Argument Schemes for Reasoning with Legal Cases Using Values

Trevor Bench-Capon
Department of Computer Science
University of Liverpool
England
tbc@csc.liv.ac.uk

Henry Prakken
Department of Information and Computing Sciences, Utrecht University
Faculty of Law, University of Groningen
The Netherlands
H.Prakken@uu.nl

Adam Wyner
Department of Computer Science
University of Aberdeen
Scotland
azwyner@abdn.ac.uk

Katie Atkinson
Department of Computer Science
University of Liverpool
England
K.M.Atkinson@liverpool.ac.uk

ABSTRACT

Argument schemes can provide a means of explicitly describing reasoning methods in a form that lends itself to computation. The reasoning required to distinguish cases in the manner of CATO has been previously captured as a set of argument schemes. Here we present argument schemes that encapsulate another way of reasoning with cases: using preferences between social values revealed in past decisions to decide cases which have no exact matching precedents when the cases are described in terms of factors. We provide a set of schemes, with variations to capture different ways of comparing sets and varying degrees of promotion of values; we formalise these schemes; and we illustrate them with some examples.

1. INTRODUCTION

One fruitful way of using argument schemes is to provide a set of schemes to model a particular reasoning method. Examples are hypothetical reasoning [8], democratic deliberation [9] and case based reasoning [29], [30] and [25]. The schemes for reasoning with cases given in the last three papers, however, are primarily intended to capture only one aspect of reasoning with legal cases: distinguishing precedent cases modelled as sets of factors as modelled in the CATO system of Aleven and Ashley [1]. In this paper we will use argumentation schemes to explore another method of reasoning with legal cases: the use of social values to broaden the applicability of precedents, as developed in the work of Bench-Capon and Sartor [10], building on the notion of teleological reasoning with cases introduced to AI and Law in [11]. The key idea is that the reason why a factor favours a party is that deciding for that party when the factor is present promotes a social value. Thus preferences between factors can be explained in terms of preferences between their associated values. Having established the preferences between values, those preferences can then be used to determine the decision in cases containing different factors relating to those values, so enable us to go beyond simple *a fortiori* reasoning. It is this reasoning move based on values that we will capture as a set of argumentation schemes in this paper.

The point of using argument schemes, normally seen as practical stereotypical patterns of reasoning justifying presumptive conclusions, as described by Walton [28], is that we are able to make the reasoning method explicit in a way which is also readily computable. Effectively the argument schemes become the expert heuristics used in traditional knowledge based system developments, and so the identification of the schemes is effectively also a way of doing knowledge elicitation. Once available, the schemes can be interpreted declaratively or procedurally to provide either an explicit understanding of the reasoning method, or a computational means of reasoning with the method. Moreover, consideration of the ways in which arguments made using the schemes can be attacked (every scheme is associated with a set of characteristic "critical questions" in [28] which represent ways of challenging the presumptive conclusion) allows the process to be developed further, so that as well as providing justification for presumptive claims, attacks on these presumptions can be made and defended against. This enables a justification of the claim to be given in terms of the claim emerging from the appropriate decision making process as in [17]. Finally consideration of the terms used in expressing the scheme determines the items that need to be represented in the knowledge base to support the relevant reasoning.

We formally represent the schemes in terms of ASPIC+ [23, 20], following the method of [25] to reconstruct distinguishing cases as in CATO. The point of formalising argument schemes in ASPIC+ is twofold. First, ASPIC+ allows for full formalisation of argument schemes, thus allowing them to be disambiguated and their role in argumentation to be precisely defined. Second, results on the metatheory of ASPIC+ can be used to investigate consistency and closure properties of our formal specification.

The remainder of the paper is organised as follows. Section 2 will describe the reasoning of [10] and represent this reason-

ICAIL '13 June 10-14 2013 Rome, Italy

ing informally as a set of argument schemes using a very simple mechanism for comparison between values. Section 3 will discuss some more sophisticated ways in which sets of values can be compared. Section 4 will supply the necessary formal background on ASPIC+, and Section 5 will give a formal presentation of the informal schemes. Section 6 will discuss some further points of interest that arise from this work. Section 7 concludes with some remarks.

We believe that the contributions of this paper are threefold. Conceptually we believe it clarifies and furthers the understanding of the role of purpose and values in reasoning with legal cases, a topic which has been important since its introduction in [11]. Theoretically, the use of ASPIC+ gives a precise definition of the reasoning method, allowing various alternatives to be specified and formal properties to be investigated. Finally, the formal specification of a knowledge base and inference rules can be used as the basis for an implementation (cf the discussion in [5] of the implementation of the schemes of [29]).

2. ARGUMENT SCHEMES USING VALUES

In [10] cases are represented as pairs comprising a set of factors and an outcome, either *plaintiff* or *defendant*. Factors are represented as triples comprising the name of the factor, the party (plaintiff or defendant) favoured by the presence of the factor in a case, and the value promoted by deciding the case for that party when that factor is present. The idea is to construct a *theory* from this background knowledge: simple rules can be introduced by taking a factor <*factor,party,value*> and adding a rule *if factor then party*. Complex rules can be formed by merging the antecedents of simple rules with the same consequents. Preferences between rules and between sets of values are added to the theory based on the factors, values and outcomes of previous cases. The idea is that a current case is decided according to the best (typically measured in terms of explanatory power and simplicity) theory that can be constructed.

The translation of factors to rules can be contrasted with that of [24]. There a case translates into three rules: one with *all* the pro-plaintiff factors as antecedent and plaintiff as consequent, one with *all* the pro-defendant factors as antecedent and defendant as consequent, and one expressing a priority between them. By including all the factors, [24] forms the weakest rules supported by the case, and so can be considered to reason conservatively, not going beyond what is definitely confirmed by the precedent. In [10] the rules are not tied to cases, and represent the strongest possible rule supported by the factor: these rules will be weakened by including more factors in the antecedent as dictated by the desire to cover more cases as the theory is constructed. A middle position can be found in [18], where each precedent case gives rise to three rules: for the losing side, the rule is the same as [24]; but for the winning side, only a subset of the factors is used; and precedent yields a priority between these two rules. This enables the import of the precedent to go beyond *a fortiori* reasoning.

To represent the reasoning of [10] as argumentation schemes, we will need three schemes: one to establish a value preference from a precedent case, one to apply a value preference to a new case, and one to establish that a value is promoted by deciding a case for a particular party when a given factor is present. An important decision to make is therefore how to view these value preferences. The theories of [10] contain preferences between *sets* of values, and so require a basis of comparison for sets of values. One approach is simply to compare sets on their most preferred distinct member. This was adopted effectively in, for example, [6], where only the strongest argument (which may vary for different audiences) was used to justify an action. A key advantage of this approach, is that the strength of a set becomes determined by a single member, and so no arithmetic is involved: all that is needed is that the values be ordered. While this offers a simple and effective way of comparing sets of values, alternatives have been proposed and used, and are probably required if the preferences are to cover a substantial body of case law [15]. In order to keep matters as simple as possible initially, we will begin by presenting the schemes informally and suppose that the strength of a set is given by its strongest element.

