Instantiating Knowledge Bases in Abstract Argumentation Frameworks

Adam Wyner
University College London
Department of Computer Science

Trevor Bench-Capon and **Paul Dunne**
University of Liverpool
Department of Computer Science

Abstract

Abstract Argumentation Frameworks (AFs) provide a fruitful basis for exploring issues of defeasible reasoning. Their power largely derives from the abstract nature of the arguments within the framework, where arguments are atomic nodes in an undifferentiated relation of attack. This abstraction conceals different conceptions of argument, and concrete instantiations encounter difficulties as a result of conflating these conceptions. We distinguish three distinct senses of the term. We provide an approach to instantiating AFs in which the nodes are restricted to literals and rules, encoding the underlying theory directly. Arguments, in each of the three senses, then emerge from this framework as distinctive structures of nodes and paths. Our framework retains the theoretical and computational benefits of an abstract AF, while keeping notions distinct which are conflated in other approaches to instantiation.

Introduction

Abstract Argumentation Frameworks (AFs) ((Dung 1995), (Bondarenko et al. 1997), (Caminada and Amgoud 2007), among others) provide a fruitful basis for exploring issues of defeasible reasoning. Their power largely derives from the abstract nature of the arguments within the framework, where arguments are atomic nodes in an undifferentiated relation of attack; such AFs provide a very clean acceptability semantics, e.g. (Dunne and Bench-Capon 2002).

While abstract approaches facilitate the study of arguments and the relations between them, it is necessary to instantiate arguments to apply the theory. Methods for instantiation have been proposed which combine AFs with Logic Programs ((Prakken and Sartor 1997), (García and Simari 2004), (Besnard and Hunter 2008) and (Amgoud et al. 2004)). Such systems start with a knowledge base comprised of facts and rules, where the rules typically may include both strict (*SI*) and defeasible (*DI*) inference rules. Arguments are generated from this knowledge base using an appropriate logic and organised into an AF for evaluation.

While such approaches have attractions, (Caminada and Amgoud 2007) point out they also have difficulties. Starting with a benchmark instantiation method as a basis for comparing the approaches, they show the approaches give rise to counter-intuitive results. More generally, the theories fail to account for the *rationality postulates* of *consistency* and *closure* on argumentation frameworks: to satisfy *consistency*, the theory must ensure that it does not return both a postive and negative literal from an extension of arguments; to satisfy *closure*, the literals returned by an extension of arguments must be closed such that if the antecedents of a strict rule are returned, then so too is the claim of that rule. They suggest ways to amend the knowledge base, allowing the approaches to account for the problems and satisfy the *rationality postulates*.

In our view, the problems of the approaches discussed in (Caminada and Amgoud 2007) stem from the method of instantiating the knowledge base in an AF along with the notions of argument defeat and justified conclusion. Knowledge bases are instantiated in a two-step process: from the knowledge base, one constructs the arguments; given the arguments, one determines the relation (defeat) among the arguments, puts the arguments (as nodes) into an AF, and determines the justified conclusions. A two-step process introduces a variable, indirect relationship between the knowledge base and the AF, and different approaches instantiate the knowledge base in different ways. In the benchmark approach of (Caminada and Amgoud 2007), arguments can have sub-arguments, so arguments are no longer atomic as in (Dung 1995). An argument may attack the subarguments of another argument, again varying the uniform attack relation of (Dung 1995). In addition, defeat of a subargument implies defeat of the whole argument. Such attacks imply that additional, potentially significant, information is lost when the framework is evaluated, presenting a problem for the clean acceptability semantics. Finally, we show that a notion of justified conclusions which does not distinguish between defeasible and strict justifications may lead to an expectation of further conclusions which in turn contravene the rationality postulates.

In this paper, we provide a novel approach to instantiating an AF and to defining arguments which does not have the problems and concerns just outlined. Our approach gives correct results on the problems, intuitively satisfies the *rationality postulates*, and addresses the concerns. We present a direct, single-step translation of a knowledge base into an AF, providing a graph theoretic representation of the knowledge base itself. Every rule in the knowledge base is represented in the AF. We have a uniform attack relation between nodes, which is given by incompatibility: complementary literals are incompatible, the negation of a rule antecedent is incompatible with the applicability of the rule, and the applicability of a rule is incompatible with the falsity of its consequent. The various argumentation attack relations (rebuttal, premise defeat, and undercut) are derivative on the AF structure rather than being independently defined and being required to determine argument attack relations. The admissibility of a node indicates that the literal holds or the rule is applicable. Thus, we evaluate claims of the theory directly, without first going through the step of generating arguments. Given that we derive AFs from consistent knowledge bases and we only output admissible sets, *consistency* is ensured. We indicate how *closure* is satisfied given the construction of strict rules in derived AFs. Our approach retains the appeal of AFs, evaluates the AF with the well understood semantics, allows reasoning with respect to knowledge bases, and retains the appropriate level of abstraction of the nodes of the AF.

In (Wyner, Bench-Capon, and Atkinson 2008), several senses of the term *argument* were identified and the problems of conflating them were discussed: a specific reason for a claim (which we call an *Argument*), a set of such specific reasons for a claim (which we term a *Case*), or a set of reasons for and against a claim (which we refer to as a *Debate*). In our approach, these various senses of *argument* emerge from the framework as distinctive structures in an AF; keeping them distinct avoids the confusions that can arise when these different senses are conflated. In particular no *Argument* has subarguments. Our approach clarifies justification and explanation of conclusions in a complex debate.

The contribution of this paper is that it provides an integrated framework in which we represent a knowledge base with SI and DI within an AF so that argument generation and evaluation are performed in a single step and in which we distinguish the alternative senses of *argument* to bring out their distinctive features.

The structure of the paper is as follows. We first outline AFs (Dung 1995) and characterise the types of knowledge base we are working with. We then show how a knowledge base is represented in a derived AF. We provide examples and discuss non-monotonic reasoning. The different senses of *argument* and the various kinds of attack of one *argument* by another are then characterised in terms of particular structures within the AF. We discuss the benchmark approach to knowledge base instantiation of (Caminada and Amgoud 2007) along with the problems raised by a key example. We show how our approach addresses the problems of the example. We end with some concluding remarks and future work.

Argumentation Frameworks

An *Argumentation Framework* AF comprises objects, relations, and definitions of auxiliary concepts. For our purposes, we take (Dung 1995) as the basis for an AF.

