Practical First-order Argumentation

Philippe Besnard
IRIT-CNRS,
Université Paul Sabatier,
118 rte de Narbonne,
31062 Toulouse, France

Anthony Hunter
Department of Computer Science,
University College London,
Gower Street,
London, WC1E 6BT, UK

Abstract

There are many frameworks for modelling argumentation in logic. They include a formal representation of individual arguments and techniques for comparing conflicting arguments. A problem with these proposals is that they do not consider arguments for and against first-order formulae. We present a framework for first-order logic argumentation based on argument trees that provide a way of exhaustively collating arguments and counter-arguments. A difficulty with first-order argumentation is that there may be many arguments and counterarguments even with a relatively small knowledgebase. We propose rationalizing the arguments under consideration with the aim of reducing redundancy and highlighting key points.

Introduction

Argumentation is a vital aspect of intelligent behaviour by humans. Consider diverse professionals such as politicians, journalists, clinicians, scientists, consultants and administrators, who all need to collate and analyse information looking for pros and cons for consequences of importance when attempting to understand problems and make decisions. There are a number of proposals for logic-based formalisations of argumentation (Prakken & Vreeswijk 2002; Chesnevar, Maguitman, & Loui 2001). These proposals allow for the representation of arguments for and against some conclusion, and for attack or undercut relationships between arguments. A key shortcoming of these proposals is that they do not support arguments for/against first-order formulae. Yet in many professional domains, it is apparent that there is a need to support first-order argumentation. As an example, consider a senior clinician in a hospital who may need to consider the pros and cons of a new drug regime in order to decide whether to incorporate the regime as part of hospital policy: This could be expedited by considering the pros and cons of a first-order statement formalizing that piece of policy. As another example, consider an information systems consultant who is collating requirements from users within an organization. Due to conflicts between requirements from different users, the consultant may need to consider arguments for and against particular requirements being adopted in the final requirements specification. Towards this end, first-order statements provide a format for readily and thoroughly capturing constraints and compromises.

It is paramount to notice that using a propositional framework to encode first-order statements leads to mishaps, e.g., when attempting to mimic $\forall x.\alpha[x]$ by means of its instances $\alpha[t]$ for all ground elements t in the universe of discourse: Due to circumstantial properties, it may happen that, whatever t, a particular argument for $\alpha[t]$ can be found but there is no guarantee that an argument for $\forall x.\alpha[x]$ exists. Here is an example. Consider the statements "if x satisfies P and Q then x satisfies R or S"and "if x satisfies Q and R and S then x satisfies T". Clearly, these do not entail the statement "if x satisfies P then x satisfies T". Assume the set of all ground terms from the knowledgebase is $\{a, b\}$. The obvious idea is to consider $\forall x.\alpha[x]$ as being equivalent with both instances $\alpha[a]$ and $\alpha[b]$. Unfortunately, should Qa and $Ra \vee Sa \rightarrow Ta$ be incidentally the case as well as Sb and $Pb \rightarrow Qb \wedge Rb$, then "if x satisfies P then x satisfies T" would be regarded as argued for! The moral is that a propositional approach here cannot be substituted for a first-order one. In such situations, a first-order approach cannot be dispensed with.

To address this need, we present a framework for first-order argumentation. As the first-order case raises the issue of efficiency, both representational and computational, the question of redundancy becomes important. Given even a small set of first-order formulae, the number of arguments that can be generated can be large. However, this can be rendered manageable if we can rationalise the arguments we need to consider. Most proposals for argumentation do not consider such issues. An exception is (Besnard & Hunter 2001) which we will take as the basis for defining our framework for first-order argumentation.

First-order Argumentation

We now extend to the first-order case an existing proposal for logic-based argumentation (Besnard & Hunter 2001) that is based on classical propositional logic. For a first-order language $\mathcal{L}$, the set of formulae that can be formed is given by the usual inductive definitions for classical logic. Roman symbols $P, Q, \ldots$ denote predicates, Greek symbols $\alpha, \beta, \ldots$ denote formulas. Deduction in classical logic is denoted by the symbol $\vdash$ and deductive closure by Cn so that $\mathsf{Cn}(\Phi) = \{\alpha \mid \Phi \vdash \alpha\}$.

For the following definitions, we first assume a knowledgebase Δ (a finite set of formulae) and use this Δ throughout (i.e., except where indicated, everything is parameterized with Δ). A further requirement, which really is not demanding, is as follows: We also assume that every subset of Δ is given an enumeration $\langle\alpha_1,\ldots,\alpha_n\rangle$ of its elements, which we call its canonical enumeration.

The paradigm for the approach is a large repository of information, represented by Δ, from which arguments can be constructed for and against arbitrary claims. Apart from information being understood as declarative statements, there is no a priori restriction on the contents, and the pieces of information in the repository can be as complex as needed. Therefore, Δ is not expected to be consistent. It need even not be the case that every single formula in Δ is consistent.