We represent background cases and factors as in [10].

2.1 Decision Based on Value Scheme

Our first scheme is to apply the value preference to the current case. We need a premise that a value is promoted by deciding for one party, a premise that some other value is promoted by deciding for the other party, and a premise that the first value is preferred to the second, So:

VAS1: Decision Based on Value Scheme

Promotion Premise 1: Decision for party 1 in current case promotes value1

Promotion Premise 2: Decision for party 2 in current case promotes value2

Preference Premise: Value1 is preferred to value 2

Conclusion: Decide current case for party 1

The first two premises need to be established using a second scheme, and the third premise requires a third scheme. The third premise also requires that the values be distinct. Note that the scheme assumes that the value for the losing side is the best value for party 2: the value in the first premise need not be the best value for party 1, provided that it is better than value 2. Because different audiences may rank values differently, whether this assumption is satisfied will depend on the audience. We address this issue through VAS4 below.

2.2 Promotion Scheme

The Promotion scheme VAS2 is used to establish the two promotion premises of VAS1. VAS2 requires two premises, both of which can be established on the basis of information available in the case and factor background.

VAS2: Promotion Scheme

Factor Premise: Factor is present in case

Value Premise: Decision for party when factor is present promotes value

Conclusion: Decision for party in case promotes value

2.3 Preference from Case Scheme

For the scheme to establish a preference between values we need four premises: one to identify a value promoted by deciding the case for a particular party, one to identify a different value promoted by deciding the precedent for the other party (the difference being explicitly required by the fourth premise), and one to state that the precedent was decided for the first party.

VAS3: Value Preference from Case Scheme

Decision Premise 1: Decision for party 1 in precedent case promotes value1

Decision Premise 2: Decision for party 2 in precedent case promotes value2

Precedent Outcome Premise: Precedent case was decided for party 1

Distinct Value Premise: value 1 and value 2 are distinct

Conclusion: Value1 is preferred to value 2

The first two premises can be justified using VAS2, while the third is available directly from the background. Like VAS1 an assumption is needed: here the assumption is that the value for party 1 used in the first decision premise is the strongest available for that party in that precedent case, since otherwise it might be a different value that explained the victory. Since, however, that value must be able to defeat all the values for the other party, any value in the precedent can appear in the second decision premise.

2.4 Better Value Scheme

The assumptions about using the best values in the second promotion premise of VAS1 and the first decision premise of VAS3 mean that it is possible to attack arguments made using these schemes by questioning the assumptions required by these premises. Thus we have an undercutter for VAS1 and VAS3:

VAS4: Better Value Scheme

Decision Premise 1: Decision for party in case promotes value1

Decision Premise 2: Decision for party in case promotes value2

Preference Premise: Value 2 is preferred to value 1

Conclusion: Argument made with VAS1/VAS3 is not applicable

Note that all three of the premises in VAS4 can be justified using the schemes already presented: the two decision premises using VAS2 and the preference premise using VAS3. In this way a number of precedents may be needed to form the complete debate.

2.5 Example

We can illustrate the above scheme with the wild animals cases introduced in [11] and used as the leading example in [10]. This example is simple enough for the single value method of comparison to be appropriate, and has been often used in discussions of reasoning with values, allowing comparison with previous work.

The wild animal cases of [11] are: *Pierson v Post*, where Post was chasing a fox, which Pierson killed and carried off; *Keeble v Hickeringill*, where Keeble was a commercial duck hunter and Hickeringill maliciously scared the ducks away from Keeble's pond; and *Young v Hitchens* in which Young and Hitchens were both commercial fishermen. As Young was closing his nets to complete his catch, Hitchens intervened and scooped up the concentration of fish. Winners were Pierson, Keeble and Hitchens.

We use factors and values taken from [10]. All the cases contain *noposs*, since in none of them did the plaintiff get physical possession of the animal. *Keeble* and *Young* contain *pliv*, since the plaintiff was trying to earn a living. Young also contains *dliv*, since Hitchens was also acting commercially. The value of *llit* (less litigation, legal clarity) is promoted by deciding for the defendant when there is no physical possession, since otherwise there is no clear criterion for establishing possession. The value *econ* (economic worth) is promoted by deciding for a party engaged in earning their livelihood. We ignore factors such as the status of the land and the social benefits of fox hunting, which are not needed for the basic example. The case and factor background is therefore: *case(Pierson,[noposs]), case(Keeble, [noposs, pliv]), case(Young, [noposs, pliv, dliv]), factor(noposs, defendant, llit), factor(pliv, plaintiff, econ), factor(dliv, defendant, econ).*

We suppose that *Young* is the current case. Young begins with an instantiation of VAS1: *Deciding for the plaintiff promotes econ; deciding for the defendant promotes llit. Econ is preferred to llit.*

Young then uses two instantiations of VAS2: *Pliv is in Young and deciding for the plaintiff when pliv is present promotes econ; noposs is in Young and deciding for the defendant when noposs is present promotes llit.*

Young then cites Keeble using VAS3: *Deciding for the plaintiff in Keeble promotes econ; deciding for the defendant in Keeble promotes llit. Keeble was decided for the plaintiff. Econ and llit are distinct. Econ is preferred to Llit.*

But Young does not win: Hitchens can use VAS4: *Deciding for the defendant promotes llit, Deciding for the defendant promotes econ. Econ is preferred to llit.*

Two of the premises are already established: the promotion of *econ* on the basis of *dliv* can be established using VAS2. Having defeated Young's argument based on VAS1, Hitchens can now propose his own argument: *Deciding for the defendant promotes llit; deciding for the defendant promotes no value. Llit is preferred to no value.*

Thus a decision for Young promotes {*econ*}, but a decision for Hitchens promotes {*econ*, *llit*}. Although *Keeble* establishes *econ* as the more important value, this is present in both sets and so cancels, leaving *llit* as decisive, so that Hitchens wins.

3. COMPARING SETS OF VALUES

The schemes given thus far make an important assumption about how sets of values are compared: that is, sets of values are compared by ignoring values common to both sets and then using the best remaining value in each set, so that only a single value from each set needs to be considered when comparing two sets. This is a possible and not unreasonable approach, corresponding roughly to the view that laws are designed to serve one overriding purpose and that other values are subordinate to this. For example, Trade Secrets Law might be designed with the chief aim either to foster innovation or to promote competition in production. This position may not, however, be appropriate in every case: for example it may be that Trade Secrets Law is intended to serve or balance both these purposes. We must therefore examine alternative ways of comparing sets of values.