Definition 1 *An argumentation framework* AF *is a pair* $\langle \mathcal{L}^A, \mathcal{R}^A \rangle$*, where* $\mathcal{L}^A$ *is a finite set of* objects*,* $\{p_1, p_2, \ldots, p_n\}$ *and* $\mathcal{R}^A$ *is an* attack *relation between elements of* $\mathcal{L}^A$*. For* $\langle p_i, p_j \rangle \in \mathcal{R}^A$ *we say the the object* p_i *attacks object* p_j*. We assume that no object attacks itself.*

The relevant auxiliary definitions are as follows, where S is a subset of $\mathcal{L}^A$:

Definition 2 *We say that* $p \in \mathcal{L}^A$ *is* acceptable with respect to S *if for every* $q \in \mathcal{L}^A$ *that attacks* p *there is some* $r \in S$ *that attacks* q*. A subset,* S*, is* conflict-free *if no object in* S *is attacked by any other object in* S*. A conflict-free set* S *is* admissible *if every* $p \in S$ *is acceptable to* S*. A* preferred extension *is a maximal (w.r.t.* $\subseteq$*) admissible set. The object* $p \in \mathcal{L}^A$ *is* credulously accepted *if it is in at least one preferred extension, and* sceptically accepted *if it is in* every *preferred extension.*

As we clarify the notion of *argument* itself, we refer to the basic objects as *nodes* (denoted by $\mathcal{L}^A$) and their relations as *arcs* (denoted by $\mathcal{R}^A$); indeed, we do not want to introduce presumptions about the properties of the objects, in particular what should count as an argument.

Representing a Theory as an AF

The approach has two basic parts. In the first part, we represent a Theory Base $\mathcal{T}$, which represents the knowledge base, directly in an AF: the nodes of an AF are labeled with respect to the literals and inference rules of the Theory Base, while the attack relation is partitioned with respect to the nodes. SI and DI are represented as distinct structures of nodes and attack relations. In the second part, we impose conditions on the assertion of literals with respect to the AF. The AF is then evaluated according to Definitions 1 and 2. We have used $\mathcal{L}^A$ and $\mathcal{R}^A$ in AFs and $\mathcal{L}$ and $\mathcal{R}$ in Theory Bases to highlight the correspondence between the nodes of the AF with the language of the Theory Base as well as the relationships between the nodes in the AF with the rules in the Theory Base from which the relationships are derived.

Definition 3 *A* Theory Base*,* $\mathcal{T}$*, comprises a pair* $(\mathcal{L}, \mathcal{R})$ *in which*

$$\mathcal{L} = \{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\}$$

is a set of literals over a set of propositional variables $\{x_1, \ldots, x_n\}$*. We use* y_i *to denote an arbitrary literal from* $\{x_i, \neg x_i\}$*.*

The component $\mathcal{R}$ *describes a set of* rules

$$\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$$

where $\mathcal{R} = \mathcal{R}_{str} \cup \mathcal{R}_{dfs}$*, in which* $r \in \mathcal{R}$ *has a body, bd(r)* $\subseteq \mathcal{L}$*, and a head, hd(r)* $\in \mathcal{L}$*. Rules are either strict (*$r \in \mathcal{R}_{str}$*) or defeasible (*$r \in \mathcal{R}_{dfs}$*), and* $\mathcal{R}_{str} \cap \mathcal{R}_{dfs} = \emptyset$*.*

We informally refer to the literals in bd(r) as *premises* and the literal in hd(r) as the *claim*.

We use $\{r_1, r_2, \ldots, r_n\}$ as *proper names* of rules. For easy reference to the "content" of the rule, we assume each rule has an associated *definite description* as follows. For $r \in \mathcal{R}_{str}$, the definite description of r has the form r :

$bd(r) \rightarrow hd(r)$, where $hd(r) \in \mathcal{L}$ and $bd(r) \subseteq \mathcal{L}$. Similarly, the definite description for $r \in \mathcal{R}_{dfs}$, has the form $r : bd(r) \Rightarrow hd(r)$. Where a rule has an empty body, bd(r) = $\emptyset$, we have $r :\rightarrow hd(r)$ or $r :\Rightarrow hd(r)$, which are *strict* and *defeasible* assertions, respectively. To refer distinctly to the set of rules with non-empty bodies and those with empty bodies (*assertions*), we have $\mathcal{R}$ = TRules ∪ ARules, where TRules = $\{r \mid r \in \mathcal{R} \wedge bd(r) \neq \emptyset\}$ and ARules = $\{r \mid r \in \mathcal{R} \wedge bd(r) = \emptyset\}$.

We constrain a Theory Base. First, the relationship between literals of strict and defeasible rules is constrained:

Constraint 1 *For* Theory Base $(\mathcal{L}, R)$, $\forall r \in \mathcal{R}_{str}$, *there is no rule, r' $\in \mathcal{R}_{dfs}$ with hd(r) = hd(r') and bd(r) $\subseteq$ bd(r').*

Furthermore, no literal and its negation can both be strictly asserted.

Constraint 2 *For* Theory Base $(\mathcal{L}, R)$, *if r $\in \mathcal{R}$, where r :$\rightarrow$ hd(r), then r' $\notin \mathcal{R}$, where r' :$\rightarrow \neg$hd(r).*

In addition, every literal appears in some rule.

Constraint 3 *For* Theory Base $(\mathcal{L}, R)$, *if y $\in \mathcal{L}$, then $\exists$ r $\in \mathcal{R}$, y $\in$ bd(r) $\vee$ y = hd(r).*

Finally, every rule has a claim.

Constraint 4 *For* Theory Base $(\mathcal{L}, \mathcal{R})$, *if r $\in \mathcal{R}$, then $\exists$ y $\in \mathcal{L}$, y = hd(r).*

Semantically, a rule $r \in \mathcal{R}_{str}$ represents the notion that hd(r) holds if *all* of the literals in bd(r) simultaneously hold; with respect to the rule, we say the bd(r) strictly implies the hd(r). We assume standard notions of *truth* and *falsity* of literals along with the truth-tables of Propositional Logic for material implication which are models under which the rule is *true* or *false*. Semantically, a rule $r \in \mathcal{R}_{dfs}$ represents the notion that hd(r) "usually" holds if *all* of the literals in bd(r) simultaneously hold, but there are circumstances where ¬hd(r) holds though *all* of the literals in bd(r) simultaneously hold. With respect to the rule, we say the bd(r) defeasibly implies the hd(r).

While the clauses are similar to the *Horn Clauses* of logic programming, the head literal can be in a positive or negative form. We only have classical negation, not negation as failure; we do not allow iterated negation. The rationale for this choice of clauses is that it naturally supports our analysis of the senses of *argument*.

