cases: an inevitable intertwining. *AI and Law*, 1(1):3–44, 1992.

[19] D. Walton. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.

[20] A. Wyner and T. Bench-Capon. Argument schemes for legal case-based reasoning. In A. R. Lodder and L. Mommers, editors, *JURIX 2007*, pages 139–149, Amsterdam, 2007. IOS Press.