Towards Computational Models of Natural Argument using Labelled Deductive Systems

Carlos Iván Chesñevar
Departament of Computer Science
Universitat de Lleida
C/Jaume II, 69 – 25001 Lleida, SPAIN
Email: cic@eps.udl.es

Guillermo Ricardo Simari
Dept. of Computer Science and Engineering
Universidad Nacional del Sur
Alem 1253 – 8000 Bahía Blanca, ARGENTINA
Email: grs@cs.uns.edu.ar

Abstract

During the last decade computational models of argument have emerged as a successful approach to the formalization of commonsense reasoning, encompassing many other alternative formalisms. Common elements can be identified in such frameworks along with a number of particular features which make it difficult to compare them with each other from a logical viewpoint. This paper presents a unifying approach to computational models of argument using Labelled Deductive Systems (LDS), a rigorous but flexible methodology which has been developed to formalize complex systems as logical frameworks with labelled deduction capabilities. In the context of defeasible argumentation, we show how labels can be used to represent arguments as well as argument trees. In particular, we will describe the wide range of possibilities allowed by an LDS-based formalization, such as capturing fallacies, formalizing arguments in social contexts, weighing arguments, and formalizing argument-based consequence operators.

Keywords: argumentation theory, labelled deductive systems, knowledge representation

1 Introduction and Motivation

During the last decade computational models of argument have emerged as a successful approach to the formalization of commonsense reasoning, encompassing many other alternative formalisms. Many different argument-based frameworks have been developed [Prakken and Vreeswijk, 2002; Chesñevar *et al.*, 2000]. In recent years, such frameworks have had considerable impact on multi-agent systems as a vehicle for facilitating "rational interaction" among agents. As highlighted in [Prakken and Vreeswijk, 2002; Chesñevar *et al.*, 2000] several common elements can be identified in such frameworks: an underlying logical language, the concept of argument, etc. However, these elements appear along with a number of particular features which make it difficult to compare such frameworks with each other from a logical viewpoint.

Labelled Deductive Systems (LDS) [Gabbay, 1996; Gabbay *et al.*, 2004] were developed as a rigorous but flexible methodology to formalize complex systems as logical frameworks with labelled deduction capabilities ($e.g.$, temporal logics, database query languages and defeasible reasoning systems). This paper analyzes the role of LDS as a methodology for building computational models of argument. We show how different features in such models can be expressed within a unified labelling methodology. As a basis for our analysis we will use LDS_{AR}, an LDS-based approach to defeasible argumentation based on logic programming. In the context of LDS_{AR} we show how labels can be used to express common elements in argument-based frameworks ($e.g.$, knowledge, arguments, argument trees). In particular, we will describe the wide range of possibilities allowed by an LDS-based formalization, such as capturing fallacies, formalizing arguments in social contexts, weighing arguments, and formalizing argument-based consequence operators.

The rest of this paper is structured as follows. Section 2 introduces some fundamentals of LDS. Section 3 presents the main elements of the LDS_{AR} framework [Chesñevar and Simari, 2000], along with a worked example. Section 4 describes different aspects of computational models of argument in the context of the proposed LDS-based approach, including fallacy detection, argumentation in social contexts, measuring the impact of arguments in dialectical trees, and formalizing argument-based consequence operators. Section 5 discusses related work, and finally Section 6 summarizes future research and the main conclusions that we have obtained.

2 Labelled Deductive Systems: Fundamentals

Logic has been traditionally perceived as the study of 'consequence relations' between sets of formulæ. The complexity of problems and the associated formalizations in different application areas have emphasized the need for the definition of consequence relations between *structures* of formulas, such as multi-sets, sequences, or trees [Gabbay *et al.*, 2004]. An interesting question is how to formally define the differences existing among logics, which involves characterizing a way of comparing them. According to [Gabbay, 1996], the answers to these considerations are to be found in metalevel considerations, which can be better identified by analyzing those aspects which are *uniform* in most logics ($e.g.$, the structure of inference rules, rules for quantifiers, etc.) Labelled deductive systems aim to characterize a logical system by 'abstracting away' these common aspects. As a first approximation, a labelled deductive system is a 3-uple $(\mathbf{A}, \mathcal{L}, M)$, where $\mathcal{L}$ is a *logical language* (including connectives and wffs), $\mathbf{A}$ is an *algebra* on labels (with given operations), and M is a *discipline* which indicates how to label formulas in the logic, given the algebra $\mathbf{A}$ of labels [Gabbay, 1996]. Such a discipline will be formulated using deduction rules. In order to characterize an LDS, a labelled language must be defined including wffs and labels. In such language labels can be seen as carriers of information which is not present in the wffs themselves.