3.1 Audiences

Values were originally motivated by a desire to provide a computational realisation of Perelman's notion of *audience* [22]. Perelman stressed that arguments needed to be evaluated by reference to an audience - an argument is persuasive only to the extent that those to whom it is addressed are persuaded. Audiences are particularly important when elements of subjective aspiration, interest and purpose affect the acceptability of arguments, as is invariably the case in domains such as law and politics. In [3] a way of introducing audiences to abstract argumentation was proposed. The idea was that arguments can be related to a (relatively small) set of values, namely the social values that would be promoted by accepting the arguments, and that audiences can then be characterised by a (total) preference ordering on these values. In this way variance in legal decisions over time and jurisdiction can be explained (cf Marshall's remark in *Furman v Georgia* that "*stare decisis* must give way to changing values"). In the context of [3], where arguments relate to a single value, and comparison is always value-value to adjudicate between two arguments, a simple ordering on values is all that is needed. Applying this general idea to reasoning with legal cases, however, as in [10], means that comparison between sets of values is essential, since deciding a case for a particular party will typically promote several values, since several factors favouring the party will be present.

Here we suppose that preferences between values, and hence the audience, can only be determined by the precedent cases cited. In some approaches, e.g. [7] the audience can be constrained by other independent arguments, based, for example, on legal principles, or a mixture of precedent cases and other arguments. In such cases we would have ways other than **VAS3** of establishing preferences between values, but as our aim is to capture the reasoning of [10], we will not explore these further here.

3.2 Strengths of Values

Now the relationship between these values is crucial for set comparison. Suppose for example that

> For all values V, W, if there exists a $v \in V$ such that for all $w \in W$ it holds that v is preferred to w, then V is preferred to W.

If this holds then the comparison method we have used so far is appropriate, since if a value is preferred to a set of values, then *a fortiori*, any set containing it is preferred to that set. Effectively this means that the gap between the values in the ordering is large and that the preference between values is very strong. Conversely if the preference between values is very slight, so that if, for example, two less important values are always jointly preferred to a single more important one, then the preference between sets can be determined simply by cardinality, that is counting the elements in the sets, using the preference order only as a tie-breaker when the cardinalities are the same. This approach has an appealing simplicity. It is, however, more elegant to generalise this to an arbitrary preference ordering on sets of values, similar to the ordering on sets of desires in [9], which itself built on the work of [13]. Then one way to define this ordering is counting, another is lexicographically, and so on. We will adopt this more general approach when we come to formalise the schemes in section 5.2.

3.3 Comparing Sets of Values

So far we have assumed that

- difference in strength between adjacent values in the value order is uniform;
- every factor promotes its value to the same degree.

Neither of these assumptions is obviously correct: for example in [21] the gap between variables is allowed to vary, and in particular values are divided into a group of essential values separated by a relatively small gap, and a group of desirable values, also separated by a relatively small gap, but with a large gap between the two groups. That was appropriate for the application in [21], but different applications - even different areas of law - are likely to have different needs, and so we should be able to accommodate various comparison methods. The second assumption is also questionable. Indeed if we compare the factors of CATO with the dimensions of HYPO [2] we see that several factors represent different points on the *same* dimensions. More importantly, several factors, such as *disclosure to outsider*s and *disclosure in a public forum* represent points favouring the same party to differing degrees. This strongly suggests that CATO factors promote their values to differing degrees, and so we should address this point.

3.4 Different Degrees of Promotion

In CATO itself different degrees of promotion are indicated in the factor hierarchy by the thin and thick lines linking factors to their parents [1]. This might suggest that we could add this information to the factor background by extending the triples of [10] to 4-tuples: <*factor,party,value,degree*>, where degree may be either *ordinary* ('thick') or *weak* ('thin'). Now we have the additional critical question to pose against VAS2: *is the value only weakly promoted by factor?* The objection can be met either by showing that the factor is recorded as a 'thick' factor in the background, or by showing that the value is additionally promoted by some other factor. If this other factor is a 'thick' factor, that will suffice to meet the objection, but if this additional factor is also 'thin', we might require more than one such additional factor to overcome the objection. This approach broadly retains the idea that a set is as strong as its strongest member. It is, of course, still qualitative and so does not support definite comparison of sets, but this is to be expected as CATO was not intended to predict the outcomes of cases, but merely to identify the arguments available to both parties.

Two other approaches to reasoning with legal cases should be briefly mentioned although they do not strictly pertain to arguing with cases on the basis of values. Both are intended to support outcomes, not simply to identify arguments. Issue Based Prediction (IBP) [14] introduced the notion of *Knockout (KO) Factors* which if present serve to decide the case for their side without further consideration. These could be interpreted as promoting a value to a sufficient degree to override any other value promotion, but it is perhaps better to see them as providing an argument separate from values: thus if a case contains a KO factor there will be an additional argument for the favoured side. Chorley [15] uses values differently: preferences between values are represented as weights, and factors are given weights to reflect the degrees to which they promote values, positive if they favour the plaintiff and negative if the favour the defendant. The product of the weights gives a contribution for each factor and these contributions are summed. If the result is positive the plaintiff should win, if negative, the defendant. Thus the role of values is only to give a formula which is then applied to cases to give an outcome, and there is no explicit argumentation. In [15] all the work is done in the analysis which leads to the chosen weights, and the program presents a calculation without showing any of the underlying reasoning that connects the analysis to the precedents, which is the main reason for using argumentation. This issue is discussed further in section 6.1.

4. FORMAL BACKGROUND

We first briefly summarise the formal frameworks used in this paper. An *abstract argument framework*, as introduced by Dung, [16] is a pair $AF = \langle \mathcal{A}, \textit{defeat} \rangle$, 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. A general framework for giving structure to arguments is the ASPIC framework, most fully defined as ASPIC+ in [23, 20]. The ASPIC+ framework first defines the notion of an *argumentation system*, which consists of a logical language $\mathcal{L}$ with a binary contrariness relation $^{-}$ and two sets of inference rules $\mathcal{R}_s$ and $\mathcal{R}_d$ of *strict* and *defeasible inference rules* defined over $\mathcal{L}$, written as $\varphi_1, \ldots, \varphi_n \rightarrow \varphi$ and $\varphi_1, \ldots, \varphi_n \Rightarrow \varphi$. Informally, that an inference rule is strict means that if its antecedents are accepted, then its consequent must be accepted *no matter what*, while that an inference rule is defeasible means that if its antecedents are accepted, then its consequent must be accepted *if there are no good reasons not to accept it*.

In the present paper we use an argumentation system in which $\mathcal{L}$ is a first-order language with equality further specified in the coming sections, its contrariness relation corresponds to classical nega-

$$\cfrac{\cfrac{P1 \quad P2}{P3}\ (A1) \qquad P4}{C1}\ (A2)$$

Figure 1: An argument

tion, the strict rules $\mathcal{R}_s$ are all valid first-order inferences over $\mathcal{L}$ and the defeasible rules $\mathcal{R}_d$ are as specified in the coming sections.