The framework adopts a very common intuitive notion of an argument. Essentially, an argument is a set of relevant formulae that can be used to classically prove some point, together with that point. A point is represented by a formula.

Definition 1. *An* **argument** *is a pair $\langle\Phi,\alpha\rangle$ such that: (1) $\Phi\nvdash\bot$; (2) $\Phi\vdash\alpha$; and (3) no $\Phi'\subset\Phi$ satisfies $\Phi'\vdash\alpha$. We say that $\langle\Phi,\alpha\rangle$ is an argument for α. We call α the consequent of the argument and Φ the support of the argument:* $\mathsf{Support}(\langle\Phi,\alpha\rangle)=\Phi$, *and* $\mathsf{Consequent}(\langle\Phi,\alpha\rangle)=\alpha$.

Example 1. *Let $\Delta=\{\forall x.(Px\rightarrow Qx\vee Rx), Pa, \neg\forall x.Sx, \neg\exists x.Rx, \neg\exists x.(Px\rightarrow Qx\vee Rx)\}$. Some arguments are:*

$$\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx)\}, Qa\vee Ra\rangle$$
$$\langle\{\neg\forall x.Sx\},\neg\forall x.Sx\rangle$$
$$\langle\{\neg\exists x.Rx\},\forall x.\neg Rx\rangle$$

Arguments are not independent. Some encompass others (possibly up to some form of equivalence) in a sense:

Definition 2. *An argument $\langle\Phi,\alpha\rangle$ is* **more conservative** *than an argument $\langle\Psi,\beta\rangle$ iff $\Phi\subseteq\Psi$ and $\beta\vdash\alpha$.*

Example 2. *$\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx)\}, Qa\vee Ra\rangle$ is a more conservative argument than $\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx),\neg\exists x.Rx\}, Qa\rangle$.*

Some arguments directly oppose the support of others, which amounts to the notion of an undercut.

Definition 3. *An* **undercut** *for an argument $\langle\Phi,\alpha\rangle$ is an argument $\langle\Psi,\neg(\phi_1\wedge\ldots\wedge\phi_n)\rangle$ where $\{\phi_1,\ldots,\phi_n\}\subseteq\Phi$.*

Example 3. *It is easy to find an undercut for the argument $\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx)\}, Qa\vee Ra\rangle$, an obvious one is: $\langle\{\neg\exists x.(Px\rightarrow Qx\vee Rx)\},\neg\forall x.(Px\rightarrow Qx\vee Rx)\rangle$. Now, there is another one, which actually is more conservative: $\langle\{\neg\exists x.(Px\rightarrow Qx\vee Rx)\},\neg(Pa\wedge\forall x.(Px\rightarrow Qx\vee Rx))\rangle$*

Example 4. *Provided the conditions for Definition 1 are met, we have the general cases below:*

$\langle\{\forall x.\alpha[x]\},\alpha[a]\rangle$ *is undercut by* $\langle\{\neg\exists x.\alpha[x]\},\neg\forall x.\alpha[x]\rangle$
$\langle\{\forall x.\alpha[x]\},\alpha[a]\rangle$ *is undercut by* $\langle\{\exists x.\neg\alpha[x]\},\neg\forall x.\alpha[x]\rangle$
$\langle\{\forall x.\alpha[x]\},\alpha[a]\rangle$ *is undercut by* $\langle\{\neg\alpha[b]\},\neg\forall x.\alpha[x]\rangle$
$\langle\{\forall x.\alpha[x]\},\alpha[a]\rangle$ *is undercut by* $\langle\{\neg\alpha[c]\},\neg\forall x.\alpha[x]\rangle$

Definition 4. *$\langle\Psi,\beta\rangle$ is a* **maximally conservative undercut** *of $\langle\Phi,\alpha\rangle$ iff $\langle\Psi,\beta\rangle$ is an undercut of $\langle\Phi,\alpha\rangle$ such that for all undercuts $\langle\Psi',\beta'\rangle$ of $\langle\Phi,\alpha\rangle$, if $\Psi'\subseteq\Psi$ and $\beta\vdash\beta'$ then $\Psi\subseteq\Psi'$ and $\beta'\vdash\beta$.*

Example 5. *A maximally conservative undercut for $\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx)\}, Qa\vee Ra\rangle$ is $\langle\{\neg\exists x.(Px\rightarrow Qx\vee Rx)\},\neg(Pa\wedge\forall x.(Px\rightarrow Qx\vee Rx))\rangle$.*

The value of the following definition of canonical undercut is that we only need to take the canonical undercuts into account. This means we can justifiably ignore the potentially very large number of non-canonical undercuts.

Definition 5. *An argument $\langle\Psi,\neg(\phi_1\wedge\ldots\wedge\phi_n)\rangle$ is a* **canonical undercut** *for $\langle\Phi,\alpha\rangle$ iff it is a maximally conservative undercut for $\langle\Phi,\alpha\rangle$ and $\langle\phi_1,\ldots,\phi_n\rangle$ is the canonical enumeration of Φ.*

Proposition 1. *Given two canonical undercuts for the same argument, none is more conservative than the other.*

Proposition 2. *Any two different canonical undercuts for the same argument have distinct supports whereas they do have the same consequent.*

An argument tree describes various ways an argument can be challenged, how the counterarguments to the initial argument can themselves be challenged, and so on recursively.