A core element of our approach is the concept of the AF derived from a Theory Base. The AF uses a set of labels for the nodes in the graph: $\{\mathsf{x}_1,\ldots,\mathsf{x}_n\}\cup\{\neg\, \mathsf{x}_1,\ldots,\neg\, \mathsf{x}_n\}\cup\{\mathsf{r}_1,\ldots,\mathsf{r}_n\}$ (or for clarity, the definite description of the rule name). Thus, we can see how elements of the Theory Base correspond to elements of the derived AF; however, the elements are distinct in terms of their intuitive content and function.

Definition 4 *Let $\mathcal{T} = (\mathcal{L}, \mathcal{R})$ be a Theory Base with*

$$\begin{array}{lcl} \mathcal{L} & = & \{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, \neg x_n\} \\ \mathcal{R} & = & \mathcal{R}_{str} \cup \mathcal{R}_{dfs} \end{array}$$

The derived framework *from $\mathcal{T}$, is the* AF, $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$ *in which,*

$$\begin{array}{lcl} \mathcal{L}^A_{\mathcal{T}} & = & \{\, \mathsf{x},\, \neg \mathsf{x} \,:\, x, \neg x \,\in \mathcal{L}\} \\ & & \cup \,\{\, \mathsf{r : bd(r) \rightarrow hd(r)} \,:\, r \,\in \mathcal{R}_{str}\} \\ & & \cup \,\{\, \mathsf{r : bd(r) \Rightarrow hd(r)} \,:\, r \,\in \mathcal{R}_{dfs}\} \end{array}$$

Furthermore,

$$\begin{array}{l} \forall \mathsf{x} \in \mathcal{L}^A_{\mathcal{T}}, x \,\in \mathcal{L},\ and \\ \forall \mathsf{r} \in \mathcal{L}^A_{\mathcal{T}}, r \,\in \mathcal{R} \end{array}$$

In an AF, the nodes have no internal content.

The attack set $\mathcal{R}^A_{\mathcal{T}}$ comprises three disjoint sets which describe: attacks by nodes labeled with names for literals on other nodes labeled with names for literals; attacks by nodes labeled with names for literals on nodes labeled with names for rules; and attacks by nodes labeled with names for rules on nodes labeled with names for literals. We recall that $y_i \in \{x_i, \neg x_i\}$ so that $\neg y_i$ is the complementary literal to y_i.

Definition 5 *In the* AF $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$, $\mathcal{R}^A_{\mathcal{T}}$ = $\mathcal{R}^A_{ll} \cup \mathcal{R}^A_{lr} \cup \mathcal{R}^A_{rl}$ *where:*

$$\begin{array}{lcl} \mathcal{R}^A_{ll} & = & \{\langle \mathsf{y}_i, \neg \mathsf{y}_i\rangle,\ \langle \neg \mathsf{y}_i, \mathsf{y}_i\rangle \,:\, 1 \le i \le n \\ & & and\ \mathsf{y}_i, \neg \mathsf{y}_i \in \mathcal{L}^A_{\mathcal{T}}\} \\ \mathcal{R}^A_{lr} & = & \{\langle \neg \mathsf{y}_i, \mathsf{r}_j\rangle \,:\, y_i \in bd(r_j)\ and\ \neg \mathsf{y}_i, \mathsf{r}_j \in \mathcal{L}^A_{\mathcal{T}}\} \\ & & \cup\, \{\langle \neg \mathsf{y}_i, \mathsf{r}_j\rangle \,:\, r_j \in \mathcal{R}_{dfs}\ and\ hd(r_j) = y_i \\ & & and\ \neg \mathsf{y}_i \in \mathcal{L}^A_{\mathcal{T}}\} \\ \mathcal{R}^A_{rl} & = & \{\langle \mathsf{r}_j, \neg \mathsf{y}_i\rangle \,:\, hd(r_j) = y_i\ and\ \neg \mathsf{y}_i, \mathsf{r}_j \in \mathcal{L}^A_{\mathcal{T}}\} \end{array}$$

The following hold for an AF derived from a $\mathcal{T}$:

1. Each literal y in $\mathcal{L}$ of Theory Base $\mathcal{T}$ corresponds to a node labeled y in $\mathcal{L}^A$ of the derived AF; $\mathcal{L}^A$ of the derived AF contains, in addition, the node labeled $\neg\mathsf{y}$. Nodes labeled for literals of opposite polarity are mutually attacking.
2. Each rule in r in $\mathcal{R}$ of a Theory Base $\mathcal{T}$ corresponds one-to-one to a node label r in $\mathcal{L}^A$ of the derived AF. Whereas a rule in $\mathcal{R}$ is true (or false) in the Theory Base, in the derived AF we say it *has been applied* relative to the admissible set where it appears and otherwise *has not been applied*. In the AF, a rule node is attacked by the nodes which correspond to the negation of the body literals and, in addition, attacks the node which corresponds to the negation of the head literal.
3. For *strict* rules, the rule node in the AF has not been applied relative to an admissible set if a node which corresponds to the negation of one of the body literals is in that set. In this case, the node which corresponds to the head literal is only credulously admissible. If all the nodes which correspond to the body literals are in an admissible set, then the rule node has been applied and the node which corresponds to head literal is admissible in that set. This essentially fulfills the *closure* rationality postulate.
4. For *defeasible* rules, the rule node has not been applied relative to an admissible set if a node which corresponds to the negation of a body is in that set or if the node which corresponds to the negation of the head of the rule is in that set. In both instances, the node corresponding to the

literal attacks the rule node. Even if all nodes which correspond to the body literals of a rule are in an admissible set, the rule node or the node which corresponds to the head literal may not be in that set, for they can be defeated.

In addition, we have a condition relating to the ARules of $\mathcal{T}$ in the derived AF.

5. Consider r, $r' \in$ ARules (the rules of the theory $\mathcal{T}$ which have empty bodies). For $r \in \mathcal{R}_{str}$, if hd(r) $\in \mathcal{A}$, where $\mathcal{A}$ is an admissible set of the derived AF $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$, then the node which represents hd(r) is *sceptically acceptable* relative to the derived AF. For $r' \in \mathcal{R}_{dfs}$, hd(r') is *credulously acceptable* relative to the derived AF. We refer to these as *strict* and *defeasible assertions* in the AF. This is a constraint on acceptable AF semantics in this framework.

We evaluate the derived AFs only following the definitions of extensions relative to the standard AF $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$; that is, while the partitions of nodes or arcs are important for deriving the AF from $\mathcal{T}$, they are ignored for the purposes of evaluation, so that we have a standard abstract framework. Thus, the semantics of abstract AFs are maintained in evaluation.

