From: AAAI-00 Proceedings. Copyright © 2000, AAAI (www.aaai.org). All rights reserved.

Towards a Logic-based Theory of Argumentation

Philippe Besnard and **Anthony Hunter**
IRIT-CNRS, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse Cedex, France
Department of Computer Science, University College London, Gower Street, London WC1E 6BT, U.K.

Abstract

There are a number of frameworks for modelling argumentation in logic. They incorporate formal representation of individual arguments and techniques for comparing conflicting arguments. In these frameworks, if there are a number of arguments for and against a particular conclusion, an aggregation function determines whether the conclusion is taken to hold. We propose a generalization of these frameworks. In particular, this new framework makes it possible to define aggregation functions that are sensitive to the number of arguments for or against(in most other frameworks, aggregation functions just consider the existence of arguments for and against). In this paper, we explore this framework (based on classical logic) in which an argument is a pair where the first item in the pair is a minimal consistent set of formulae that proves the second item (which is a formula).

Introduction

Modelling argumentation has been a subject of research as long as the study of logic. They are closely intertwined topics, and modelling argumentation in logic is a natural, and important, research goal. A useful introduction to argumentation is in (Tou58), and a comprehensive recent review of modelling argumentation in logics is in (PV00).

Since paraconsistent logics have been suggested for applications including reasoning with specifications (HN98) and reasoning with news reports in structured text (Hun00), a logic-based theory of argumentation such as proposed below may be applicable in various roles.

Whilst most proposals have been made for modelling argumentation in logic, all are limited in the way that they combine arguments for and against a particular conclusion following. None are sensitive to the number of arguments for and against, apart from some proposals for counting the number for and the number against, and if there are more arguments for, then the conclusions follows, otherwise it is defeated.

Most proposals for modelling argumentation in logic are based on some form of binary argumentation (only the existence of arguments for and against is considered). A simple form of argumentation is that a conclusion follows if and if only if there is an argument for the conclusion, and no argument against the conclusion. So a conclusion follows only if it is not rebutted. A development of this idea is to only consider arguments that have not been undercut, and to check this by recursion for subarguments. An argument is undercut if and only if one of the assumptions for the argument is rebutted. Each undercut to a subargument is itself an argument and so may be undercut, and so by recursion each undercutter needs to be considered.

Copyright © 2000, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

In this paper, we propose a new framework for argumentation with non-binary aggregation functions. For this framework, we have the following requirements:

- to derive arguments from a set of formulae that is potentially inconsistent;
- to compare arguments for and against a particular consequent;
- to identify undercuts for each argument, and by recursion, to identify undercuts for all subarguments for an argument;
- to evaluate each argument in terms of all its undercuts, and by recursion, all undercuts to its subarguments, and the value assigned to an argument decreases with increasing number of undercuts, and increases with increasing number of undercuts to each of the undercuts of the subarguments;
- to accumulate arguments for a consequent so that each extra argument contributes less to the accumulated value;

In our framework, that is based on classical logic, an argument is a pair where the first item in the pair is a minimal consistent set of formulae that proves the second item (which is a formula). Non-binary aggregation functions can be defined that are sensitive to the number of arguments for and against a conclusion. This paper defines the framework and explores its properties.

Preliminaries

We assume familiarity with classical logic.

We consider a propositional language. We use $\alpha, \beta, \gamma, \ldots$ to denote formulae and $\Delta, \Phi, \Psi, \ldots$ to denote sets of formulae. Deduction in classical propositional logic is denoted by the symbol $\vdash$ and deductive closure by Th so that $Th(\Phi) = \{\alpha \mid \Phi \vdash \alpha\}$.

For the following definitions, we first assume a database Δ (a finite set of formulae) and use this Δ throughout.

We further 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. This really is not a demanding constraint: In particular, the constraint is satisfied whenever we impose an arbitrary total ordering over Δ. Importantly, the order has no meaning and is not meant to represent any respective importance of formulae in Δ. It is only a convenient way to indicate the order in which we assume the formulae in any subset of Δ are conjoined to make a formula logically equivalent to that subset.

Arguments

Here we adopt a very common intuitive notion of an argument and consider some of the ramifications of the definition. Essentially, an argument is a set of relevant formulae that can be used to classically prove some point, together with that point (we represent a point by a formula).