Arguments are in ASPIC+ constructed from a knowledge base $\mathcal{K}$, which contains two disjoint kinds of formulas: the *axioms* $\mathcal{K}_n$ and the *ordinary premises* $\mathcal{K}_p$. The formal definition of an argument is as follows:

DEFINITION 4.1. *[Argument] An* argument *A on the basis of a knowledge base* $\mathcal{K}$ *in an argumentation system* $(\mathcal{L}, ^-, \mathcal{R}_s, \mathcal{R}_d)$ *is:*

1. φ *if* $\varphi \in \mathcal{K}$ *with:* $\texttt{Prem}(A) = \{\varphi\}$; $\texttt{Conc}(A) = \varphi$; $\texttt{Sub}(A) = \{\varphi\}$; $\texttt{TopRule}(A)$ = *undefined.*
2. $A_1, \dots A_n \rightarrow/\Rightarrow \psi$ *if* $A_1, \dots, A_n$ *are arguments such that there exists a strict or a defeasible rule* $\texttt{Conc}(A_1), \dots, \texttt{Conc}(A_n) \rightarrow/\Rightarrow \psi$ *in* $\mathcal{R}_s/\mathcal{R}_d$.
   $\texttt{Prem}(A) = \texttt{Prem}(A_1) \cup \dots \cup \texttt{Prem}(A_n)$; $\texttt{Conc}(A) = \psi$; $\texttt{Sub}(A) = \texttt{Sub}(A_1) \cup \dots \cup \texttt{Sub}(A_n) \cup \{A\}$;
   $\texttt{TopRule}(A) = \texttt{Conc}(A_1), \dots, \texttt{Conc}(A_n) \rightarrow/\Rightarrow \psi$.

An argument is *strict* if all its inference rules are strict and *defeasible* otherwise, and it is *firm* if all its premises are in $\mathcal{K}_n$ and *plausible* otherwise.

Arguments can be displayed as inference trees. An example argument, A_2, is shown in Figure 1. A_2 has premises P_1, P_2, P_4, and conclusion C_1. A single and double bar stand for, respectively, a strict and defeasible inference. Argument A_2 has four subarguments, namely A_1, which has premises P_1 and P_2 and conclusion P_3, and the formulas P_1, P_2 and P_4 as atomic subarguments.

An argumentation system and a knowledge base are combined with an *argument ordering* into an *argumentation theory*. The argument ordering could be defined in any way, for example, in terms of orderings on $\mathcal{R}_d$ and $\mathcal{K}_p$.

DEFINITION 4.2. *[Argumentation theories] An* argumentation theory *is a triple* $AT = (AS, \mathcal{K}, \preceq)$ *where AS is an argumentation system,* $\mathcal{K}$ *is a knowledge base in AS and* $\preceq$ *is a partial preorder on the set of all arguments on the basis of* $\mathcal{K}$ *in AS (below denoted by* $\mathcal{A}_{AT}$*).*

Arguments can be attacked in three ways: attacking a conclusion of a defeasible inference, attacking the defeasible inference itself, or attacking a premise. To define how a defeasible inference can be attacked, a function n is assumed that assigns to each element of $\mathcal{R}_d$ a well-formed formula in $\mathcal{L}$. Informally, $n(r)$ (where $r \in R_d$) means that r is applicable. For our argumentation system, ASPIC+'s definitions of attack can be simplified as follows:[1]

DEFINITION 4.3. *[attacks] A* attacks *B iff A* undercuts*,* rebuts *or* undermines *B, where:*

- *A* undercuts *argument B (on* B'*) iff* $\texttt{Conc}(A) = -n(r)$ *for some* $B' \in \texttt{Sub}(B)$ *such that* B'*'s top rule r is defeasible.*
- *A* rebuts *argument B (on* B'*) iff* $\texttt{Conc}(A) = -\varphi$ *for some* $B' \in \texttt{Sub}(B)$ *of the form* $B''_1, \dots, B''_n \Rightarrow \varphi$.
- *Argument A* undermines *B (on* φ*) iff* $\texttt{Conc}(A) = -\varphi$ *for some ordinary premise* φ *of B.*

In Figure 1, argument A_2 can only be rebutted or undercut on its defeasible subargument A_1.

Attacks combined with the preferences defined by an argument ordering yield three kinds of defeat.

DEFINITION 4.4. *[Successful rebuttal, undermining and defeat]*

- *A* successfully rebuts *B if A rebuts B on* B' *and* $A \not\prec B'$.
- *A* successfully undermines *B if A undermines B on* φ *and* $A \not\prec \varphi$.
- *A* defeats *B iff A undercuts or successfully rebuts or successfully undermines B.*

The success of rebutting and undermining attacks thus involves comparing the conflicting arguments at the points where they conflict. The definition of successful undermining exploits the fact that an argument premise is also a subargument. For undercutting attack no preferences are needed to make it succeed, since undercutters state exceptions to the rule they attack.

ASPIC+ thus defines a set of arguments with a binary relation of defeat, that is, it defines abstract argumentation frameworks in the sense of [16]. Formally:

DEFINITION 4.5. *[Argumentation framework] An* abstract argumentation framework *(AF)* corresponding to an argumentation theory *AT is a pair* $< \mathcal{A}$, Def$>$ *such that:*

- $\mathcal{A}$ *is the set* $\mathcal{A}_{AT}$ *as defined by Definition 4.1,*
- Def *is the relation on* $\mathcal{A}$ *given by Definition 4.4.*

Thus any semantics for abstract argumentation can be applied to ASPIC+.

5. FORMALISATION OF THE SCHEMES

The method for formalising the schemes of Section 2 follows the one of [25], which we here summarise as far as relevant.

We assume a many-sorted first-order language with sorts for parties, cases (with subsorts for current cases and precedents), factors and values. We trust that the types of the terms and predicate and function symbols will be clear from the context and wording. Constants begin with an upper case letter while variables start with a lower case letter.

Factors are in [25] declared to be either pro-plaintiff or pro-defendant by adding formulas of the following form to $\mathcal{K}_p$:

- $pFactor(factor)$, meaning that *factor* is a pro-plaintiff factor;
- $dFactor(factor)$, meaning that *factor* is a pro-defendant factor.

No factor can be both pro-plaintiff and pro-defendant, expressed by adding to $\mathcal{K}_n$:

1. $\forall factor \neg(pFactor(factor) \wedge dFactor(factor))$

For notational convenience we now add to $\mathcal{K}_n$ the following definitions[2] with [25]:

2. $\forall factor.Favoured(factor) = Plaintiff \equiv pFactor(factor)$
3. $\forall factor.Favoured(factor) = Defendant \equiv dFactor(factor)$

The expression $Favoured(factor) = Plaintiff$ says that the party favoured by *factor* is the plaintiff (defendant in other expression).

For each case the factors in the case are specified with the following predicate:

[1] In the definitions below, $-\neg\varphi$ denotes φ, while if φ does not start with a negation, $-\varphi$ denotes $\neg\varphi$.

[2] The background factor triples <*factor,party,value*> of [3] are written here as *fPromotes*(*factor, value*) and *Favoured*(*factor*) = *party* to give compatibility

- *hasFactor*(*case*,*factor*), meaning that *factor* is a factor in *case*.

The following function expression is used to denote a case's decision:

- $outcome(case) = party$.