To this point, we have Theory Bases and corresponding derived AFs. Fundamental observations of our approach are:

Observation 1 *For the literals and the rules which are* true *of a Theory Base $\mathcal{T}$, the corresponding nodes of the derived* AF *are elements of some admissible set. For the literals and the rules which are* true *of every model for the Theory Base $\mathcal{T}$, the corresponding nodes of the derived* AF *are sceptically acceptable, otherwise they are credulously acceptable.*

Observation 2 *For the literals and the rules which are* false *of a Theory Base $\mathcal{T}$, the corresponding nodes of the derived* AF $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$ *are not an element of any admissible set.*

Both of these follow by the evaluation of a derived framework $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$ relative to a $\mathcal{T}$. Thus, the derived AF is *information-preserving* with respect to the Theory Base. The derived AF is an instantiation of the corresponding Theory Base, and the preferred extensions of the AF correspond to models of the Theory Base.

We now give some examples. First, we provide a Theory Base $\mathcal{T}_1$ with just one strict rule, the derived AF, a graphic representation of the derived AF, and then the preferred extensions. Since it is always clear in context where we have literals and rules (in a Theory Base) and where we have labels (in an AF), we use one typographic form without confusion.

Example 1 *Let $\mathcal{T}_1$ be the pair with ($\mathcal{L}_1$, $\mathcal{R}_1$), where*

$$\begin{array}{lcl} \mathcal{L}_1 & = & \{x_1, x_2\} \cup \{\neg x_1, \neg x_2\} \\ \mathcal{R}_1 & = & \{r_1\},\ where\ r_1\ has\ rule\ name\ r_1\!:\!x_1 \rightarrow x_2 \end{array}$$

The derived framework *from $\mathcal{T}_1$ is $\langle \mathcal{L}^A_{\mathcal{T}_1}, \mathcal{R}^A_{\mathcal{T}_1} \rangle$ in which,*

$$\mathcal{L}^A_{\mathcal{T}_1} \quad = \quad \{x_1, x_2\} \,\cup\, \{\neg x_1, \neg x_2\} \,\cup\, \{r_1\}$$

and in which $\mathcal{R}^A_{\mathcal{T}_1}$ comprises the union of three disjoint sets:

$$\begin{array}{lcl} \mathcal{R}^A_{ll} & = & \{\langle x_1,\neg x_1\rangle, \langle \neg x_1, x_1\rangle, \langle x_2, \neg x_2\rangle, \langle \neg x_2, x_2\rangle\} \\ \mathcal{R}^A_{lr} & = & \{\langle \neg x_1, r_1\rangle\} \\ \mathcal{R}^A_{rl} & = & \{\langle r_1, \neg x_2\rangle\} \end{array}$$

We graphically represent $\langle \mathcal{L}^A_{\mathcal{T}_1}, \mathcal{R}^A_{\mathcal{T}_1} \rangle$ as in Figure 1.

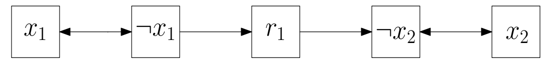

Figure 1: AF of $x_1 \rightarrow x_2$

In $\langle \mathcal{L}^A_{\mathcal{T}_1}, \mathcal{R}^A_{\mathcal{T}_1} \rangle$, the preferred extensions are:

$$\{x_1, r_1, x_2\}, \{\neg x_1, x_2\}, \{\neg x_1, \neg x_2\}$$

Each of the nodes is *credulously accepted* and none is *sceptically* accepted. The interpretation of the presence of a rule node in a preferred extension is that the rule *has been applied*. Moreover, the rule is not *defeated* in the sense that where the premises hold, the conclusion *must* hold. No admissible set contains both x_1 and $\neg x_2$: if x_1 is in the set, then r_1 is in the set; r_1 attacks $\neg x_2$, leaving x_2 in the set; if $\neg x_2$ is in the set, then r_1 must be attacked; r_1 can only be attacked by $\neg x_1$, which also attacks x_1, leaving $\neg x_1$ in the set.

The following is an example of a *defeasible* rule.

Example 2 *Let $\mathcal{T}_2$ be the pair with ($\mathcal{L}_2$, $\mathcal{R}_2$), where*

$$\begin{array}{lcl} \mathcal{L}_2 & = & \{x_1, x_2\} \cup \{\neg x_1, \neg x_2\} \\ \mathcal{R}_2 & = & \{r_2\},\ where\ r_2\ has\ rule\ name\ r_2\!:\!x_1 \Rightarrow x_2 \end{array}$$

We graphically represent the derived AF $\langle \mathcal{L}^A_{\mathcal{T}_2}, \mathcal{R}^A_{\mathcal{T}_2} \rangle$ as:

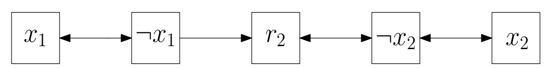

Figure 2: AF of $x_1 \Rightarrow x_2$

In $\langle \mathcal{L}^A_{\mathcal{T}_2}, \mathcal{R}^A_{\mathcal{T}_2} \rangle$, the preferred extensions are as follows, where we see that each of the nodes is *credulously accepted* and none is *sceptically* accepted.

$$\{x_1, r_2, x_2\}, \{\neg x_1, x_2\}, \{\neg x_1, \neg x_2\}, \{x_1, \neg x_2\}$$

The first three preferred extensions are similar to SI. In the last extension, $\neg x_2$ itself attacks the rule node r_2; consequently, either x_1 or $\neg x_1$ are in a preferred extension along with $\neg x_2$. This contrasts with the preferred extension of a derived AF with just a SI. While defeasible implication might be construed as the trivial logical tautology $[x_1 \rightarrow [x_2 \vee \neg x_2]]$, here we see a key difference. To make use of a defeasible rule, one must provide the means to *choose between extensions*, for example, by selecting the extension which maximises the number of applicable defeasible rules, or which uses some notion of priority or entrenchment on the rules. Different ways of making this choice give rise to different varieties of non-monotonic logic (Reiter 1980) and (Prakken and Sartor 1997)). Circumscription (McCarthy 1980) could be used by including additional designated nodes such as $ab(r_1)$ which attack the rule r_1 and attack and are attacked by $notab(r_1)$. We then choose the extension containing the most $notab(r_1)$ nodes. We can specify circumstances where the rule is not be applied.

In our third example, we show the interaction of defeasible and strict rules, which was the root of several of the problems identified in (Caminada and Amgoud 2007).