Definition 1 *An* **argument** *is a pair* $\langle \Phi, \alpha \rangle$ *such that*

1. $\Phi \not\vdash \bot$.
2. $\Phi \vdash \alpha$.
3. Φ *is a minimal subset of* Δ *satisfying 2.*

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 (we also say that* Φ *is a support for* α*).*

Example 1 *Consider* $\Delta = \{\alpha, \alpha \to \beta, \gamma \to \neg\beta, \gamma, \delta, \delta \to \beta, \neg\alpha, \neg\gamma\}$*. Some arguments are:*

$$\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$$
$$\langle \{\gamma \to \neg\beta, \gamma\}, \neg\beta \rangle$$
$$\langle \{\delta, \delta \to \beta\}, \beta \rangle$$
$$\langle \{\neg\alpha\}, \neg\alpha \rangle$$
$$\langle \{\neg\gamma\}, \neg\gamma \rangle$$
$$\langle \{\alpha \to \beta\}, \neg\alpha \vee \beta \rangle$$
$$\langle \{\neg\gamma\}, \delta \to \neg\gamma \rangle$$

Arguments are not independent. In a sense, some encompass others (possibly up to some form of equivalence). To clarify this requires a couple of definitions.

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 $\langle \{\alpha\}, \alpha \vee \beta \rangle$ *is more conservative than* $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$*. Here, the latter argument can be obtained from the former (using* $\alpha \to \beta$ *as an extra hypothesis) but the reader is warned that such need not be the case in general as we now discuss.*

Example 2 suggests that an argument $\langle \Psi, \beta \rangle$ can be obtained from a more conservative argument $\langle \Phi, \alpha \rangle$ by using $\Psi \setminus \Phi$ together with α in order to deduce β (in symbols, $\{\alpha\} \cup \Psi \setminus \Phi \vdash \beta$ or equivalently, $\Psi \setminus \Phi \vdash \alpha \to \beta$). As just mentioned, this does not hold in full generality. A counterexample consists of $\langle \{\alpha \wedge \gamma\}, \alpha \rangle$ and $\langle \{\alpha \wedge \gamma, \neg\alpha \vee \beta \vee \neg\gamma\}, \beta \rangle$. However, a weaker property holds:

Theorem 1 *If* $\langle \Phi, \alpha \rangle$ *is more conservative than* $\langle \Psi, \beta \rangle$ *then* $\Psi \setminus \Phi \vdash \phi \to (\alpha \to \beta)$ *for some formula* ϕ *such that* $\Phi \vdash \phi$ *and* $\phi \not\vdash \alpha$ *unless* α *is a tautology.*

The interesting case, as in Example 2, is when ϕ can be a tautology.

Theorem 2 *Being more conservative defines a pre-ordering over arguments. Minimal arguments always exist, unless all formulas in* Δ *are inconsistent. Maximal arguments always exist: They are* $\langle \emptyset, \top \rangle$ *where* $\top$ *is any tautology.*

A useful notion is then that of a normal form (a function such that any formula is mapped to a logically equivalent formula and, if understood in a strict sense as here, such that any two logically equivalent formulas are mapped to the same formula).

Theorem 3 *Given a normal form, being more conservative defines an ordering provided that only arguments which have a consequent in normal form are considered. The ordered set of all such arguments is an upper semilattice (when restricted to the language of* Δ*). The greatest argument always exists, it is* $\langle \emptyset, \top \rangle$*.*

Example 3 *The g.l.b. of* $\langle \{\alpha \wedge \beta\}, \alpha \rangle$ *and* $\langle \{\alpha \wedge \neg\beta\}, \alpha \rangle$ *does not exist. If* $\Delta = \{\alpha \wedge \beta, \alpha \wedge \neg\beta\}$*, then there is no least argument. Taking now* $\Delta = \{\alpha, \beta, \alpha \leftrightarrow \beta\}$*, there is no least argument either (although* Δ *is consistent). Even though* $\Delta = \{\alpha, \beta \wedge \neg\beta\}$ *is inconsistent, the least argument exists:* $\langle \{\alpha\}, \alpha' \rangle$ *(where* α' *stands for the normal form of* α*). As the last illustration,* $\Delta = \{\alpha \vee \beta, \beta\}$ *admits the least argument* $\langle \{\beta\}, \beta' \rangle$ *(where* β' *stands for the normal form of* β*).*

No normal form is assumed in the rest of the paper (i.e., we only consider the case where being more conservative is a pre-ordering). In any case, $\langle \emptyset, \top \rangle$ is more conservative than any other argument.