Definition 6. *An* **argument tree** *for α is a tree where the nodes are arguments such that*

1. *The root is an argument for α.*
2. *For no node $\langle\Phi,\beta\rangle$ with ancestor nodes $\langle\Phi_1,\beta_1\rangle,\ldots,\langle\Phi_n,\beta_n\rangle$ is Φ a subset of $\Phi_1\cup\cdots\cup\Phi_n$.*
3. *Each child of a node A is an undercut of A that obeys 2.*

A **canonical argument tree** *is an argument tree where each undercut is a canonical undercut. A* **complete argument tree** *is a canonical argument tree for each node A, such that if A' is a canonical undercut for A, then A' is a child of A.*

The second condition in Definition 6 (Besnard & Hunter 2001) (see (García & Simari 2004) as well) ensures that each argument on a branch introduces at least one formula in its support that has not already been used by ancestor arguments. I.e., each argument on a branch conveys something new, not just turning around (some of) its ancestor arguments. As a notational convenience, the $\diamond$ symbol denotes the consequent of an argument when that argument is a canonical undercut.

Example 6. *A complete argument tree for Qa is:*

$$\langle\{Pa,\forall x.(Px\rightarrow Qx\vee Rx),\neg\exists x.Rx\}, Qa\rangle$$
$$\uparrow$$
$$\langle\{\neg\exists x.(Px\rightarrow Qx\vee Rx)\},\diamond\rangle$$

A complete argument tree is an efficient representation of the counterarguments, counter-counterarguments, ...

Proposition 3. *Let $\alpha\in\mathcal{L}$. If Δ is finite, there is a finite number of argument trees with the root being an argument with consequent α that can be formed from Δ, and each of these trees has finite branching and a finite depth.*

For an argument tree T, $\mathsf{Root}(T)$, $\mathsf{Nodes}(T)$, $\mathsf{Width}(T)$, $\mathsf{Depth}(T)$, are as usual for a tree. For an argument A_i in an argument tree T, $\mathsf{Undercuts}(T,A_i)$ denotes the set of children of A_i. Let $\mathsf{Siblings}(T)$ be the set of sibling sets in T, i.e. $S\in\mathsf{Siblings}(T)$ iff $S=\mathsf{Undercuts}(T,A_i)$ for some A_i in T.

Example 7. *For the argument tree T in Example 9,* $\mathsf{Siblings}(T) = \{\{A_0\}, \{A_1\}, \{A_2\}, \{A_3, A_4\}, \{A_5\}\}$.

As for notation, we write Ω for the set of all arguments.

Degree of Undercut

An argument conflicts with each of its undercuts, by the very definition of an undercut. Now, some may conflict more than others, and some may conflict a little while others conflict a lot: Conflict of an argument with each of its undercuts is reflected by a position in an ordering (possibly a partial one) but not necessarily a numerical value in some interval (i.e., orders of magnitude are not necessarily needed).

Definition 7. *A* **degree of undercut** *is a mapping* $\mathsf{Degree} : \Omega \times \Omega \to O$ *where* $\langle O, \leq \rangle$ *is some poset such that for* $A_i = \langle \Phi, \alpha \rangle$ *and* $A_j = \langle \Psi, \beta \rangle$ *in* Ω,
– $\mathsf{Degree}(A_j, A) \leq \mathsf{Degree}(A_i, A)$ *for all* $A \in \Omega$ *if* $\Phi \vdash \Psi$.
– $\mathsf{Degree}(A_i, A_j)$ *is minimal iff* $\Phi \cup \Psi \not\vdash \bot$.

The last clause in Definition 7 means that $\mathsf{Degree}(A, A')$ is minimal when A and A' are two arguments which do not conflict with each other (so, none is an undercut of the other, as Degree is rather a degree of conflict but it is called degree of undercut here because we are only interested in its value when A' is an undercut of A). Definition 7 allows for many possibilities, leaving you to choose a suitable mapping. In the rest of the paper, we assume that O is the interval $[0, 1]$.

We now introduce labelled argument trees. I.e., we label each arc with the degree of undercut.

Definition 8. *A* **labelled argument tree** *is an argument tree such that if A_j is a child of A_i in the argument tree, then the arc from A_j to A_i is labelled with* $\mathsf{Degree}(A_i, A_j)$.

Example 8. *A labelled argument tree for $\forall x.\alpha[x]$ is:*

$$\begin{array}{ccc} & \langle\{\forall x.\alpha[x]\}, \forall x.\alpha[x]\rangle & \\ \nearrow 1/n & & \nwarrow m/n \\ \langle\{\neg\alpha[a]\}, \diamond\rangle & & \langle\{\neg\alpha[b_1] \wedge \ldots \wedge \neg\alpha[b_m]\}, \diamond\rangle \end{array}$$

From now on, n is some reasonable upper bound for the size of the universe of discourse (it is supposed to be finite).