Example 3 *Suppose $\mathcal{T}_3$ with rules r_2: $x_1 \Rightarrow x_2$ and r_3: $x_2 \rightarrow x_3$ which has derived* AF $\langle \mathcal{L}^A_{\mathcal{T}_3}, \mathcal{R}^A_{\mathcal{T}_3} \rangle$ *graphically represented as in Figure 3.*

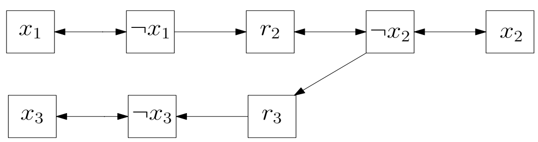

Figure 3: AF derived from $\mathcal{T}$ with $x_1 \Rightarrow x_2$ and $x_2 \rightarrow x_3$

AF $\langle \mathcal{L}^A_{\mathcal{T}_3}, \mathcal{R}^A_{\mathcal{T}_3} \rangle$ has the following six preferred extensions:

1.$\{x_1, r_2, x_2, r_3, x_3\}$	4.$\{\neg x_1, x_2, r_3, x_3\}$
2.$\{x_1, \neg x_2, x_3\}$	5.$\{\neg x_1, \neg x_2, x_3\}$
3.$\{x_1, \neg x_2, \neg x_3\}$	6.$\{\neg x_1, \neg x_2, \neg x_3\}$

Given a strict assertion that x_1, we would normally choose the preferred extension (1) from among (1)-(3), maximising the number of defeasible rules. Thus, normally, we say that x_1 implies x_3. However, we are not obliged to make this choice. In particular, if $\neg x_2$ is strictly asserted, r_2 and r_3 are inapplicable, and x_3 is credulously acceptable ((2) and (3)); thus, in this AF, a strict assertion of x_1 does not imply that x_3 necessarily holds as well. Where the claim of a defeasible rule is a premise of a strict rule (x_2), we cannot use the defeasibly inferred claim to draw strict inferences about the claim of the strict rule (x_3). Similarly, the defeasible rule is inapplicable where either the claim of the rule ($\neg x_2$) is false ((2), (3), (5), and (6)) or the claim of the strict rule ($\neg x_3$) is false ((3) and (6)). Whereas in e.g. (Reiter 1980), the defeasible rule is inapplicable only where the claim of the defeasible rule itself is asserted to be false, here the falsity of any consequences of that claim, however remote, will also block the application of the rule.

Three Senses of Argument

We show how arguments emerge from the framework as distinctive structures. As discussed in (Wyner, Bench-Capon, and Atkinson 2008), the term *argument* is ambiguous. It can mean the reasons for a claim given in one step (an *Argument*); or it can mean a train of reasoning leading towards a claim (a *Case*), that is, a set of linked *Arguments*; or it can be taken as reasons for and against a claim (a *Debate*), that is a *Case* for the claim and a *Case* against the claim. An additional structure is where the intermediate claims of the *Debate* are also points of dispute, but we will not consider this further here. In the following, we formally define these three senses of *argument* as structures in the argumentation framework, starting with *Arguments*, then providing *Cases*, and finally *Debates*. We provide a graphic, examples, and then definitions for the three different kinds of attack: *Rebuttal*, *Undercut*, and *Premise Defeat*.

We provide a recursive, pointwise definition of a graph which is constructed relative to an AF. Since the sets are constructed relative to an AF, we can infer the attack relations which hold among them. The different senses of *argument* are defined as subgraphs.

Definition 6 *Suppose there is a derived* AF $= \langle \mathcal{L}^A, \mathcal{R}^A \rangle$, *where y and z are arbitrary literals from $\mathcal{L}^A$ and r and r' are arbitrary rules from $\mathcal{L}^A$. $\mathcal{F}$ abbreviates $\{r : r \textit{ added in } \rho_{2k-1}\}$.*

$$
\begin{array}{lcl}
\rho_0(y) & = & \{y, \neg y\} \\
\rho_1(y) & = & \rho_0(y) \cup \bigcup_{\{r:hd(r)=y\}} \{r\} \\
\rho_{2k}(y) & = & \rho_{2k-1}(y) \cup \bigcup_{\{r \in \mathcal{F}\}} \{z, \neg z : z \in bd(r)\} \\
\rho_{2k+1}(y) & = & \rho_{2k}(y) \cup \bigcup_{\{r \in \mathcal{F}\}} \{r' : z \in hd(r') \cap bd(r)\} \\
\rho_{2k+2}(y) & = & \rho_k(y)
\end{array}
$$

$\rho_0(y)$ provides the basis for the construction, which are nodes labeled by literals in an AF that attack one another with respect to the node labeled y. At $\rho_1(y)$, we add to the previous set of rules which have y as their head; depending on whether we have a strict or a defeasible rule, the rule node attacks and may be attacked by the literal which is the negation of the head. At $\rho_{2k}(y)$, we add the positive and negative literals relative to the body of the rules; each of the negative literals associated with literals of the body of the rule attacks the rule node. At $\rho_{2k+1}(y)$, we link rules: the literals in the body of a rule added at $\rho_1(y)$ serve as the heads of other rules. At $\rho_{2k+2}(y)$, we have iterated the steps $\rho_1(y)$-$\rho_{2k+1}(y)$ until there is no further change. Constructions for negations of literals are similarly defined.

Supposing a derived AF, Arg_{S1} and Arg_{S2} are subgraphs of that AF. An *Argument for* y, $Arg_{S1}(y)$, is defined at $\rho_{2k}(y)$: it is the nodes and their attacks defined at this step relative to the derived AF. A graph defined as $Arg_{S1}(y)$ can *only have one rule in the set of nodes*, namely a rule of the Theory Base with y as head (other rules with y as head will give rise to distinct arguments for y in sense 1). In $Arg_{S1}(y)$, y is the *claim* of $Arg_{S1}(y)$ and the literals in the body of the rule are the *premises*. A *Case for* y, $Arg_{S2}(y)$, is defined where $\rho_{k+1}(y) = \rho_k(y)$. $Arg_{S2}(y)$ is comprised of $Arg_{S1}(y)$ along with graphs of form Arg_{S1} for the literals that are bodies of every rule constructed relative to $Arg_{S1}(y)$. In other words, a Case links together all those graphs of Arguments for a particular y where the claim of one rule is the premise of another rule.