Irrespective of whether we consider an ordering, being more conservative induces, as any pre-ordering does, an equivalence relation (linking any two arguments that are more conservative than each other). Now, another basis for identifying two arguments with each other comes to mind: Pairwise logical equivalence of the components of both arguments. Hence the next definition.

Definition 3 *Two arguments* $\langle \Phi, \alpha \rangle$ *and* $\langle \Psi, \beta \rangle$ *are* **equivalent** *iff* Φ *is logically equivalent to* Ψ *and* α *is logically equivalent to* β.

Theorem 4 *Two arguments are equivalent whenever each is more conservative than the other. In partial converse, if two arguments are equivalent then either each is more conservative than the other or neither is.*

So, there exist equivalent arguments $\langle \Phi, \alpha \rangle$ and $\langle \Psi, \beta \rangle$ that fail to be more conservative than each other (as in Example 4). However, if $\langle \Phi, \alpha \rangle$ is strictly more conservative than $\langle \Psi, \beta \rangle$ (meaning that $\langle \Phi, \alpha \rangle$ is more conservative than $\langle \Psi, \beta \rangle$ but $\langle \Psi, \beta \rangle$ is not more conservative than $\langle \Phi, \alpha \rangle$) then $\langle \Phi, \alpha \rangle$ and $\langle \Psi, \beta \rangle$ are not equivalent.

Example 4 *Let $\Phi = \{\alpha, \beta\}$ and $\Psi = \{\alpha \vee \beta, \alpha \leftrightarrow \beta\}$. The arguments $\langle \Phi, \alpha \wedge \beta \rangle$ and $\langle \Psi, \alpha \wedge \beta \rangle$ are equivalent even though none is more conservative than the other. This means that there exist two distinct subsets of Δ (namely, Φ and Ψ) supporting $\alpha \wedge \beta$.*

Whilst equivalent arguments make the same point (that is, the same inference), we do want to distinguish equivalent arguments from each other. What we do not want is to distinguish between arguments that are more conservative than each other.

Undercuts

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

Definition 4 *An* **undercut** *for an argument $\langle \Phi, \alpha \rangle$ is an argument $\langle \Psi, \neg(\phi_1 \wedge \ldots \wedge \phi_n) \rangle$ where $\{\phi_1, \ldots, \phi_n\} \subseteq \Phi$ and $\Phi \cup \Psi \subseteq \Delta$ by definition of an argument.*

Example 5 *Let $\Delta = \{\alpha, \alpha \to \beta, \gamma, \gamma \to \neg\alpha\}$. Then, $\langle \{\gamma, \gamma \to \neg\alpha\}, \neg(\alpha \wedge (\alpha \to \beta)) \rangle$ is an undercut for $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$. A less conservative undercut for $\langle \{\alpha, \alpha \to \beta\}, \beta \rangle$ is $\langle \{\gamma, \gamma \to \neg\alpha\}, \neg\alpha \rangle$.*

Theorem 5 *Δ is inconsistent if there exists an argument that has at least one undercut. The converse is true when no formula in Δ is inconsistent.*

As arguments can be ordered from more conservative to less conservative, there is a clear and unambiguous notion of maximally conservative undercuts for a given argument (the ones which are representative of all undercuts for that argument):

Definition 5 *$\langle \Phi, \alpha \rangle$ is a* **maximally conservative undercut** *of $\langle \Psi, \beta \rangle$ iff $\langle \Phi, \alpha \rangle$ is an undercut of $\langle \Psi, \beta \rangle$ such that no undercuts of $\langle \Psi, \beta \rangle$ are strictly more conservative than $\langle \Phi, \alpha \rangle$ (i.e., for all undercuts $\langle \Phi', \alpha' \rangle$ of $\langle \Psi, \beta \rangle$, if $\Phi' \subseteq \Phi$ and $\alpha \vdash \alpha'$ then $\Phi \subseteq \Phi'$ and $\alpha' \vdash \alpha$).*

Notice that the consequent of a maximally conservative undercut for an argument is exactly the negation of the full support of the argument:

Theorem 6 *If $\langle \Psi, \neg(\alpha_1 \wedge \ldots \wedge \alpha_n) \rangle$ is a maximally conservative undercut to an argument $\langle \Phi, \beta \rangle$, then $\Phi = \{\alpha_1, \ldots, \alpha_n\}$.*

Note that if $\langle \Psi, \neg(\alpha_1 \wedge \ldots \wedge \alpha_n) \rangle$ is a maximally conservative undercut for an argument $\langle \Phi, \beta \rangle$, then so are $\langle \Psi, \neg(\alpha_2 \wedge \ldots \wedge \alpha_n \wedge \alpha_1) \rangle$ and $\langle \Psi, \neg(\alpha_3 \wedge \ldots \wedge \alpha_n \wedge \alpha_1 \wedge \alpha_2) \rangle$ and so on. However, they are all identical (in the sense that each is more conservative than the others). We can ignore the unnecessary variants by just considering the canonical undercuts defined as follows.