It may look from Example 8 that labels for arcs to a node $\langle\{\forall x.\alpha[x]\}, \diamond\rangle$ are easy to assess as numbers in $[0, 1]$:

undercut	*label*
$\langle\{\neg\alpha[a]\}, \diamond\rangle$	$1/n$
$\langle\{\neg\alpha[b] \wedge \neg\alpha[c]\}, \diamond\rangle$	$2/n$
$\vdots$	
$\langle\{\forall x.\neg\alpha[x]\}, \diamond\rangle$	1

However, some cases are less obvious:

undercut	*label*
$\langle\{\neg\alpha[a] \vee \neg\alpha[b]\}, \diamond\rangle$	$1/n$
$\vdots$	
$\langle\{\exists x.\neg\alpha[x]\}, \diamond\rangle$	$1/n$

Turning to the undercuts for $\langle\{\forall x.\forall y.\alpha[x, y]\}, \diamond\rangle$ gives:

undercut	*label*
$\langle\{\exists x.\exists y.\neg\alpha[x, y]\}, \diamond\rangle$	$1/n^2$
$\langle\{\neg\alpha[c, d]\}, \diamond\rangle$	$1/n^2$
$\langle\{\forall y.\neg\alpha[a, y]\}, \diamond\rangle$	$1/n$
$\langle\{\forall x.\neg\alpha[x, b]\}, \diamond\rangle$	$1/n$
$\langle\{\forall x.\exists y.\neg\alpha[x, y]\}, \diamond\rangle$	$1/n$
$\langle\{\exists x.\forall y.\neg\alpha[x, y]\}, \diamond\rangle$	$1/n$
$\langle\{\forall x.\neg\alpha[x, x]\}, \diamond\rangle$	$1/n$
$\vdots$	
$\langle\{\forall x.\forall y.\neg\alpha[x, y]\}, \diamond\rangle$	1

In general, $\langle\{\forall x_1 \ldots \forall x_p.\alpha[x_1 \ldots x_p]\}, \diamond\rangle$ is undercut by $\langle(\forall^*\exists^*)^*.\theta, \diamond\rangle$ such that θ is in α-DNF:

$$\theta = \bigvee_{i=1..q} (\neg\alpha[t_{i11} \ldots t_{i1p}] \wedge \ldots \wedge \neg\alpha[t_{im_i1} \ldots t_{im_ip}])$$

Then, the label is

$$\min_{i=1..q} \sum_{j=1..m_i} n^{val(\neg\alpha[t_{ij1} \ldots t_{ijp}]) - p}$$

where $val(\neg\alpha[t_{ij1} \ldots t_{ijp}])$ is the number of (different) universally quantified variables among $t_{ij1} \ldots t_{ijp}$.

A labelled argument tree provides extra information that leads to a useful abstraction of the original argument tree.

Example 9. *Provided $A_0, A_1, A_2, \ldots, A_5$ as well as k, m, n (where $k < n$ and $m < n$) conform with Definition 8, here is a labelled argument tree in abstract form:*

$$\begin{array}{ccccc} & & A_0 & & \\ & 1/n \nearrow & & \nwarrow m/n & \\ & A_1 & & A_2 & \\ 1/n \nearrow & & \nwarrow 1/n & \uparrow k/n & \\ A_3 & & A_4 & A_5 & \end{array}$$

In this labelled argument tree, if n is significantly greater than 1, then it may be natural to concentrate our attention on the rightmost branch of the tree, since if, in addition, m is close to n, then A_2 is an important undercut of A_0.

The tension of an argument tree is the cumulative conflict obtained from all the undercuts in the tree.

Definition 9. *Let T be an argument tree, and let A_r be the root node. The degree of tension in T, denoted* $\mathsf{Tension}(T)$, *is given by the value of* $\mathsf{Retension}(A_r)$, *where for any node A_i in the tree, if A_i is a leaf, then* $\mathsf{Retension}(A_i) = 0$ *otherwise* $\mathsf{Retension}(A_i)$ *is*

$$\sum_{A_j \in \mathsf{Undercuts}(A_i)} \mathsf{Retension}(A_j) + \mathsf{Degree}(A_i, A_j)$$

Here, measuring conflicts between arguments and undercuts requires orders of magnitude. In fact, it must now be assumed that the poset $\langle O, \leq \rangle$ comes with an additive measure (written $+$ in the above definition).

Tension provides a useful way to refine the value given by the degree of undercut: The latter merely indicates that an undercut with degree $3k/n$ is three times more important than an undercut with degree k/n but this may need to be reconsidered in view of the overall tension (e.g., if huge).