Definition 7 *Suppose an* AF *derived from Theory Base $\mathcal{T}$, $\langle \mathcal{L}^A_{\mathcal{T}}, \mathcal{R}^A_{\mathcal{T}} \rangle$. We define* Arg$_{S1}$-Arg$_{S2}$ *as subgraphs of a derived* AF*:*

$An\ Argument\ for\ y\ is\ Arg_{S1}(y) = \langle \mathcal{L}^A_{S_1 y}, \mathcal{R}^A_{S_1 y} \rangle,$
$where\ \mathcal{L}^A_{S_1 y} \subseteq \mathcal{L}^A_{\mathcal{T}}\ and\ \mathcal{R}^A_{S_1 y} \subseteq \mathcal{R}^A_{\mathcal{T}},$
$r,\ r' \in \mathcal{L}^A_{S_1 y}\ r = r',\ is\ a\ subgraph\ at \rho_{2k}(y).$

$A\ Case\ for\ y\ is\ Arg_{S2}(y) = \langle \mathcal{L}^A_{S_2 y}, \mathcal{R}^A_{S_2 y} \rangle,$
$where\ \mathcal{L}^A_{S_2 y} \subseteq \mathcal{L}^A_{\mathcal{T}}\ and\ \mathcal{R}^A_{S_2 y} \subseteq \mathcal{R}^A_{\mathcal{T}},\ is\ a\ subgraph$
$at\ \rho_{k+1}(y) = rho_k(y).$

Where we have Arg_{S2}(y) and Arg_{S2}(¬y), we have a *Single-point Debate about* y, $Arg_{S3}(y)$. The two graphs share only the literals $\{y, \neg y\}$, and no other rules or literals.

Definition 8 *Suppose two derived* AF*s,* $Arg_{S2}(y) = \langle \mathcal{L}^A_{S_2\neg y}, \mathcal{R}^A_{S_2\neg y} \rangle$ *and* $Arg_{S2}(y) = \langle \mathcal{L}^A_{S_2 y},\ \mathcal{R}^A_{S_2 y} \rangle$*:*

$$
\begin{aligned}
&A\ Single - point\ Debate\ about\ y\ is\\
&Arg_{S3}(y)\ =\ \langle \mathcal{L}^A_{S_2 y} \cup \mathcal{L}^A_{S_2\neg y}, \mathcal{R}^A_{S_2 y} \cup \mathcal{R}^A_{S_2\neg y} \rangle,\\
&where\ \mathcal{L}^A_{S_2\neg y} \cap \mathcal{L}^A_{S_2 y}\ =\ \{y, \neg y\}\\
&and\ \mathcal{R}^A_{S_2\neg y} \cap \mathcal{R}^A_{S_2 y}\ =\ \emptyset.
\end{aligned}
$$

Clearly a debate with subsidiary debates can be constructed to argue pro and con about other literals in the base debate; we start with a Arg$_{S2}$(y), then add further *Single-point Debates* about some literal in the graph other than y.

Example 4 shows the senses in a derived AF only with SI rules since they restrict the available preferred extensions.

Example 4 *Suppose a Theory Base comprised of the rules (and related literals):* $r_7 : x_6 \rightarrow \neg x_8$*,* $r_{10} : x_5, x_7 \rightarrow x_8$*,* $r_{11} : \neg x_3, x_4 \rightarrow x_7$*. Figure 4 graphically represents the various senses of* argument *in an* AF *derived from this Theory Base.*

In Figure 4, we have three subgraphs which represent an Argument; each Argument is derived from the corresponding rule of the Theory Base. For example $Arg_{S1}\neg x_8$, the argument for $\neg x_8$, is the graph comprised of nodes $\{\neg x_8, x_8, r_7, \neg x_6, x_6\}$ with the relations among them as given; the graph is derived from the rule of the Theory Base which corresponds to $r_7 : x_6 \rightarrow \neg x_8$. The other two rules of the Theory Base are also represented in the graph as subgraphs that represent an Argument.

Figure 4 presents two Cases. The Case $Arg_{S2}(x_8)$ is derived from the following rules: $r_{10} : x_5, x_7 \rightarrow x_8, r_{11} : \neg x_3, x_4 \rightarrow x_7$. We see how the Arguments in the Case are linked; for instance, the graph of $r_{11} : \neg x_3, x_4 \rightarrow x_7$ has as claim x_7, which is the premise of $r_{10} : x_5, x_7 \rightarrow x_8$. The Case $Arg_{S2}(\neg x_8)$ is derived from the following rule (recall that an Argument can also be a Case): $r_7 : x_6 \rightarrow \neg x_8$.

The Single-point Debate for x_8, $Arg_{S3}(x_8)$, is comprised of the Cases $Arg_{S2}(x_8)$ and $Arg_{S2}(\neg x_8)$.

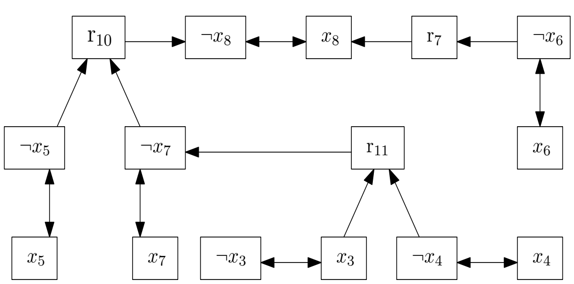

Figure 4: Arguments, Cases, and Single-point Debates

Rebuttal, Premise Defeat, and Undercutting

Given these structures we can express the various familiar notions of attack in different senses of argument: a *Premise Defeat of an Argument* is an Argument with claim that is the negation of the premise of another Argument; a *Rebuttal of an Argument* is an Argument with a claim that is the negation of the claim of another Argument; the *Rebuttal of a Case* is similar to the Rebuttal of an Argument; and an *Undercutter of an Argument* is an attacker of the rule node of an argument.

Definition 9

$A\ Premise\ Defeat\ of\ an\ Argument\ for\ y,$
$Arg_{S1}(y)\ =\ \langle \mathcal{L}^A_{S_1 y},\ \mathcal{R}^A_{S_1 y} \rangle,\ is\ Arg_{S1}(\neg y_i),$
$where\ y_i,\ r \in \mathcal{L}^A_{S_1 y},\ and\ y_i \in bd(r).$

$An\ Undercutter\ of\ a\ Argument\ for\ y,$
$Arg_{S1}(y)\ = \langle \mathcal{L}^A_{S_1 y},\ \mathcal{R}^A_{S_1 y} \rangle,\ is\ Arg_{S1}(\neg y_i),$
$where\ y_i,\ r \in \mathcal{L}^A_{S_1 y},\ and\ y_i \in hd(r).$

$A\ Rebuttal\ of\ an\ Argument\ for\ y,\ Arg_{S1}(y),\ is\ Arg_{S1}(\neg y).$

$A\ Rebuttal\ of\ a\ Case\ for\ y, Arg_{S2}(y), is\ Arg_{S2}(\neg y).$

We refer to *RPU*$_{S2}$*(y)* as the set of rebuttals, premise defeats, and undercutters relative to Arg$_{S2}$(y) (keeping the derivative notions in mind).