We now add the following new predicates to those used in [25]:

- *fPromotes*(*factor*, *value*), meaning that a given *factor* promotes a given *value*.
- *oPromotes*(*outcome*(*case*), *value*), meaning that a given outcome of a given *case* promotes a given *value*.

Finally, that one value $value_1$, is preferred over another, $value_2$, is expressed as follows:.

- $preferred(value_1, value_2)$

We add the following definition to $\mathcal{K}_n$[3].

4. $\forall case, value, party.Promotes(outcome(case), value, party) \equiv outcome(case) = party \supset oPromotes(outcome(case), value)$.

The expression $Promotes(outcome(case), value, party)$ reads as: the *outcome* of the *case* promotes the *value* if the *case* is decided for the *party*.

5.1 Formalisation of the Simple Schemes

We are now ready to specify the argument schemes of Section 2 as defeasible inference rules $\mathcal{R}_d$ of our ASPIC+ argumentation system. For readability we will not specify them with the rule symbol $\Rightarrow$ but with a double horizontal inference bar. Rule schemes will be named by expressions $Name(x_1, \ldots, x_n)$ where the predicate *Name* stands for the informal name of the rule and $x_1, \ldots, x_n$ are all free variables occurring in the scheme. These variables are replaced by ground terms for each instance of the scheme, resulting in closed formulas that are the names of the scheme instances according to the function *n* mentioned just before Definition 4.3.

VAS1$(cur, value_1, value_2, party_1, party_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(cur), value_1, party_1) \\ Promotes(outcome(cur), value_2, party_2) \\ party_1 \neq party_2 \\ value_1 \neq value_2 \\ preferred(value_1, value_2) \end{array}}{outcome(cur) = party_1}$$

VAS2$(case, factor, value)$:

$$\frac{\begin{array}{c} hasFactor(case, factor) \\ fPromotes(factor, value) \\ Favoured(factor) = party \end{array}}{Promotes(outcome(case), value, party)}$$

VAS3$(prec, value_1, value_2, party_1, party_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_1, party_1) \\ Promotes(outcome(prec), value_2, party_2) \\ outcome(prec) = party_1 \\ party_1 \neq party_2 \end{array}}{preferred(value_1, value_2)}$$

[3]The $\supset$ symbol denotes the material implication from standard propositional logic.

VAS4a$(cur, value_1, value_2, value_3, party_1, party_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(cur), value_2, party_2) \\ Promotes(outcome(cur), value_3, party_2) \\ preferred(value_3, value_2) \end{array}}{\neg \mathbf{VAS1}(cur, value_1, value_2, party_1, party_2)}$$

VAS4b$(prec, value_1, value_2, value_3, party_1, party_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_1, party_1) \\ Promotes(outcome(prec), value_3, party_1) \\ preferred(value_3, value_1) \end{array}}{\neg \mathbf{VAS3}(cur, value_1, value_2, party_1, party_2)}$$

If no values for one side or the other are promoted, we use a dummy null value; when comparing sets of values this comes out automatically, given any sensible value set ordering.

5.2 Formalisation of the General Representation of Schemes

To formalise reasoning about sets of values as in **VAS5** we now adapt the techniques used in [25] to formalise reasoning about sets of factors. To our many-sorted first-order language we add a sort for *sets of values*. Then the following function expression is used to denote the value sets denoted by an outcome of a case:

- $PromotedValues(case, party) = setOfValues$, meaning that deciding the *case* for this *party* promotes this *setOfValues*.

Then the contents of a value set are defined as follows:

4. $\forall case, value, party.value \in PromotedValues(case, party) \equiv Promotes(outcome(case), value, party))$

In this method, two things have to be taken care of. First, to correctly represent and reason with set-theoretic expressions, definitions concerning these expressions must be added to $\mathcal{K}_n$. Next, to ensure that a value belongs to a value set of a case if and only if specified as such, the predicate completions of the predicates *hasFactor* and *oPromotes*, as well as the unique-names and domain-closure axioms for objects satisfying these predicates have to be added to $\mathcal{K}_n$. Since both things can be done in exactly the same ways as was done in [25] for factor sets, we refer the reader to that paper for the technical details.

We now formalise a more general version of the argumentation scheme **VAS5** in which the simple notion of ordering sets according to cardinality has been replaced by any preference ordering on sets. We use *spreferred* as the preference relation between sets of values.

VAS5$(cur, setOfValues_1, setOfValues_12, party_1, party_2)$:

$$\frac{\begin{array}{c} PromotedValues(cur, party_1) = setOfValues_1 \\ PromotedValues(cur, party_2) = setOfValues_2 \\ party_1 \neq party_2 \\ spreferred(setOfValues_1, setOfValues_2) \end{array}}{outcome(cur) = party_1}$$

The preference ordering on value sets remains to be specified. One method is the lexicogaphic ordering used in [9], which combines a preference ordering on values with cardinality of sets of values. In that paper an ASPIC+ formalisation of the lexicographic ordering to value sets was already presented, and so we can adapt that formalisation here. The definition given in [9] can be incorporated by adapting that paper's ASPIC+ formalisation of the lexicographic ordering to value sets. Since such an adaptation is entirely straightforward, we refer to that paper for the details.

To formalise reasoning about degrees of promotion we first change the logical language $\mathcal{L}$ as follows. The predicates *fPromotes*, *oPromotes* and *Promotes* now receive an additional place for *degrees*. For example,

$$Promotes(outcome(cur), value_1, party_1, degree_1)$$

reads as: the outcome of the current case promotes value $value_1$ to degree $degree_1$ if the current case is decided for $party_1$. Accordingly, a sort for degrees of promotion is added to $\mathcal{L}$. Then the preference relation is not defined on values but on value-degree pairs, so we now write expressions like the following, using *dvpreferred* for this preference relation:

$$dvpreferred((value_1, degree_1), (value_2, degree_2))$$

We also add an ordering predicate $\leq$ for degrees to $\mathcal{L}$, intended to be a total ordering on degrees of promotion, and where $<$ is defined as usual. Accordingly, the axioms of a total ordering are assumed to be in $\mathcal{K}_n$. For example, the degrees in CATO could be captured as $weak < strong$. To capture *a fortiori* reasoning with degrees, we then add the following axiom to $\mathcal{K}_n$:

5. $\forall value_1, value_2, degree_1, degree_2, degree_3, degree_4.$
   $dvpreferred((value_1, degree_1), (value_2, degree_2)) \wedge$
   $degree_1 \leq degree_3 \wedge degree_4 \leq degree_2 \supset$
   $dvpreferred((value_1, degree_3), (value_2, degree_4))$

Then argument schemes **VAS1-VAS4a,b** are rewritten as follows.