By Definition 9, tension is fully determined once degree of undercut is specified. Tension is nonetheless a useful complement because Definition 7 is fairly liberal, as already mentioned. Indeed, Definition 7 is rather meant to act as guidelines and there are many possibilities in choosing Degree hence Degree need not be definable from the non-labelled argument tree: Degree may convey extraneous information (e.g., a certain undercut gets more weight because it concerns a certain topic, or it comes from a certain source, or ...). In other words, Degree and Tension may be tailored to the need of a particular application.

As an illustration, if the knowledgebase induces a finite skolemization, the degree of undercut for $\langle \Phi, \alpha \rangle$ with respect to $\langle \Psi, \beta \rangle$ could be set to be the minimum of the Dalal distance (Dalal 1988) between a model of Φ and a model of Ψ. If desired, it would be possible to normalize with a factor of $1/N$ for N being the cardinal of the Herband universe resulting from Skolemization.

Labelled complete arguments trees are the ones of interest in that they contain all the information but some are needlessly too large: The remaining sections aim at getting rid of redundant and/or negligible information in argument trees.

Rationalizing Argument Trees

Even for small first-order knowledgebases, the number of arguments can be overwhelming. So, we propose rationalization of argument trees including (1) Pruning arguments, e.g. that have a degree of undercut that is below a certain threshold; and (2) Merging arguments to create fewer undercuts but without losing vital information. Below we will consider compressing and condensing as types of merging. Rationalization is part of a process of editing a set of arguments and counterarguments to allow focusing on key issues.

Label-based pruning of argument trees

For pruning, we introduce a threshold for a minimum degree of undercut. If an undercut has a degree of undercut below the threshold, then the undercut is dropped, together with any offspring of that undercut.

Definition 10. *A threshold, denoted τ, is a value in O such that if T is an argument tree,* $\mathsf{Prune}(T, \tau)$ *is the* **pruned argument tree** *obtained from T by removing each undercut A_j of an argument A_i if* $\mathsf{Degree}(A_i, A_j) \leq \tau$ *and for any undercut removed, all the offspring of the undercut are also removed.*

Example 10. *Let T be the following labelled argument tree.*

$$
\begin{array}{ccccc}
 & & A_1 & & \\
\nearrow 80/100 & & & \nwarrow 1/100 & \\
A_2 & & & A_3 & \\
\uparrow 40/100 & & \nearrow 10/100 & & \nwarrow 75/100 \\
A_4 & & A_5 & & A_6
\end{array}
$$

Below, the left argument tree is $\mathsf{Prune}(T, 0.3)$ *and the right one is* $\mathsf{Prune}(T, 0.5)$.

$$
\begin{array}{lcl}
A_1 & \quad & A_1 \\
\uparrow 80/100 & & \uparrow 80/100 \\
A_2 & & A_2 \\
\uparrow 40/100 & & \\
A_4 & &
\end{array}
$$

So pruning of argument trees allows us to focus our attention on the most important undercuts.

Proposition 4. *For all $i \in O$, if $T' =$* $\mathsf{Prune}(T, i)$ *then* $\mathsf{Tension}(T') \leq \mathsf{Tension}(T)$, $|\mathsf{Nodes}(T')| \leq |\mathsf{Nodes}(T)|$, $\mathsf{Depth}(T') \leq \mathsf{Depth}(T)$, *and* $\mathsf{Width}(T') \leq \mathsf{Width}(T)$.

Also, $\mathsf{Prune}(T, 0) = T$ and $\mathsf{Prune}(T, 1)$ returns the root of T. For all $i \in [0, 1]$, if T is a canonical argument tree, then $\mathsf{Prune}(T, i)$ is a canonical argument tree. However, if T is a complete argument tree, then $\mathsf{Prune}(T, i)$ is not necessarily a complete argument tree.

One may wonder about how justified it is to consider downsized argument trees, e.g. pruned trees. In order to discuss this issue, we need a couple of notions. In an argument tree T, an *attacker* of a node N is any node N' of which N is an ancestor and is such that the path from N to N' has a length n where n is odd (e.g., any child of N is an attacker of N). In an argument tree T, a *defender* of a node N is any attacker of a child of N.

It seems right to ignore the offspring even if it goes above the threshold: If the offspring with large degree is an attacker of U the undercut to be removed, then U is weaker and this is a further reason not to consider it. If the offspring is a defender of U, then that defender anyway fails to make U stronger than it would if U was not attacked at all, in which case U be removed anyway (so it is coherent that U is removed).

Label-free pruning of argument trees

However, before considering degrees of undercuts, there is a way to ignore some weak subarguments without even taking into account any quantitative value about their disagreement with the undercut argument.

Definition 11. *Let $\langle \Psi, \beta \rangle$ be an undercut of $\langle \Phi, \alpha \rangle$. Then, $\langle \Psi, \beta \rangle$ is an* **overzealous undercut** *of $\langle \Phi, \alpha \rangle$ whenever $\Psi \vdash \alpha$.*

The idea can be illustrated as follows: Andrew notices, e.g., "Hume is not hard to understand and the new BMW is a 6-cylinder and the new BMW has ABS and the new BMW has injection" so he claims "the new BMW is a luxury car". Basil notices "Hume is hard to understand and ... here are some impressive facts about the new BMW ... and the new BMW has injection" so he contradicts Andrew by pointing out "your support is wrong". Certainly Basil's counterargument is weak, plagued with irrelevance (he contradicts something apart from the main subject of the new BMW).