Definition 6 *An argument $\langle \Psi, \neg(\alpha_1 \wedge \ldots \wedge \alpha_n) \rangle$ is a canonical undercut for $\langle \Phi, \beta \rangle$ iff it is a maximally conservative undercut for $\langle \Phi, \beta \rangle$ and $\langle \alpha_1, \ldots, \alpha_n \rangle$ is the canonical enumeration of Φ.*

Theorem 7 *Any two different canonical undercuts for the same argument have the same consequent, but distinct supports.*

Theorem 8 *Given two different canonical undercuts for the same argument, none is more conservative than the other.*

Example 6 *If $\Delta = \{\alpha, \beta, \neg\alpha, \neg\beta\}$, both the following*

$$\langle \{\neg\alpha\}, \neg(\alpha \wedge \beta) \rangle$$
$$\langle \{\neg\beta\}, \neg(\alpha \wedge \beta) \rangle$$

are canonical undercuts for $\langle \{\alpha, \beta\}, \alpha \leftrightarrow \beta \rangle$, but neither is more conservative than the other.

We adopt a lighter notation, writing $\langle \Psi, \diamond \rangle$ for a canonical undercut of $\langle \Phi, \beta \rangle$. Clearly, $\diamond$ is $\neg(\alpha_1 \wedge \ldots \wedge \alpha_n)$ where $\langle \alpha_1, \ldots, \alpha_n \rangle$ is the canonical enumeration of Φ.

Argument trees

An argument tree describes the various ways an argument can be challenged, as well as how the counter-arguments to the initial argument can themselves be challenged, and so on recursively.

Definition 7 *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. *The children nodes of a node N consist of all canonical undercuts for N that obey 2.*

We first give an illustration of an argument tree in Example 7 and then we motivate the conditions of Definition 7 as follows: Condition 2 is meant to avoid the situation illustrated by Example 8; and Condition 3 is meant to avoid the situation illustrated by Example 9.

Example 7 *For $\Delta = \{\alpha, \alpha \to \beta, \gamma, \gamma \to \neg\alpha, \neg\gamma \vee \neg\alpha\}$, we have the following argument tree where $\diamond$ stands for $\neg(\alpha \wedge (\alpha \to \beta))$.*

$$\begin{array}{ccccc} & & \langle \{\alpha, \alpha \to \beta\}, \beta \rangle & & \\ & \nearrow & & \nwarrow & \\ \langle \{\gamma, \gamma \to \neg\alpha\}, \diamond \rangle & & & & \langle \{\gamma, \neg\gamma \vee \neg\alpha\}, \diamond \rangle \end{array}$$

Note the two undercuts are equivalent. They do count as two arguments because they are based on two different items of the database (even though these items turn out to be logically equivalent).

Example 8 *Let $\Delta = \{\alpha, \alpha \to \beta, \gamma \to \neg\alpha, \gamma\}$.*

$$\begin{array}{c} \langle \{\alpha, \alpha \to \beta\}, \beta \rangle \\ \uparrow \\ \langle \{\gamma, \gamma \to \neg\alpha\}, \diamond \rangle \\ \uparrow \\ \langle \{\alpha, \gamma \to \neg\alpha\}, \diamond \rangle \end{array}$$

This is not an argument tree because the undercut to the undercut is actually making exactly the same point (that α and γ are incompatible in the context of Δ) as the undercut itself does, just by using modus tollens instead of modus ponens.