VAS1'$(cur, value_1, value_2, party_1, party_2, degree_1, degree_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(cur), value_1, party_1, degree_1) \\ Promotes(outcome(cur), value_2, party_2, degree_2) \\ party_1 \neq party_2 \\ value_1 \neq value_2 \\ dvpreferred((value_1, degree_1), (value_2, degree_2)) \end{array}}{outcome(cur) = party_1}$$

VAS2'$(case, factor, value, degree)$:

$$\frac{\begin{array}{c} hasFactor(case, factor) \\ fPromotes(factor, value, degree) \\ Favoured(factor) = party \end{array}}{Promotes(outcome(case), value, party, degree)}$$

VAS3'$(prec, value_1, value_2, party_1, party_2, degree_1, degree_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_1, party_1, degree_1) \\ Promotes(outcome(prec), value_2, party_2, degree_2) \\ outcome(prec) = party_1 \\ party_1 \neq party_2 \end{array}}{dvpreferred((value_1, degree_1), (value_2, degree_2))}$$

VAS4a'$(cur, value_1, value_2, value_3, party_1, party_2, degree_2, degree_3)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_2, party_2, degree_2) \\ Promotes(outcome(prec), value_3, party_2, degree_3) \\ dvpreferred((value_3, degree_3), (value_2, degree_2)) \end{array}}{\neg \mathbf{VAS1'}(cur, value_1, value_2, party_1, party_2, degree_1, degree_2)}$$

VAS4b'$(prec, value_1, value_2, value_3, party_1, party_2, degree_1, degree_3)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_1, party_1, degree_1) \\ Promotes(outcome(prec), value_3, party_2, degree_3) \\ dvpreferred((value_3, degree_3), (value_1, degree_1)) \end{array}}{\neg \mathbf{VAS3'}(cur, value_1, value_2, party_1, party_2, degree_1, degree_2)}$$

Then in **VAS5** we do not consider sets of values but sets of value-degree pairs, by adapting axiom 4 as follows:

6. $\forall case, value, party, degree.$
   $(value, degree) \in GradpromotedValues(case, party) \equiv$
   $Promotes(outcome(case), value, party, degree))$

Scheme **VAS5** is then changed to:

VAS5'$(cur, setOfGradValues_1, setOfGradValues_2, party_1, party_2)$:

$$\frac{\begin{array}{c} GradpromotedValues(cur, party_1) = setOfGradValues_1 \\ GradpromotedValues(cur, party_2) = setOfGradValues_2 \\ party_1 \neq party_2 \\ dvspreferred(setOfGradValues_1, setOfGradValues_2) \end{array}}{outcome(cur) = party_1}$$

Note that we now speak of *dvspreferred*, since we now use a preference relation on sets of value-degree pairs. Again this ordering can be defined in various ways, for instance, by a lexicographic ordering as in [9].

It is also possible to retain the preference ordering on values (now renamed to *vpreferred*) and to extend the *a fortiori* axioms to the value preference ordering. The first question to be asked here is how the value preferences can be derived from case decisions now that cases are specified with value-degree pairs instead of values simpliciter. For this we propose the following scheme, analoguous to scheme **VAS3**:

VAS6$(prec, value_1, value_2, party_1, party_2, degree_1, degree_2)$:

$$\frac{\begin{array}{c} Promotes(outcome(prec), value_1, party_1, degree_1) \\ Promotes(outcome(prec), value_2, party_2, degree_2) \\ outcome(prec) = party_1 \\ party_1 \neq party_2 \\ degree_1 \leq degree_2 \end{array}}{vpreferred(value_1, value_2)}$$

Then the new *a fortiori* axiom is:

7. $\forall value_1, value_2, value_3, value_4, degree_1, degree_2.$
   $dvpreferred((value_1, degree_1), (value_2, degree_2)) \wedge$
   $vpreferred(value_1, value_3) \wedge$
   $vpreferred(value_4, value_2) \supset$
   $dvpreferred((value_1, degree_3), (value_2, degree_4))$

Note that this axiom assumes that degrees of promotion for different values are commensurable.

In future research, we hope to investigate how our model can be enriched with Sartor's [27] more fine-grained methods comparing value sets: for example considering demoted values as well as promoted values, and exploring the effects of a marginal increase in promotion or demotion of values.

5.3 Consistency and Closure

Consistency and strict closure of Dung extensions generated by our formalisation can be verified as follows. The only nontrivial condition that needs to hold to make the results of [23, 20] apply is that the closure of the axiom premises $\mathcal{K}_n$ under strict rules is consistent (that is, does not contain two formulas φ and $\neg\varphi$). Since we chose $\mathcal{L}$ to be a first-order language and $\mathcal{R}_s$ to be all valid first-order inferences, this reduces to classical consistency and we can use the completeness theorem of first-order logic as follows. Create a model that is as simple as possible, with, for example, just one current case, one precedent, one factor and one value, and then verify that all axioms are true in that model. For the details of how this can be done see [25].

5.4 Example

For our second example we will use the automobile exception to the Fourth Amendment, as featured in *California v Carney* (1985), introduced into AI and Law in [26], and discussed in [8]. A fuller line of cases is discussed in [4]. The Fourth Amendment protects against unreasonable search: by *reasonable* it is generally meant that there is "good cause" for the search, and good cause is normally shown by obtaining a warrant. In some cases, however, it is not possible to obtain a warrant: the Carrol case (1925) established that an automobile travelling on the freeway could be searched without a warrant if there was good cause. In this case, where we have factors such as *vehicle* and *onHighway* that relate to the values of *privacy* and *exigency* respectively, deciding for the search strongly promotes exigency, while deciding against the search promotes privacy. As time went on importance came to be given to the diminished expectations of privacy relating to a motor vehicle (explicitly stated in *South Dakota v Opperman* (1976)), which is subject to inspection arising from traffic regulations. Also at issue might be where the car was parked (roadside, garage, suspect's driveway), and the source of the good cause, in particular when the contraband was known to be in a container (suitcase, bag, etc) seen to be placed in the car.

In the case of *Carney v California*, the search was of a mobile home from which Carney was dealing drugs, parked in a San Diego parking lot. Degrees of promotion are important: the location is more exigent than would have been a trailer park, but less than the freeway of the *Carrol* case. The privacy was more than a car, but less than a home. How privacy in this case relates to luggage seems to be disputed by the Justices. What seems to have been decided (by the majority at least) was that the location was important here: a mobile home in a car park would, because in use *as a vehicle*, only weakly promote privacy, whereas in a trailer park requiring a warrant to search a mobile home would promote privacy strongly (it being used as a home, or at least like a hotel room). The third value considered seems to be something like scrupulous *observance* of the constitution, since a number of Justices (especially Marshall and Brennan) consistently argue that a warrant should be obtained wherever it is possible to do so, even if a search might be justified without one. Thus the search in *Arkansas v Sanders* (1977), which involved the search an unlocked suitcase that the police observed being placed in the car meant that it could have been stopped and a warrant obtained before exigency became a consideration.

We will use three cases for our example. Note that we take *Vehicle* as a factor. Thus a mobile home in use as a vehicle (on the roads or in a public lot) will mean that the factor *Vehicle* applies: a mobile home in a trailer park would give the factor *Dwelling*. This is intended to reflect the majority opinion, but like several aspects of the analysis here is debateable (especially as regards the issues discussed in 6.3). Our purpose, however, is only to illustrate our schemes, not to definitively analyse this important line of cases.