There is a parallel to be drawn here: Remember that the fallacy of argumenting ad hominem consists of attacking the person who puts the argument forward, instead of attacking the argument itself. Consider the (usual) situation in which the promoter Phil of an argument A regards himself/herself to be a good person. Formally, the statement (of course, the example neither claims nor disclaims that Phil —whoever we are talking about— is a good person!) that Phil is a good person becomes part of the support as a conjunct. Now, argumenting ad hominem amounts to denigrating the consequent of A just by disparaging Phil. Formally, this gives rise to an argument A' whose consequent denies that Phil is a good person.

Overzealous undercuts should be deleted, turning an argument tree T into a "focussed" argument tree denoted $\mathsf{Zealousfree}(T)$.

Proposition 5. *If $T' = \mathsf{Zealousfree}(T)$ then $\mathsf{Tension}(T') \leq \mathsf{Tension}(T)$, $|\mathsf{Nodes}(T')| \leq |\mathsf{Nodes}(T)|$, $\mathsf{Depth}(T') \leq \mathsf{Depth}(T)$, and $\mathsf{Width}(T') \leq \mathsf{Width}(T)$.*

Observe that an argument need not have an overzealous undercut (Δ being fixed) but if it has one then at least one of its canonical undercuts is overzealous.

Compressing argument trees

Compression combines arguments without loss of essential information. Compression merges siblings in order to reduce the number of arguments and to reduce the redundancy arising by having arguments equivalent in some sense, and to make appropriate simplifications of the syntax of some arguments.

Definition 12. *Let T_1 and T_2 be argument trees. T_2 is a* **compression** *of T_1 iff there is a surjection $G : \mathsf{Nodes}(T_1) \to \mathsf{Nodes}(T_2)$ such that for all $B \in \mathsf{Nodes}(T_2)$,*

$$\mathsf{Cn}(\mathsf{Support}(B)) = \mathsf{Cn}(\bigcup_{A \in G^{-1}(B)} \mathsf{Support}(A))$$

We call G the compression function.

Example 11. *T_3 is a compression of T_2:*

$$T_2: \quad \langle\{\forall x.Px\}, \forall x.Px\rangle \;\nwarrow\; \langle \neg Pa \vee \neg Pb, \diamond\rangle, \;\; \langle \neg Pa \wedge \neg Pb, \diamond\rangle$$

$$T_3: \quad \langle\{\forall x.Px\}, \forall x.Px\rangle \;\uparrow\; \langle \neg Pa \wedge \neg Pb, \diamond\rangle$$

while each of T_2 and T_3 is a compression of T_1:

$$T_1: \quad \langle\{\forall x.Px\}, \forall x.Px\rangle \;\nwarrow\; \langle \neg Pa \vee \neg Pb, \diamond\rangle, \;\; \langle \neg Pa, \diamond\rangle, \;\; \langle \neg Pb, \diamond\rangle, \;\; \langle \neg Pa \wedge \neg Pb, \diamond\rangle$$

Compression does not affect the depth of the tree, but it has a downward effect on the branching.

Proposition 6. *If T_2 is a compression of T_1, then $\mathsf{Tension}(T_1) \leq \mathsf{Tension}(T_2)$, $\mathsf{Depth}(T_1) = \mathsf{Depth}(T_2)$, $|\mathsf{Nodes}(T_2)| \leq |\mathsf{Nodes}(T_1)|$, and $\mathsf{Width}(T_2) \leq \mathsf{Width}(T_1)$.*

Compression is not unique, and there are limits to compression, for example when an argument tree is a chain, and when all pairs of siblings have supports that are mutually contradictory.

Undercutting is preserved in compression. The following proposition explains the nature of offspring when some siblings are merged in a tree compression.

Proposition 7. *Let T_1 and T_2 be argument trees. If T_2 is a compression of T_1, with compression function G, and $A \in \mathsf{Nodes}(T_1)$, and $G(A) = B$, and $C \in \mathsf{Undercuts}(T_1, A)$ then $G(C) \in \mathsf{Undercuts}(T_2, B)$.*

Proposition 8. *Let T_1 and T_2 be argument trees. (1) If T_2 is a compression of T_1 and T_1 is a compression of T_2 then T_1 and T_2 are isomorphic (the compression function is a bijection); (2) If T_3 is a compression of T_2 and T_2 is a compression of T_1 then T_3 is a compression T_1; and (3) T_1 is a compression of T_1. So, "is a compression of" is a partial ordering relation over non-isomorphic argument trees.*

Proposition 9. *Let T_1 be an argument tree. If A_1 and A_2 are siblings in T_1, and $\mathsf{Cn}(\mathsf{Support}(A_1)) = \mathsf{Cn}(\mathsf{Support}(A_2))$, then there is an argument tree T_2 such that T_2 is a compression of T_1, with compression function G, and $G(A_1) = G(A_2)$.*

Compressing a canonical argument tree always yields a canonical argument tree. In contrast, compressing need not turn a complete argument tree into a complete argument tree. Actually, proper compressing always yields an incomplete argument tree. How well then does compressing argument trees meet our needs? In general, an incomplete argument tree is not a reliable account of the counterarguments (and counter-counterarguments, ...) of an argument. However, starting with an exhaustive account, namely a complete argument tree, compressing only discards redundant information: The resulting incomplete argument tree can still be viewed as exhaustive.