Example 9 *Given $\Delta = \{\alpha, \beta, \alpha \rightarrow \gamma, \beta \rightarrow \delta, \neg\alpha \vee \neg\beta\}$, consider the following tree.*

$$\begin{array}{ccc} & \langle\{\alpha, \beta, \alpha \rightarrow \gamma, \beta \rightarrow \delta\}, \gamma \wedge \delta\rangle & \\ \nearrow & & \nwarrow \\ \langle\{\alpha, \neg\alpha \vee \neg\beta\}, \neg\beta\rangle & & \langle\{\beta, \neg\alpha \vee \neg\beta\}, \neg\alpha\rangle \end{array}$$

This is not an argument tree because the two children nodes are not maximally conservative undercuts. The first undercut is essentially the same argument as the second undercut in a rearranged form (relying on α and β being incompatible, assume one and then conclude that the other doesn't hold). If we replace these by the maximally conservative undercut $\langle\{\neg\alpha \vee \neg\beta\}, \diamond\rangle$, we obtain an argument tree.

The following result is important in practice — particularly in light of other results we present in the next subsection.

Theorem 9 *Argument trees are finite.*

Theorem 10 *Let T be an argument tree. If Δ is consistent, then T has exactly one node. The converse is untrue.*

The form of an argument tree is not arbitrary. It summarizes all lines of discussion about the argument in the root node. Each node except the root node is the starting point of an implicit series of related arguments as shown by Theorem 11. We call these related arguments duplicates. We define this notion formally in the next subsection.

Duplicates

Equivalent arguments are arguments that express the same reason for the same point. For undercuts, a more refined notion than equivalent arguments is useful:

Definition 8 *Two undercuts $\langle\Gamma \cup \Phi, \neg\psi\rangle$ and $\langle\Gamma \cup \Psi, \neg\phi\rangle$ are* **duplicates** *of each other iff ϕ is $\phi_1 \wedge \ldots \wedge \phi_n$ such that $\Phi = \{\phi_1, \ldots, \phi_n\}$ and ψ is $\psi_1 \wedge \ldots \wedge \psi_m$ such that $\Psi = \{\psi_1, \ldots, \psi_m\}$.*

Duplicates introduce a symmetric relation which fails to be transitive (and reflexive). Arguments which are duplicates of each other are essentially the same argument in a rearranged form.

Example 10 *The two arguments below are duplicates of each other.*

$$\langle\{\alpha, \neg\alpha \vee \neg\beta\}, \neg\beta\rangle$$
$$\langle\{\beta, \neg\alpha \vee \neg\beta\}, \neg\alpha\rangle$$

Example 11 *To illustrate the failure of transitivity in the duplicate relationship, the following two arguments are duplicates,*

$$\langle\{\gamma, \alpha, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg\beta\rangle$$
$$\langle\{\gamma, \beta, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg\alpha\rangle$$

and the following two arguments are also duplicates,

$$\langle\{\gamma, \beta, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg\alpha\rangle$$
$$\langle\{\alpha, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg(\beta \wedge \gamma)\rangle$$

but the following two are not duplicates.

$$\langle\{\alpha, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg(\beta \wedge \gamma)\rangle$$
$$\langle\{\gamma, \alpha, \alpha \wedge \gamma \rightarrow \neg\beta\}, \neg\beta\rangle$$

The following result shows how we can systematically obtain duplicates.

Theorem 11 *For every maximally conservative undercut $\langle\Psi, \neg\alpha\rangle$ to an argument $\langle\Phi, \beta\rangle$, there exist at least m arguments each of which undercuts the undercut (m is the size of Ψ). Each of these m arguments is a duplicate to the undercut.*

Theorem 12 *No two maximally conservative undercuts of the same argument are duplicates.*

Theorem 13 *No branch in an argument tree may contain duplicates, except possibly one duplicate to the root.*

These two results are important. They show that argument trees are an efficient and lucid way of representing the pertinent counter-arguments to each argument: Theorem 12 shows it regarding breadth and Theorem 13 shows it regarding depth. Moreover, they show that the intuitive need to eliminate duplicates from argument trees is taken care of through an efficient syntactical criterion (condition 2 of Definition 7).

Argument structures

We now consider how we can gather argument trees for and against a point. In order to do this, we define argument structures.