Why are labels needed? In LDS labels will be used to store information of different sort of the one encoded in the predicate associated with it. There may exist different reasons for doing this: it can be the case that the information on the label is of a different nature or purpose than the one coded in the main predicate, and therefore it is more convenient to keep it as an annotation or label; or it may also be the case that the manipulation of this extra information is too complex, and so we want to keep it apart from the predicate associated with it. Instead of referring to a formula A, the name of *declarative unit* is generically used to refer to labelled formulæ $t : A$. As Gabbay remarks (*op. cit.*), there may be many uses for a label t in such a declarative unit $t : A$. The value t might correspond to a confidence value in fuzzy logic ($e.g.$, t could be a real number between 0 and 1), an indicator of the origin of the wff A ($e.g.$, in a very complex database), or an annotation of the proof of A ($e.g.$, t can include the set of assumptions that lead to believe A).

Finally, whereas in traditional logical systems the consequence is defined using proof rules on formulas, in the LDS methodology the consequence is defined by using rules on *both* formulæ and labels. Thus the traditional notion of consequence between formulæ of the form $A_1, \ldots, A_n \vdash B$ is replaced by the notion of consequence between labelled formulæ $t_1{:}A_1; t_2{:}A_2; \ldots; t_n{:}A_n \mid\!\sim s{:}B$. Accordingly, we will have formal rules for manipulating labels and this will allow for more scope and detailed analysis when decomposing the various features of the consequence relation. The meta features can be reflected in the algebra or logic of the labels, and the object features in the rules of the formulas. An in-depth discussion on LDS is outside the scope of this paper, and for further details the reader is referred to [Gabbay, 1996; Gabbay *et al.*, 2004].

3 Modelling Argumentation with LDS: the *LDS*$_{AR}$ Framework

Argumentation frameworks[1] are characterized by representing certain features of informal argumentation using a formal language, along with an inference mechanism. Although these frameworks differ in their aims and characterization, the notion of *argument* is quite similar, having a strong resemblance to the notion of *proof* in logic. In fact, the difference between arguments and logical proofs is more 'pragmatic' than 'syntactic' [Krause *et al.*, 1995].

Prakken & Vreeswijk [Prakken and Vreeswijk, 2002] have defined a conceptual framework in which most argumentation systems can be characterized. This conceptual framework involves five elements, namely:

a) an underlying logical language $\mathcal{L}$;

b) a concept of *argument*;

c) a concept of *conflict* among arguments;

d) a notion of *defeat* among arguments;

e) a notion of *acceptability* of arguments according to a well-defined criterion.

We contend that the above elements can be embedded as different parts of an LDS-based framework for argumentation called *LDS*$_{AR}$ [Chesñevar and Simari, 2000], whose salient features will be summarized in this section. In our approach the underlying logical language $\mathcal{L}$ will be a labelled language $\mathcal{L}_{\mathsf{Arg}} = (\mathcal{L}_{\mathsf{Labels}}, \mathcal{L}_{\mathsf{KR}})$, where $\mathcal{L}_{\mathsf{Labels}}$ is a labelling language (representing epistemic status of knowledge, as well as arguments and their interrelationships) and $\mathcal{L}_{\mathsf{KR}}$ represents object-level knowledge. Thus, the labelling language $\mathcal{L}_{\mathsf{Labels}}$ will encode different information features which correspond to the elements (a)–(e) in Prakken & Vreeswijk's conceptualization.

Usually $\mathcal{L}_{\mathsf{KR}}$ will be a distinguished subset of FOL ($e.g.$, the language of Horn clauses or the language of extended logic programming). For practical purposes, $\mathcal{L}_{\mathsf{KR}}$ will usually be restricted to *rules* and *facts*, in which the notion of *contradictory information* can be expressed in terms of complementary literals p and $\overline{p}$.[2] We will also assume an underlying inference procedure $\vdash$ associated with $\mathcal{L}_{\mathsf{KR}}$ ($e.g.$, SLD derivation). Given a set $P \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$, and $\phi \in \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$, we will write that $P \vdash \phi$ to denote that ϕ follows from P via $\vdash$. If two complementary literals can be derived from P via $\vdash$ we will just write $P \vdash \bot$. Following [Gabbay, 1996], labelled wffs in $\mathcal{L}_{\mathsf{Arg}}$ will be called *declarative units*, having the form *Label:wff*.