Given the definitions of the three senses of "argument" and various notions of attack, it would be possible to define a more abstract, derivative AF in which we represent structures at levels Arg$_{S1}$ or Arg$_{S2}$ as nodes in that AF and which, given the requisite attack relations defined above, are in a higher level attack relation. However, we leave such proposals and the analysis of them for future research.

Assertions

So far we have only TRules in a Single-point Debate. Usually in a Theory Base there are assertions which further restrict the preferred extensions. In our framework, assertions are ARules. A Case such as $Arg_{S2}(y)$ is a directed tree where y is the root and the literals in the bodies of constituent arguments at the level where $\rho_{k+1}(y) = \rho_k(y)$ are the leaves. If all the leaves are strictly asserted and all the rules are strict rules, then the root is sceptically accepted. However, where even just one of the leaves of $Arg_{S2}(y)$ is defeasibly asserted, then the root of the tree is only credulously accepted. Such a Case is vulnerable to attack. Similarly, where there are defeasible rules, the root is always credulously accepted. Where we have *Arg*$_{S2}$(y) and *Arg*$_{S2}$(¬y) for and against a claim and the leaves of each Case are strictly asserted, then the resultant Single-point Debate must be adjudicated according to some principle for choosing between preferred extensions. Matters are complex where we have strict and defeasible assertions of literals at *different levels of the tree*. Further investigation would consider how to adopt some notion of *accrual* in AFs (Prakken 2005) to overcome some of these limitations. However, these matters are beyond the scope of this paper.

Discussion

In this section, we briefly review the key components of the benchmark argument instantiation method of (Caminada and Amgoud 2007), compare it to our proposal, then provide one of the key examples which showed a flaw in the instantiation method as well as motivated the *Rationality Postulates*.

In constructing arguments, they introduce three functions: `Conc` returns the last conclusion of an argument, `Sub` returns all the subarguments of an argument, and `StrictRules` and `DefRules` return all the strict and defeasible rules used in an argument, respectively. Theory Bases $\mathcal{T}$ are comprised of strict and defeasible implications. Arguments have a deductive form and are constructed recursively from the rules of the Theory Base. To distinguish strict or defeasible rules from the deductive form of arguments, we use *short* arrows, $\rightarrow$ and $\Rightarrow$, for the former and *long* arrows, $\longrightarrow$ and $\Longrightarrow$ for the latter. For brevity, we only provide the clauses for the construction of strict arguments as the clauses for the construction of defeasible arguments are analogous (Caminada and Amgoud 2007).

Definition 10 *(Argument) Suppose a* Theory Base*, $\mathcal{T}$, with strict and defeasible rules. An argument* A *is:*
$A_1, \ldots, A_n \longrightarrow \psi$ if $A_1, \ldots, A_n$, with $n \geq 0$, are arguments such that there exists a strict rule Conc(A_1), ..., Conc(A_n) $\rightarrow \psi$.
Conc(A) = ψ,
Sub(A) = Sub(A_1) $\cup \ldots \cup$ Sub(A_n) $\cup \{A\}$,
StrictRules(A) = StrictRules(A_1) $\cup \ldots \cup$ StrictRules(A_n) $\cup \{$Conc(A_1), ..., Conc(A_n) $\rightarrow \psi\}$,
DefRules(A) = DefRules(A_1) $\cup \ldots \cup$ DefRules(A_n).

Consider a Theory Base with strict and defeasible rules from which we construct arguments according to this definition.

Example 5 *Let $\mathcal{T}_4$ be a Theory Base with the following rules:*
r_{21}: $\rightarrow x_1$; r_{22}: $\rightarrow x_2$; r_{23}: $\rightarrow x_3$; r_{24}: $x_4, x_5 \rightarrow \neg x_3$; r_{25}: $x_1 \Rightarrow x_4$; r_{26}: $x_2 \Rightarrow x_5$.
We construct the following arguments:
A_1: $[[\rightarrow x_1] \Rightarrow x_4]$; A_2: $[[\rightarrow x_2] \Rightarrow x_5]$; A_3: $[\rightarrow x_3]$;
A_4: $[\rightarrow x_1]$; A_5: $[\rightarrow x_2]$;
A_6: $[[\rightarrow x_1] \Rightarrow x_4], [[\rightarrow x_2] \Rightarrow x_5] \rightarrow \neg x_3$.

We see clearly that arguments can have subarguments: A_6 has a subargument A_1, and A_1 has a subargument A_4.

Several additional elements are needed to define *justified conclusions*. An argument is strict if it has no defeasible subargument, otherwise it is defeasible (non-strict). An argument A_i rebuts an argument A_j where the conclusion of some subargument of A_i is the negation of the conclusion of some non-strict subargument of A_j; rebuttal is one way an argument defeats another argument. Note that a strict argument can defeat a defeasible argument, but not vice versa. Moreover, one argument can defeat another argument with respect to subarguments; in effect, defeat of a part is inherited as defeat of a whole. With respect to our example, the undefeated arguments are A_1, A_2, A_3, A_4, and A_5. A_3, which is a strict argument, defeats A_6 but not vice versa since A_6 is a non-strict argument in virtue of having a defeasible subargument. Given the arguments and defeat relation between them, we can provide an AF and the different extensions. The `Output` of an AF, understood as the *justified conclusions* of the AF, is given as the sceptically accepted conclusions of the arguments A of the AF.

With respect to the example, (Caminada and Amgoud 2007) claim that the justified conclusions are x_1, x_2, x_3, x_4, and x_5 since these are all conclusions of arguments which are not attacked. However, $\neg x_3$ is *not* a justified conclusion, even though it is the conclusion of a strict rule in which all the premises are justified conclusions. This is so since the argument A_6 of which $\neg x_3$ is the conclusion is defeated by but does not defeat A_3 because A_6 has a *subargument* which is a non-strict argument (namely A_1 or A_2), so making A_6 a non-strict argument, while A_3 is a strict argument. Yet, given the antecedents of the strict rule are justified conclusions, it would seem intuitive that the claim of a strict rule should also be a justified conclusion. This, they claim, shows that justified conclusions are not closed under strict rules or could even be inconsistent.

In our view, these notions of argument and defeat are problematic departures from (Dung 1995), which has no notion of subargument or of defeat in terms of subarguments. In addition, they give rise to the problems with justified conclusions: what is a strict rule in the Theory Base can appear in the AF as a non-strict argument in virtue of subarguments; what cannot be false in the Theory Base without contradiction is defeated in the AF; thus, what "ought" to have been a justified conclusion is not. In addition, the notion of justified conclusion leads to some confusion: on the one hand, it only holds for sceptically accepted arguments, which presumably implies that the propositions which constitute them are sceptically accepted; on the other hand, there is no reason to expect that $\neg x_3$ as sceptically accepted, given that it only follows from defeasible antecedents. Clearly the anomaly arises because of the way that arguments can have defeasible subarguments, that the defeat of the whole can be determined by the defeat of a part, and that justified conclusions depend on these notions.