Definition 9 *An* **argument structure** *for a formula α is a pair of sets $\langle\mathcal{P}, \mathcal{C}\rangle$ where $\mathcal{P}$ is the set of argument trees for α and $\mathcal{C}$ is the set of argument trees for $\neg\alpha$.*

Example 12 *Let $\Delta = \{\alpha \vee \beta, \alpha \rightarrow \gamma, \neg\gamma, \neg\beta, \delta \leftrightarrow \beta\}$. We obtain three argument trees for the argument structure for $\alpha \vee \neg\delta$.*

$$\begin{array}{c} \langle\{\alpha \vee \beta, \alpha \rightarrow \gamma, \neg\gamma, \delta \leftrightarrow \beta\}, \neg\alpha \wedge \delta\rangle \\ \uparrow \\ \langle\{\neg\beta\}, \diamond\rangle \end{array}$$

$$\begin{array}{c} \langle\{\alpha \vee \beta, \neg\beta\}, \alpha \vee \neg\delta\rangle \\ \uparrow \\ \langle\{\alpha \rightarrow \gamma, \neg\gamma\}, \diamond\rangle \end{array}$$

$$\begin{array}{c} \langle\{\delta \leftrightarrow \beta, \neg\beta\}, \alpha \vee \neg\delta\rangle \\ \uparrow \\ \langle\{\alpha \vee \beta, \alpha \rightarrow \gamma, \neg\gamma\}, \diamond\rangle \end{array}$$

Theorem 14 *Let $\langle\mathcal{P}, \mathcal{C}\rangle$ be an argument structure such that $\mathcal{P} \neq \emptyset$. If Δ is consistent, then each argument tree in $\mathcal{P}$ has exactly one node, and $\mathcal{C}$ is the empty set. The converse is untrue.*

Theorem 15 *Let $\langle[X_1, \ldots, X_n], [Y_1, \ldots, Y_m]\rangle$ be an argument structure. For any i and any j, the support of the root node of Y_j (resp. X_i) is a superset of the support of a canonical undercut for the root node of X_i (resp. Y_j).*

Aggregation

We now propose an approach to evaluate argument structures. This approach is actually an illustration of the possibilities arising from our framework: We will see that it needs be modified and generalized to cope with some particular requirements.

Definition 10 *A* **categorizer** *is a mapping from argument trees to numbers. A categorization is then a pair of multisets obtained by applying the same categorizer to each argument tree in an argument structure.*

The number assigned by a categorizer is intended to capture the relative strength of an argument taking into account the undercuts, undercuts to undercuts, and so on. I.e., it is an attempt to provide an abstraction of an argument tree in the form of a single number.

The **h-categorizer**, *denoted h, is an example of a categorizer. An argument tree of root R is assigned a number $h(R)$ defined recursively by*

$$h(N) = \frac{1}{1 + h(N_1) + \cdots + h(N_l)}$$

where $N_1, \ldots, N_l$ are the children nodes for N (if $l = 0$, $h(N_1) + \cdots + h(N_l) = 0$).

Definition 11 *An* **accumulator** *is a function that takes a categorization for a formula α and returns a pair of numbers (α^+, α^-) st α^+ is the accumulated value for α and α^- is the accumulated value against α. The balance of accumulated values is calculated as $\alpha^+ - \alpha^-$.*

If the balance of accumulated values is 0, then the arguments for the formula "equal" the arguments against the formula. If the balance of accumulated values is positive (negative), then the arguments for the formula are stronger (weaker) —when aggregated— than the arguments against the formula.

The **l-accumulator** *is an example of an accumulator. For any categorization $\langle [X_1, \ldots, X_n], [Y_1, \ldots, Y_m] \rangle$, let*

$$l(\langle [X_1, \ldots, X_n], [Y_1, \ldots, Y_m] \rangle) = (\log(1 + X_1 + \cdots + X_n), \log(1 + Y_1 + \cdots + Y_m)).$$

Example 13 *Consider the categorization $\langle [1], [1/2] \rangle$. Using the l-accumulator, we obtain 0.47 as the balance of the accumulated values. For the categorization $\langle [1, 1/2], [1/2, 1/2] \rangle$, the l-accumulator gives 0.41 as the balance of the accumulated values. So we can see that adding an argument tree of value $1/2$ to both the pro and con sides benefits the con side since initially the con side is a much weaker argument than the pro side.*

Example 14 *For $\langle [1/2, 1/2], [1] \rangle$, the l-accumulator gives -0.25 as the balance of the accumulated values. For $\langle [1/2, 1/2, 1/2], [1, 1/2] \rangle$, the l-accumulator gives -0.29 as the balance of the accumulated values. Here, we see that adding an argument tree of value $1/2$ to both the pro and con sides benefits the con side since initially the pro side has two arguments of value $1/2$ but the con side has a single argument of value 1 (in particular, we want an argument to have a more profound effect when confirming a single argument than when joining a hundred similar arguments already agreeing).*

Definition 10 and Definition 11 make it possible to have categorizer and accumulator functions conforming to a probabilistic approach and the same holds for a diverse range of other approaches to argumentation. An example is binary argumentation (RM70; HTT90; BDP93; Nut94; GLV98), and counting arguments for and against (SL92; PL92) is another simple case. A general approach that can be incorporated in our framework is argumentative logics (see e.g., (EGH95)). Categorizers for doing so include constant functions ($\forall T, cat(T) = 1$), and other very simple functions ($cat(T) = 1$ iff T has exactly one node). No complex accumulator is then needed either.