Definition 1 (Labeling language $\mathcal{L}_{\mathsf{Labels}}$) *The labelling language $\mathcal{L}_{\mathsf{Labels}}$ is a set of labels $\{ L_1, L_2, \ldots L_k, \ldots \}$, such that every label $L \in \mathcal{L}_{\mathsf{Labels}}$ is:*

1. *The empty set $\emptyset$, or any $\phi \in \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$. These labels are called* epistemic labels.
2. *A set $\Phi \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$. This is a label called* argument label.
3. *A functor $\mathbf{T}$ is a label called* dialectical label, *defined as follows:*
   - *(a) If Φ is an argument label, then $\mathbf{T}^U(\Phi)$, $\mathbf{T}^D(\Phi)$ and $\mathbf{T}^*(\Phi)$ are* dialectical labels *in $\mathcal{L}_{\mathsf{Labels}}$.*[3]
   - *(b) If $\mathbf{T}_1, \ldots, \mathbf{T}_k$ are dialectical labels, then $\mathbf{T}_n^U(\mathbf{T}_1, \ldots, \mathbf{T}_k)$, $\mathbf{T}_n^*(\mathbf{T}_1, \ldots, \mathbf{T}_k)$ and $\mathbf{T}_m^D(\mathbf{T}_1, \ldots, \mathbf{T}_k)$ will also be dialectical labels in $\mathcal{L}_{\mathsf{Labels}}$.*

[1] See [Prakken and Vreeswijk, 2002; Chesñevar *et al.*, 2000] for a detailed description of relevant logic-based approaches to argumentation.

[2] In this respect we follow an approach similar to the one introduced in [Bondarenko *et al.*, 1997] for abstract argumentation frameworks.

[3] For the sake of simplicity, we will just write $\mathbf{T}_1$, $\mathbf{T}_2$, etc. to denote arbitrary dialectical labels.

- Intro-NR: $\dfrac{}{\emptyset{:}\alpha}$
  for any $\emptyset{:}\alpha$

- Intro-RE: $\dfrac{}{\Phi{:}\alpha}$
  for any $\Phi{:}\alpha$ such that $Strict(\mathbf{\Gamma}) \cup \Phi \not\vdash \bot$

- Intro-$\wedge$: $\dfrac{\Phi_1{:}\alpha_1 \quad \Phi_2{:}\alpha_2 \quad \ldots \Phi_k{:}\alpha_k}{\cup_{i=1\ldots k}\Phi_i{:}\alpha_1, \alpha_2, \ldots, \alpha_k}$

  whenever $Strict(\mathbf{\Gamma}) \cup \bigcup_{i=1\ldots k} \Phi_i \not\vdash \bot$

- Elim-$\leftarrow$: $\dfrac{\Phi_1{:}\beta \leftarrow \alpha_1, \ldots, \alpha_k \quad \Phi_2{:}\alpha_1, \ldots, \alpha_k}{\Phi_1 \cup \Phi_2{:}\beta}$
  whenever $Strict(\mathbf{\Gamma}) \cup \Phi_1 \cup \Phi_2 \not\vdash \bot$

Figure 1: Inference rules for argument construction

4. *Nothing else is a label in* $\mathcal{L}_{\mathsf{Labels}}$.

Next we introduce the notion of *argumentative theory*, which will be a set of 'basic' declarative units (bdu's) in our labelled language $\mathcal{L}_{\mathsf{Arg}}$. Such bdu's will be used to encode defeasible and non-defeasible information.

Definition 2 (Argumentative theory) *A labelled formula* $\phi{:}\alpha \in \mathsf{Wffs}(\mathcal{L})$ *such that* $\alpha \in \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$ *and either (1)* $\phi = \emptyset$ *or (2)* $\phi = \{\alpha\}$ *will be called a* basic declarative unit *(bdu). Cases (1) and (2) correspond to representing non-defeasible and defeasible knowledge, resp. A finite set* $\mathbf{\Gamma} = \{\phi_1{:}\alpha_1, \ldots \phi_k{:}\alpha_k\}$ *where every* $\phi_i{:}\alpha_i$ *is a bdu will be called an* argumentative theory. *We will assume that the set* $Strict(\mathbf{\Gamma}) = \{\emptyset{:}\alpha_i \mid \emptyset{:}\alpha_i \in \mathbf{\Gamma}\}$ *is non-contradictory wrt* $\vdash$.