In our approach, the results are straightforward and without anomaly; we do not make use of arguments with subarguments, inheritance of defeasiblity, or problematic notions of justified conclusions. We consider a key example from (Caminada and Amgoud 2007) as the two other problematic examples in follow suit. The Theory Base of Example 5 appears as in Figure 5, for which the preferred extensions are given, noting that the strict rules do not appear for simplicity while the defeasible rules appear only where not defeated.

$$\{x_1, x_2, x_3, x_4, r_{25}, \neg x_5\},$$
$$\{x_1, x_2, x_3, \neg x_4, x_5, r_{26}\},$$
$$\{x_1, x_2, x_3, \neg x_4, \neg x_5\}$$

Here x_1, x_2, x_3 are all sceptically accepted, while x_4 and x_5 are credulously accepted. $\neg x_3$ is not credulously accepted given that x_3 is strictly asserted. Note that every literal which is strictly asserted is sceptically acceptable. Therefore, the rule node r_{24} must be defeated where one or both of $\neg x_4$ and $\neg x_5$ hold. There is, in our view, no reason to expect $\neg x_3$ to hold in any extension since we have no preferred extension in which both x_4 and x_5 are justified conclusions. Given admissible sets, we satisfy the *consistency* rationality postulate; *closure*, which is relevant only of strict rules where all the body literals hold, is not relevant to this problem.

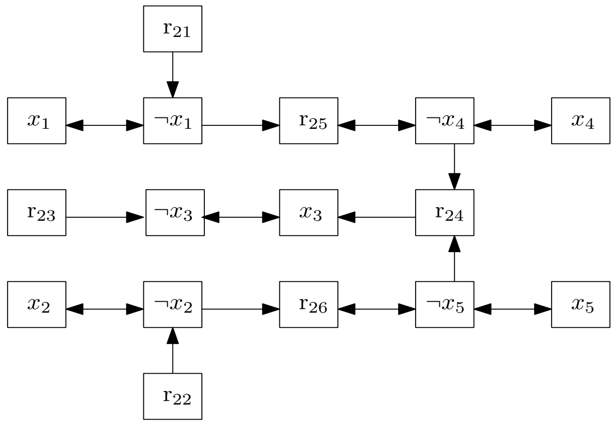

Figure 5: Graph of Problem Example

We have considered a widely adopted approach to instantiating Theory Bases in AFs (Amgoud et al. 2004) along with the problems that arise. There are other approaches not discussed in (Caminada and Amgoud 2007) which may offer alternative ways of avoiding the problems such as Assumption-based (Bondarenko et al. 1997) or Logic-based (Besnard and Hunter 2008) argumentation. We leave further comparison and contrast to future work.

Concluding Remarks

We have presented a method of instantiating a Theory Base which contains strict and defeasible rules in a Dung-style abstract argumentation framework. The Theory Base is directly represented in the framework, and the conclusions of the Theory Base can be computed as extensions of that framework. Our method avoids the logic dependent step of generating arguments from the Theory Base and then organising them in a framework for evaluation. The sceptically acceptable arguments of the framework are the consequences of the Theory Base under classical logic, assuming that the Theory Base is consistent: the consequences under a variety of non-monotonic logics can be identified as credulously acceptable arguments, with different non-monotonic logics corresponding to different ways of choosing between preferred extensions.

The variety of senses of "argument" emerge as structures within the framework, and can be used to explain the consequences. Cases and Debates are defined in terms of this basic structure. This enables us to talk meaningfully about the relations between the different structures of "argument". Thus while other approaches speak of a sub-argument relation, which is not part of normal discourse about argumentation, we can be more precise and use more natural expressions: for example, we can say that two Arguments form part of the same Case, or of independent Cases for a given claim. This in turn helps to clarify the notion of support: we can distinguish between nodes which support each other by forming part of the same Argument, the same Case, or by rebutting a Case for negation of the claim.

We believe that this method provides a very clear way of instantiating Theory Bases as abstract argumentation frameworks. By separating the notion of a node from the ambiguous notion of argument, we have clear criteria for what constitutes a node in the framework. We can explain our reasoning in terms of arguments of the appropriate granularity.

In future work we will demonstrate the formal properties of our approach. In addition, we will further compare and contrast approaches to Theory Base instantiation in AFs. In a different vein, we will explore the potential for improved explanation offered by our distinction between various senses of the term "argument".

Acknowledgements

During earlier drafts, the first author was supported by the ESTRELLA Project (Estrella, IST-2004-027655).

References

Amgoud, L.; Caminada, M.; Cayrol, C.; Lagasquie, M.-C.; and Prakken, H. 2004. Towards a consensual formal model: inference part. Technical report, ASPIC project. Deliverable D2.2: Draft Formal Semantics for Inference and Decision-Making.

Besnard, P., and Hunter, A. 2008. *Elements of Argumentation*. MIT Press.

Bondarenko, A.; Dung, P. M.; Kowalski, R. A.; and Toni, F. 1997. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence* 93:63–101.

Caminada, M., and Amgoud, L. 2007. On the evaluation of argumentation formalisms. *Artificial Intelligence* 171(5-6):286–310.

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

Dunne, P. E., and Bench-Capon, T. J. M. 2002. Coherence in finite argument systems. *Artificial Intelligence* 141(1):187–203.

García, A. J., and Simari, G. R. 2004. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming* 4(1):95–137.

McCarthy, J. 1980. Circumscription - a form of non-monotonic reasoning. *Artificial Intelligence* 13:27–39.

Prakken, H., and Sartor, G. 1997. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics* 7(1).

Prakken, H. 2005. A study of accrual of arguments, with applications to evidential reasoning. In *ICAIL '05: Proceedings of the 10th International Conference on Artificial Intelligence and Law*, 85–94. New York, NY, USA: ACM.

Reiter, R. 1980. A logic for default reasoning. *Artificial Intelligence* 13(1-2):81–132.

Wyner, A.; Bench-Capon, T.; and Atkinson, K. 2008. Three senses of "argument". In Sartor, G.; Casanovas, P.; Rubino, R.; and Casellas, N., eds., *Computable Models of the Law: Languages, Dialogues, Games, Ontologies*, LNAI 4884. Springer. 146–161.