As an illustration, a few approaches to argumentation based on reasoning with maximal consistent subsets are:

- *α is an* **existential inference** *from Δ iff $\Theta \vdash \alpha$ where Θ is some maximal consistent subset of Δ.*
- *α is an* **unrebutted inference** *from Δ iff $\Theta \vdash \alpha$ for some maximal consistent subset Θ of Δ and $\Omega \not\vdash \neg\alpha$ for each maximal consistent subset Ω of Δ.*
- *α is a* **universal inference** *from Δ iff $\Omega \vdash \alpha$ for each maximal consistent subset Ω of Δ.*
- *α is a* **free inference** *from Δ iff $\Lambda \vdash \alpha$ st Λ is the intersection of all maximal consistent subsets of Δ.*

Definition 12 *The* **unit categorizer** *is a function, denoted c, from the set of argument trees to $\{1\}$ such that $c(T) = 1$ in all cases.*

Definition 13 *The* **unit accumulator** *is a function, denoted u_a, from the set of categorizations to the set $\{(1,1), (1,0), (0,1), (0,0)\}$ st for a categorization $\langle X, Y \rangle$*

$$u_a(\langle X, Y \rangle) = (p(X), p(Y))$$

where $p(Z) = 1$ iff $Z \neq \emptyset$.

Theorem 16 *If applying the unit categorizer to the argument structure for α yields $\langle X, Y \rangle$, then α is an existential inference from Δ iff $u_a(\langle X, Y \rangle) = (1, 0)$ or $(1, 1)$.*

Theorem 17 *α is an unrebutted inference from Δ iff $u_a(\langle X, Y \rangle) = (1, 0)$ where $\langle X, Y \rangle$ results from applying the unit categorizer to the argument structure for α.*

Similarly, free inferencing is captured in our framework without recourse to maximal consistent subsets.

Definition 14 *The* **free categorizer** *is a function, denoted s, from the set of argument trees to $\{0, 1\}$ such that $s(T) = 1$ iff T is just a root node.*

Definition 15 *The* **free accumulator** *is a function, denoted f_a, from the set of categorizations to the set $\{(1,1), (1,0), (0,1), (0,0)\}$ st for a categorization $\langle X, Y \rangle$*

$$f_a(\langle X, Y \rangle) = (w(X), w(Y))$$

where $w(Z) = 1$ iff $1 \in Z$.

Theorem 18 *Let $\langle X, Y \rangle$ result from applying the free categorizer to the argument structure for α. Then, α is a free inference from Δ iff $f_a(\langle X, Y \rangle) = (1, 0)$.*

Universal inferencing cannot be captured using the above definitions for aggregation unless they are generalized (keep in mind that all results, apart from theorems 16 to 18, would still hold) and a similar observation applies for Dung's well-known approach (Dun95).

Yet, an argument structure can be viewed as an argumentation framework in Dung's sense. Also, an argument in our sense amounts to an argument in a Dung argumentation framework, and an arc in an argument tree amounts to an attack in Dung's sense. However, the way sets of arguments are compared is different.

Definition 16 *A* **Dung argumentation framework** *is a pair $\Delta = \langle \Gamma, \mathcal{A} \rangle$ where Γ is a set and $\mathcal{A} \subseteq \Gamma \times \Gamma$ (intuitively, $\mathcal{A}$ is an "attack" relation between arguments).*

Definition 17 *A subset S of Γ is* **conflict-free** *if there are no two X, Y in S such that $X\mathcal{A}Y$ or $Y\mathcal{A}X$.*

Definition 18 *$X \in \Gamma$ is* **acceptable** *wrt $S \subseteq \Gamma$ iff for each $Y \in \Gamma$, if $Y\mathcal{A}X$ then $Z\mathcal{A}Y$ for some $Z \in S$.*