3.1 Argument Construction

Given an argumentative theory $\mathbf{\Gamma}$ and a wff $\phi \in \mathcal{L}_{\mathsf{KR}}$, we will provide a labelled inference relationship "$\mathrel{|\!\!\sim}_{Arg}$" to characterize the notion of argument. Our labelled inference relationship "$\mathrel{|\!\!\sim}_{Arg}$" will be characterized by a number of suitable deduction rules Intro-NR, Intro-RE, Intro-$\wedge$ and Elim-$\leftarrow$(Figure 1). Rules Intro-NR and Intro-RE allow the introduction of non-defeasible and defeasible information when constructing arguments. Rules Intro-$\wedge$ and Elim-$\leftarrow$ stand for introducing conjunction and applying modus ponens. Note that in the last three rules a 'consistency check' wrt $\vdash$ is performed, in order to ensure that the label associated with the inferred formula does not allow the derivation of complementary literals. Note also that the label $\mathcal{A}$ associated with a formula $\mathcal{A}{:}\alpha$ contains all *defeasible* information needed to conclude α from $\mathbf{\Gamma}$.

Definition 3 (Argument) *Let* $\mathbf{\Gamma}$ *be an argumentative theory, let* α *be a literal in* $\mathcal{L}_{\mathsf{KR}}$ *and let* $\mathcal{A} \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$ *such that* $\mathbf{\Gamma} \mathrel{|\!\!\sim}_{Arg} \mathcal{A}{:}\alpha$. *Then* $\mathcal{A}{:}\alpha$ *will be called an* argument *on the basis of* $\mathbf{\Gamma}$.

- Intro-1D:
  $$\frac{\mathcal{A}{:}\alpha}{\mathbf{T}^*(\mathcal{A}){:}\alpha}$$
  whenever $\mathcal{A}$ is minimal wrt set inclusion

- Intro-ND:
  $$\frac{\mathbf{T}^*(\mathcal{A}){:}\alpha \quad \mathbf{T}^*_1(\mathcal{B}_1, \ldots){:}\beta_1 \quad \mathbf{T}^*_k(\mathcal{B}_k, \ldots){:}\beta_k}{\mathbf{T}^*(\mathcal{A}, \mathbf{T}^*_1, \ldots, \mathbf{T}^*_k){:}\alpha}$$
  whenever $\mathsf{VSTree}(\mathcal{A}, \mathbf{T}^*_i)$ holds, $i = 1 \ldots k$.

- Mark-Atom:
  $$\frac{\mathbf{T}^*(\mathcal{A}){:}\alpha}{\mathbf{T}^U(\mathcal{A}){:}\alpha}$$

- Mark-1D:
  $$\frac{\begin{array}{c}\mathbf{T}^*(\mathcal{A}, \mathbf{T}^*_1, \ldots, \mathbf{T}^*_i, \ldots, \mathbf{T}_k){:}\alpha \\ \mathbf{T}^U_i(\mathcal{B}_i \ldots){:}\beta_i\end{array}}{\mathbf{T}^D(\mathcal{A}, \mathbf{T}^*_1, \ldots, \mathbf{T}^*_{i-1}, \mathbf{T}^U_i, \mathbf{T}^*_{i+1}, \ldots, \mathbf{T}^*_k){:}\alpha}$$
  whenever $\mathsf{VSTree}(\mathcal{A}, \mathbf{T}^U_j)$ holds, for some $i \in \{1, \ldots, k\}$

- Mark-ND:
  $$\frac{\begin{array}{c}\mathbf{T}^*(\mathcal{A}, \mathbf{T}^*_1, \ldots, \mathbf{T}^*_i, \ldots, \mathbf{T}^*_k){:}\alpha \\ \mathbf{T}^D_1(\mathcal{B}_1, \ldots){:}\beta_1 \ldots \mathbf{T}^D_k(\mathcal{B}_k, \ldots){:}\beta_k\end{array}}{\mathbf{T}^U(\mathcal{A}, \mathbf{T}^D_1, \ldots, \mathbf{T}^D_i, \ldots, \mathbf{T}^D_k){:}\alpha}$$
  whenever $\mathsf{VSTree}(\mathcal{A}, \mathbf{T}^D_i)$, $\forall i \in \{1, \ldots, k\}$

Figure 2: Rules for dialectical analysis

3.2 Attack among Arguments. Dialectical Analysis

Clearly, given an argumentative theory $\mathbf{\Gamma}$ there may exist conflicting arguments ($e.g.$, $\mathcal{A}{:}\alpha$ and $\mathcal{B}{:}\overline{\alpha}$) emerging from it. We will assume that *conflict* (also *counterargument* or *attack*) among arguments is captured using the notion of contradiction associated with the $\vdash$ inference relationship used for argument construction. Note that our notion of conflict is intentionally generic, as different, more concrete formalizations are possible ($e.g.$, attack *à la* Simari-Loui, where argument $\mathcal{A}{:}\alpha$ attacks $\mathcal{B}{:}\beta$ iff there exists a subargument $\mathcal{B}'{:}\gamma$, with $\mathcal{B}' \subseteq \mathcal{B}$, such that $Strict(\mathbf{\Gamma}) \cup \{\alpha, \gamma\} \vdash \bot$).