Factors we use are:

- *hasFactor*(*Carrol*, *OnHighway*)
- *hasFactor*(*Carrol*, *Vehicle*),
- *hasFactor*(*Carney*, *ParkingLot*)
- *hasFactor*(*Carney*, *Vehicle*)
- *hasFactor*(*Sanders*, *Vehicle*)
- *hasFactor*(*Sanders*, *Unlocked*)

Factors promote values as follows:

- *fPromotes*(*OnHighway*, *Exigency*, *Strong*)
- *fPromotes*(*ParkingLot*, *Exigency*, *Strong*)
- *fPromotes*(*Vehicle*, *Privacy*, *Weak*)
- *fPromotes*(*Unlocked*, *Privacy*, *Weak)*

Now the argument for the majority in *Carney* using **VAS5'** is (distinctness premise omitted):

$$\frac{\begin{array}{c} GradpromotedValues(Cur,Cal) = \{(Exigency,Strong)\} \\ GradpromotedValues(Cur,Carney) = \{(Privacy,Weak)\} \\ dvspreferred(\{(Exigency,Strong)\},\{(Privacy,Weak)\}) \end{array}}{outcome(Cur) = Cal}$$

The dvspreference is established from *Carrol*, where we also have strongly promoted exigency and weakly promoted privacy. The minority, however, argue strongly that because the vehicle was parked in a lot in downtown San Diego, a warrant could have been obtained. We must add $fPromotes(ParkingLot, Observance, Strong)$. Their argument based on **VAS5'** is therefore:

$$\frac{\begin{array}{c} GradpromotedValues(Cur,Carney) = \\ \{(Privacy,Weak),(Observance,Strong)\} \\ GradpromotedValues(Cur,Cal) = \{(Exigency,Strong)\} \\ dvspreferred(\{(Privacy,Weak),(Observance,Strong)\}, \\ \{(Exigency,Strong)\}) \end{array}}{outcome(Cur) = Carney}$$

This preference would follow from *Sanders* where exigency was strong because the suitcase containing the drugs had been placed in a car, but privacy was weak (it was unlocked), and observance was strong (the police could have prevented the suitcase being put in the car). In practice this preference seems to have been rejected by the majority in *Carney*, possibly because they disagree with the observance value altogether. The situation remained unclear until *California v Avecedo* (1991) where it is explicitly argued that denying warrantless search on the grounds that a container could have been detained before being placed in an automobile affordes only very limited protection of privacy under the Fourth Amendment, and the current doctrine failed to provide a clear guideline

> The Chadwick-Sanders rule affords minimal protection to privacy interests. ... The Chadwick-Sanders rule also is the antithesis of a clear and unequivocal guideline and, thus, has confused courts and police officers and impeded effective law enforcement.

This confirms the weak promotion of privacy in *Sanders* and introduces a fourth value, *Clarity*, into the topic and expresses a new preference for $\{(Clarity,Strong),(Exigency,Weak)\}$ over $\{(Privacy,Weak),(Observance,Strong)\}$.

Probably the majority in *Avecedo* would support the view that strong exigency should be preferred to all other values, or combinations of values - the "automobile exception" in its purest form. But this is not consistent with some previous decisions. The prime virtue of the automobile exception is its clarity, and adding this additional value into consideration does enable their position to be reconciled with the previous body of case law.

6. DISCUSSION

Here we discuss some topics arising from the previous sections.

6.1 Transparency

Above we have primarily discussed argumentation systems, which are traditional, "good old-fashioned" symbolic AI. But we also briefly considered a highly quantitive approach, which took us very close to the statistical techniques of sub-symbolic AI. In symbolic AI a set of heuristics is discovered by analysis or by elicitation from experts. The resulting theory is then reasoned with to produce deductive chains from facts to conclusions. Examples are traditional expert systems and systems such as HYPO and CATO, especially as reconstructed in work such as [10], where the analysis is explicitly related to heuristics. In contrast are statistical, sub-symbolic, approaches in which techniques are applied to produce a kind of black box that responds to input by supplying an output. Data mining [19] and neural net systems [12] are examples. One important difference, especially for systems relating to the law, is transparency: the first kind of system is capable of providing a much more satisfying explanation than the other.

Apart from the explanation - which corresponds to arguments that, unlike mathematical calculations using a statistical model, can be presented in court - another issue concerns the nature of knowledge found in the background represented in the knowledge base, and the way this knowledge can be validated. The case and factor background of HYPO, CATO and [10] is readily understandable and can be questioned. We can argue about whether the analyst should have assigned a given factor to a particular case or not, whether the set of factors is complete, whether the factors are correctly related to values, and even whether values are strongly or weakly promoted by various factors. All of these can be the subject of reasoned debate in the light of the text of decisions, commentaries and the like. The numbers required as value weights and factor weights in [15] are a very different matter (as are the various weightings of a converged neural net). There the numbers are justified by the learning process which produced them and the quality of performance against test data. This kind of justification is very much scientific rather than legal.

Thus both in terms of explanation and in terms of verification and validation, both of which are essential to provide the trust required by legal systems, systems of the first type seem to be more suited for the legal domain.

6.2 Factors and Values

One of the aims of this paper has been to provide a clear way to compare the role of factors and values in reasoning with legal cases. We now have argument schemes for reasoning with both. The purpose of values was mainly to allow the reasoning to go beyond what was explicit in the precedents, so that we could give a justification of decisions (as opposed to possible arguments) in cases where there is no exactly matching precedent available. But, as we have seen, this requires comparison between sets of values. For some applications, where the difference in strength between values is sufficiently large, we can effectively compare sets of their strongest distinct member, and so this presents little difficulty. As we increase the sophistication of comparisons between sets of values, however, and even more when we also allow different degrees of value promotion, the dangers of compromising transparency, explanation and validation start to require very serious consideration. We can, however, say that using values always reduces the problem, since the numbers become smaller. Thus if we have five values we have only thirty two possible sets of values, and so around 1000 cases would in principle be enough to justify every possible comparison. In contrast there are more than twenty five factors in CATO, which would require in excess of 2^{25} cases to provide a full set of comparisons for sets of factors - and this is clearly infeasible. Even if, for example subsumption was used to reduce this, relying on the assumption that adding a value always increases the worth of a set, we would still need more cases that could possibly be collected and analysed. But if we restrict ourselves to a sufficiently small number of values and have a reasonably large case base, we could use *a fortiori* reasoning on cases represented as sets of values. This, however, does rely on having a small number of values - and ignoring degrees of promotion: even distinguishing strong and weak factors in a five value system would mean that there were potentially a million rather than a thousand set comparisons, each possibly needing a justifying case.