Condensing argument trees

A condensed argument tree is obtained by editing the set of assumptions and then building a new argument tree using the edited set of assumptions rather than editing the tree directly. A requirement of condensation is that the original and the condensed argument trees are complete argument trees.

Definition 13. *Let T_1 be a complete argument tree from Δ_1 and let T_2 be a complete argument tree from Δ_2. T_2 is a* **condensation** *of T_1 iff either (1) there is T_3 such that T_2 is condensation of T_3 and T_3 is a condensation of T_1; or (2) there exists $S_1 \in \mathsf{Siblings}(T_1)$ and there exists $S_2 \in \mathsf{Siblings}(T_2)$ where $|S_2| \leq |S_1|$ and the following two conditions hold:*

- $\Delta_1 - \bigcup_{A \in S_1} \mathsf{Support}(A) = \Delta_2 - \bigcup_{A' \in S_2} \mathsf{Support}(A')$
- $\mathsf{Cn}(\bigcup_{A \in S_1} \mathsf{Support}(A)) = \mathsf{Cn}(\bigcup_{A' \in S_2} \mathsf{Support}(A')) \subset \mathcal{L}$

The requirement $\subset \mathcal{L}$ in Definition 13 rules out cases such as $S_1 = \{\langle\{Pa \wedge \alpha_1\}, \diamond\rangle, \langle\{Pa \wedge \alpha_2\}, \diamond\rangle, \langle\{Pa \wedge \alpha_3\}, \diamond\rangle\}$, $S_2 = \{\langle\{Pa \wedge \beta\}, \diamond\rangle, \langle\{Pa \wedge \neg\beta\}, \diamond\rangle\}$ with $\alpha_1 \wedge \alpha_2 \wedge \alpha_3$ inconsistent so S_2 may have nothing to do with S_1 bar Pa.

Condensed argument trees are canonical argument trees. A condensed argument tree is obtained by replacing a set of siblings with a new set of siblings and then adding all the canonical undercuts appropriate for these new arguments.

Example 12. *Consider the argument below.*

$$\begin{aligned}
A_1 &= \langle\{\forall x.Px\}, \forall x.Px\rangle \\
A_2 &= \langle\{\neg Pa \wedge Qa\}, \diamond\rangle \\
A_3 &= \langle\{\neg Pa \wedge Ra\}, \diamond\rangle \\
A_4 &= \langle\{\neg Qa \vee \neg Ra, Ra\}, \diamond\rangle \\
A_5 &= \langle\{\neg Qa \vee \neg Ra, Qa\}, \diamond\rangle \\
A_6 &= \langle\{\neg Pa \wedge Qa \wedge Ra\}, \diamond\rangle \\
A_7 &= \langle\{\neg Qa \vee \neg Ra\}, \diamond\rangle \\
A_8 &= \langle\{Qa, Ra\}, \diamond\rangle
\end{aligned}$$

$\Delta_1 = \{\forall x Px, \neg Pa \wedge Qa, \neg Pa \wedge Ra, \neg Qa \vee \neg Ra, Qa, Ra\}$
$\Delta_2 = \{\forall x Px, \neg Pa \wedge Qa \wedge Ra, \neg Qa \vee \neg Ra, Qa, Ra\}$
T_2 is a condensation of T_1 but T_2 isn't a compression of T_1 ($\mathsf{Support}(A_6) \not\subseteq \Delta_1$ hence T_2 isn't an argument tree wrt Δ_1)

$$
\begin{array}{ccc}
 & T_1 & \\
 & & \\
 & A_1 & \\
\nearrow & & \nwarrow \\
A_2 & & A_3 \\
\uparrow & & \uparrow \\
A_4 & & A_5
\end{array}
\qquad\qquad
\begin{array}{c}
T_2 \\
 \\
A_1 \\
\uparrow \\
A_6 \\
\uparrow \\
A_7 \\
\uparrow \\
A_8
\end{array}
$$