Definition 19 *A conflict-free subset S of Γ is* **admissible** *iff each element of S is acceptable wrt S.*

Example 15 *Let $\Delta = \{\beta, \beta \to \alpha, \gamma, \gamma \to \neg\beta, \delta \wedge \neg\gamma, \neg\delta \wedge \neg\gamma, \neg\gamma \to \neg\alpha\}$. Consider the argument trees:*

$$\begin{array}{c} \langle \{\neg\gamma, \neg\gamma \to \neg\alpha\}, \neg\alpha \rangle \\ \uparrow \\ \langle \{\gamma\}, \diamond \rangle \end{array}$$

$$\begin{array}{ccccc} & & \langle \{\beta, \beta \to \alpha\}, \alpha \rangle & & \\ & & \uparrow & & \\ & & \langle \{\gamma, \gamma \to \neg\beta\}, \diamond \rangle & & \\ & \nearrow & & \nwarrow & \\ \langle \{\delta \wedge \neg\gamma\}, \diamond \rangle & & & & \langle \{\neg\delta \wedge \neg\gamma\}, \diamond \rangle \\ \uparrow & & & & \uparrow \\ \langle \{\neg\delta \wedge \neg\gamma\}, \diamond \rangle & & & & \langle \{\delta \wedge \neg\gamma\}, \diamond \rangle \end{array}$$

There is no admissible set of arguments that contains $\langle \{\beta, \beta \to \alpha\}, \alpha \rangle$. Thus, we see that our approach cannot capture Dung's unless categorization is redefined so that instead of categorizing each tree individually, it considers each tree in the context of the other trees.

Concluding remarks

Here, we have proposed a new framework for modelling argumentation. The key feature is that we incorporate non-binary aggregation functions. This framework can be viewed as a generalization of a wide range of existing approaches to argument aggregation. Moreover, non-binary argument aggregation offers a more realistic approach to weighing up the relative merits of arguments for and against a possible conclusion.

We believe that there are a range of possible applications of our framework in reasoning with potentially inconsistent information. Such applications include reasoning with inconsistent specifications (HN98) and inconsistent structured text (Hun00).

In order to use the framework more generally, we may wish to differentiate the information in Δ from some background knowledge Σ where we assume that Σ is uncontroversial knowledge that can be taken for granted and Δ is questionable information. We can then generalize our definition of the consequence relation $\vdash$ to that of a consequence relation $\vdash_\Sigma$ where inferences can be derived with the benefit of the formulae in Σ.

References

S. Benferhat, D. Dubois, and H. Prade. Argumentative inference in uncertain and inconsistent knowledgebases. In *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence (UAI'93)*, pages 485–491. Morgan Kaufmann, 1993.

P. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

M. Elvang-Goransson and A. Hunter. Argumentative logics: Reasoning from classically inconsistent information. *Data and Knowledge Engineering*, 16:125–145, 1995.

P. Geerts, E. Laenens, and D. Vermeir. Defeasible logics. In *Handbook of Defeasible Reasoning and Uncertainty Management Systems. Volume 2: Reasoning with Actual and Potential Contradictions*, pages 175–210. Kluwer, 1998.

A. Hunter and B. Nuseibeh. Managing inconsistent specifications: Reasoning, analysis and action. *ACM Transactions on Software Engineering and Methodology*, 7:335–367, 1998.

A. Hunter. Reasoning with inconsistency in structured text. *Knowledge Engineering Review*, to appear.

J. Horty, R. Thomason, and D. Touretzky. A skeptical theory of inheritance in nonmonotonic semantic networks. *Artificial Intelligence*, 42:311–348, 1990.

D. Nute. Defeasible logics. In *Handbook of Logic in Artificial Intelligence and Logic Programming. Volume 3: Nonmonotonic Reasoning and Uncertainty Reasoning*, pages 355–395. Oxford University Press, 1994.

G. Pinkas and R. Loui. Reasoning from inconsistency: A taxomony of principles for resolving conflict. In *Proceedings of the Third International Conference on Knowledge Representation and Reasoning (KR'92)*, pages 709–719. Morgan Kaufmann, 1992.

H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*, 2nd ed., to appear.

N. Rescher and R. Manor. On inference from inconsistent premisses. *Theory and Decision*, 1:179–217, 1970.

G. Simari and R. Loui. A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence*, 53:125–157, 1992.

S. Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.