6.3 Assigning Degrees of Promotion

In our presentation of the Fourth Amendment cases in section 5.3, we have, in accordance with the approach of [10] taken the factors as fixed, and we have also taken the degree to which a given factor promotes a value as fixed. In practice, however, the dispute could be seen as about the degree to which particular values are promoted by various decisions in the particular cases. Thus in *Carney* the choice could be seen as whether a decision for no search of a mobile home in a public parking lot would promote privacy strongly or weakly. Or it could be whether a decision for search of a parked vehicle promotes exigency strongly or weakly. Similarly, does protecting luggage under the Fourth Amendment protect privacy strongly or weakly? Does it make a difference whether it is a suitcase rather than a bag, or whether the suitcase is locked? This moves the choice from values derived from factors to the ascription of factors themselves. But this reflects the importance of values: setting a low threshold for strong promotion of privacy means that privacy is regarded as more important, in that we do not require very great intrusions on privacy to bring the value into play.

The point can also be made using *Pierson v Post*. There the minority argued that finding for Post, and so encouraging the hunting of vermin for sport, would promote a value of *social utility*. Thus the question can be seen as whether the *hotPursuit* factor which can be seen as present in *Pierson* does indeed promote social utility to the degree necessary for this to be preferred to the (strong or weak?) promotion of legal clarity achieved by finding for Pierson. There are thus a number of choices that could be made:

1. whether finding for Post promotes social utility strongly or weakly;
2. whether finding for Pierson promotes legal clarity strongly or weakly;
3. whether either or both degrees of promotion of social utility are preferred to the promotion of either or both degrees of legal clarity.

These choices are independent of one another in that it is possible to hold that finding for Post promotes social utility strongly and finding for Pierson promotes legal clarity only weakly and yet still find for Pierson, if legal clarity is much preferred to social utility. In the account of this paper we deal only with point (3). This is true of most, if not all work, in AI and Law, which does tend to take the analysis in terms of factors as fixed. The closest to addressing this point is perhaps HYPO, where it is not stated at which point on a dimension the plaintiff ceases, and the defendant starts, to be favoured. But although HYPO can provide this possible opportunity for discussion, arguments of this sort were not explored in that work. Thus how to argue which side is favoured by a particular fact situation, how strongly they are favoured, the role of values, and the preferences between values in such arguments is certainly an aspect of reasoning with legal cases that merits further exploration.

7. CONCLUDING REMARKS

In this paper we have provided a set of argumentation schemes to make explicit the reasoning involved when considering preferences between social values that has been widely recognised as an important aspect of reasoning with cases. Our schemes have been formalised in ASPIC+, enabling us to give them a precise representation for use in computational systems. We have shown how our schemes can readily be applied to well-known cases, validating the benefits of capturing such reasoning explicitly.

Acknowledgements

During the preparation of this article, the third author was supported by the IMPACT Project (FP7-ICT-2009-4).

References

[1] V. Aleven. *Teaching Case-Based Argumentation Through a Model and Examples*. PhD thesis, Intelligent Systems Program, University of Pittsburgh, 1997.

[2] K. Ashley. *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. The MIT Press, 1990.

[3] T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.

[4] T. J. M. Bench-Capon. Relating values in a series of supreme court decisions. In K. Atkinson, editor, *Legal Knowledge and Information Systems - JURIX 2011: The Twenty-Fourth Annual Conference*, pages 13–22, 2011.

[5] T. J. M. Bench-Capon. Representing Popov v Hayashi with dimensions and factors. *Artif. Intell. Law*, 20(1):15–35, 2012.

[6] T. J. M. Bench-Capon, K. Atkinson, and P. McBurney. Using argumentation to model agent decision making in economic experiments. *Autonomous Agents and Multi-Agent Systems*, 25(1):183–208, 2012.

[7] T. J. M. Bench-Capon and S. Modgil. Case law in extended argumentation frameworks. In *The 12th International Conference on Artificial Intelligence and Law*, pages 118–127, 2009.

[8] T. J. M. Bench-Capon and H. Prakken. Using argument schemes for hypothetical reasoning in law. *Artif. Intell. Law*, 18(2):153–174, 2010.

[9] T. J. M. Bench-Capon, H. Prakken, and W. Visser. Argument schemes for two-phase democratic deliberation. In *Proceedings of the 13th International Conference on AI and Law*, pages 21–30, 2011.

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

[11] D. H. Berman and C. D. Hafner. Representing teleological structure in case-based legal reasoning: The missing link. In *Proceedings of the 4th International Conference on AI and Law*, pages 50–59, 1993.

[12] L. Bochereau, D. Bourcier, and P. Bourgine. Extracting legal knowledge by means of a multilayer neural network application to municipal jurisprudence. In *Proceedings of the Third International Conference on Artificial Intelligence and Law*, pages 288–296, 1991.

[13] J.-F. Bonnefon, R. D. S. Neves, D. Dubois, and H. Prade. Background default knowledge and causality ascriptions. In *Proceedings of ECAI 2006, 17th European Conference on Artificial Intelligence*, pages 11–15, 2006.

[14] S. Brüninghaus and K. D. Ashley. Predicting outcomes of case-based legal arguments. In *Proceedings of the 9th International Conference on AI and Law*, pages 233–242, 2003.

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

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

[17] T. F. Gordon. The pleadings game. *Artif. Intell. Law*, 2(4):239–292, 1993.

[18] J. F. Horty and T. J. M. Bench-Capon. A factor-based definition of precedential constraint. *Artif. Intell. Law*, 20(2):181–214, 2012.

[19] B. Johnston and G. Governatori. Induction of defeasible logic theories in the legal domain. In *Proceedings of the 9th International Conference on Artificial Intelligence and Law*, 2003.

[20] S. Modgil and H. Prakken. A general account of argumentation with preferences. *Artif. Intell.*, 2013. In press.

[21] F. S. Nawwab, P. E. Dunne, and T. J. M. Bench-Capon. Exploring the role of emotions in rational decision making. In *Proceedings of the 3rd COMMA Conference*, pages 367–378, 2010.

[22] C. Perelman and L. Olbrechts-Tyteca. *The new rhetoric. A treatise on argumentation.* Notre Dame Press, 1969.

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

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

[25] H. Prakken, A. Wyner, T. Bench-Capon, and K. Atkinson. A formalisation of argumentation schemes for legal case-based reasoning in ASPIC+. *Journal of Logic and Computation*, 0(0):0–0, 2013. To appear.

[26] E. L. Rissland. Dimension-based analysis of hypotheticals from supreme court oral argument. In *Proceedings of the Second International Conference on Artificial Intelligence and Law*, pages 111–120, 1989.

[27] G. Sartor. Doing justice to rights and values: teleological reasoning and proportionality. *Artif. Intell. Law*, 18(2):175–215, 2010.

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

[29] A. Z. Wyner and T. J. M. Bench-Capon. Argument schemes for legal case-based reasoning. In *Proceedings of Jurix 2007*, pages 139–149, 2007.

[30] A. Z. Wyner, T. J. M. Bench-Capon, and K. Atkinson. Towards formalising argumentation about legal cases. In *Proceedings of the 13th International Conference on AI and Law*, pages 1–10, 2011.