$\Delta_3 = \{\neg Pa \vee \neg Qa, \neg Pa \wedge Qa \wedge Ra\} \cup \Delta_1 \setminus \{\neg Pa \wedge Qa\}$.
Let $A'_2 = \langle\{\neg Pa \vee \neg Qa, Qa\}, \diamond\rangle$.
T_4 is a compression of T_3 but T_4 isn't a condensation of T_3 ($S_3 = \{A'_2, A_3, A_6\}$ and $S_4 = \{A_6\}$, so Definition 13 fails $Qa \notin \Delta_3 - \bigcup_{A \in S_3} \mathsf{Support}(A)$ due to $Qa \in \mathsf{Support}(A'_2)$ $Qa \in \Delta_4 - \bigcup_{A \in S_4} \mathsf{Support}(A)$ as $Qa \notin \bigcup_{A \in S_4} \mathsf{Support}(A)$ whereas $Qa \in \Delta_4$ due to $Qa \in \mathsf{Support}(A_8)$ and $A_8 \in T_4$)

$$
\begin{array}{ccc}
 & T_3 & \\
 & & \\
 & A_1 & \\
\nearrow & \uparrow & \nwarrow \\
A'_2 & A_3 & A_6 \\
\uparrow & \uparrow & \uparrow \\
A_4 & A_5 & A_7 \\
 & & \uparrow \\
 & & A_8
\end{array}
\qquad\qquad
\begin{array}{c}
T_4 \\
 \\
A_1 \\
\uparrow \\
A_6 \\
\uparrow \\
A_7 \\
\uparrow \\
A_8
\end{array}
$$

It can happen that $\mathsf{Depth}(T_2) > \mathsf{Depth}(T_1)$ even though T_2 is a condensation of T_1 (see Example 12). Of course, it can also happen that $\mathsf{Depth}(T_2) < \mathsf{Depth}(T_1)$ where T_2 is a condensation of T_1.

Proposition 10. *The "is a condensation of" relation is a partial ordering relation.*

As with compression, condensation is not unique, and there are limits to condensation, for example when an argument tree is a chain, and when all pairs of siblings have supports that are mutually contradictory.

An interesting issue is the interplay between all the above rationalizing techniques. The first comment is that presumably all pruning should be considered before any compression or condensation. The question of the order between label-free and label-based pruning is not obvious. They seem largely independent and maybe the order in which they are performed is indifferent. In particular, they do not alter anything in the remaining subtree, unlike compression and condensation. Turning to these two rationalizing techniques, one observation which seems in order is that compression preserves the original data whereas condensation moves to a new knowledgebase —a radical shift. So, it looks like all compression should rather take place before any condensation step, relieving us from the question of whether compression and condensation "commute" to define a confluence property à la Church-Rosser.

Conclusion

The primary aim of this paper has been to extend logic-based proposals for argumentation with techniques for first-order argumentation. The framework could be implemented as a decision support system to help professionals analyse conflicting first-order information. In particular, argument-based decision support could be a valuable solution to handling inconsistencies arising in requirements engineering (Finkelstein *et al.* 1994) and in security policy development (Benferhat & El Baida 2004).

Rationalizing argument trees is an important step towards making argumentation a more manageable process for practical decision support tasks. The techniques can straightforwardly be adapted for a wide range of logic-based proposals for argumentation, for example (Amgoud & Cayrol 2002).

Compression and condensation can allow undercuts to coalesce. E.g., if the root is a general statement that is being proposed, say $\forall x.(\alpha(x) \rightarrow \beta(x))$, then an undercut may be of the form $\alpha(a_1) \wedge \neg\beta(a_1)$. But the degree of undercut is based on just one conflict. If there are many cases $a_1, .., a_k$ such that for each $i \in \{1, .., k\}$, we have $\alpha(a_i) \wedge \neg\beta(a_i)$, then these may coalesce to form a much stronger argument (i.e., an argument with a higher degree of undercut).

The framework also offers: Some ways of defining equivalence relations over argument trees; A way of determining that one argument tree is more conflicting than another; And a way of determining that an argument tree is more informative (i.e., it has inferentially stronger support) than another.

References

Amgoud, L., and Cayrol, C. 2002. A model of reasoning based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence* 34:197–216.

Benferhat, S., and El Baida, R. 2004. Handling conflicts in access control models. In *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-2004)*, 961–962.

Besnard, P., and Hunter, A. 2001. A logic-based theory of deductive arguments. *Artificial Intelligence* 128:203–235.

Chesnevar, C.; Maguitman, A.; and Loui, R. 2001. Logical models of argument. *ACM Computing Surveys* 32:337–383.

Dalal, M. 1988. Investigations into a theory of knowledge base revision: Preliminary report. In *Proceedings of the 7th National Conference on Artificial Intelligence (AAAI'88)*, 3–7.

Finkelstein, A.; Gabbay, D.; Hunter, A.; Kramer, J.; and Nuseibeh, B. 1994. Inconsistency handling in multi-perspective specifications. *IEEE Transactions on Software Engineering* 20(8):569–578.

García, A., and Simari, G. 2004. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming* 4(1):95–138.

Prakken, H., and Vreeswijk, G. 2002. Logics for defeasible argumentation. In Gabbay, D. M., and Guenthner, F., eds., *Handbook of Philosophical Logic*, volume 4. Kluwer, 2nd edition. 219–318.