Defeat among arguments involves a *preference criterion* among conflicting arguments. Approaches to characterizing defeat may differ: some argumentation frameworks will only consider attack relationships [Dung, 1995], others will distinguish between rebutting and undercutting attacks [Prakken and Sartor, 1997], etc. We will not delve into such differences here, but will rather focus on capturing the notion of dialectical analysis in terms of natural deduction rules. A usual approach involves computing (explicitly or implicitly) a so-called *dialectical tree*.[4] A dialectical tree is a dialogue tree between two parties, proponent and opponent. Branches of the tree correspond to the exchange of arguments between these two parties. A dialectical tree can be marked as an AND-OR tree according to the following procedure: nodes with no defeaters (leaves) are marked as U-nodes (undefeated nodes).

[4] Also called "argument tree" or "dialogue tree" in the literature.

Inner nodes are marked as D-nodes (defeated nodes) iff they have at least one U-node as a child, and as U-nodes iff they have every child marked as a D-node.

In the context of LDS_{AR}, the construction and marking of dialectical trees is captured in terms of *dialectical labels* (Def. 1). Special marks (*, U, D) are associated with a label $\mathbf{T}(\mathcal{A}, \ldots)$ in order to determine whether $\mathcal{A}$ corresponds to an argument which has been (a) *not analyzed yet* ($*$) in the dialectical context given by the label; or (b) *defeated* (D) (resp. *undefeated* (U)) in such context. In LDS_{AR}, the construction of dialectical trees is formalized in terms of an inference relationship $\mathrel{|\!\!\sim}_{\mathcal{T}}$ given by the natural deduction rules shown in Figure 2. Rule Intro-1D allows to generate a tree with a single argument.[5] Rule Intro-ND allows to expand a given tree $\mathbf{T}^*$ by introducing new subtrees $\mathbf{T}_1^*(\mathcal{B}_1, \ldots)$:$q_1$ $\mathbf{T}_k^*(\mathcal{B}_k, \ldots)$:$q_k$. A special condition VSTree($\mathcal{A}$, $\mathbf{T}_i^*$) , $i = 1 \ldots k$ checks that such subtrees are valid. Such checking involves several considerations, such as determining that the root of every $\mathbf{T}_i^*$ is a defeater for the root of $\mathbf{T}^*$, and no fallacious argumentation is present by appending any $\mathbf{T}_i^*$ as a subtree rooted in $\mathcal{A}$. An in-depth discussion of such fallacies is outside the scope of this paper, and details can be found elsewhere [Chesñevar and Simari, 2000].

Rules Mark-Atom, Mark-1D and Mark-ND allow to 'mark' the nodes (arguments) in a dialectical tree as defeated or undefeated. Note that the rules propagate marking from the bottom of the tree up to the root node, according to the marking criterion discussed before.

Definition 4 (Warrant – Version 1) *Let $C^k_{war}(\mathbf{\Gamma})$ be the set of all formulas that can be obtained from $\mathbf{\Gamma}$ via $\mathrel{|\!\!\sim}_{\mathcal{T}}$ in at most k steps. A literal α is said to be* warranted *iff $\mathbf{T}_i^U(\mathcal{A}, ...)$:$\alpha \in C^k_{war}(\mathbf{\Gamma})$, and there is no $k' > k$, such that $\mathbf{T}_j^D(\mathcal{A}, ...)$:$\alpha \in (C^{k'}_{war}(\mathbf{\Gamma}) \setminus Cn^k_{war}(\mathbf{\Gamma}))$.*

This approach resembles Pollock's original ideas of "ultimately justified belief" [Pollock, 1991]. Note that Def. 4 forces the computation of the deductive closure under "$\mathrel{|\!\!\sim}_{\mathcal{T}}$" in order to determine whether a literal is warranted or not. Fortunately this is not necessarily the case, since warrants can be captured in terms of a *precedence relation* "$\sqsubset$" between dialectical labels. Informally, we will write $\mathbf{T} \sqsubset \mathbf{T}$' whenever $\mathbf{T}$ reflects a state in a dialogue which is *previous* to $\mathbf{T}$' (in other words, $\mathbf{T}$' stands for a dialogue which evolves from $\mathbf{T}$ by incorporating new arguments). A *final label* is a dialectical label that cannot be extended any further.

Definition 5 (Warrant – Version 2) [6] *Let $\mathbf{\Gamma}$ be an argumentative theory, such that $\mathbf{\Gamma} \mathrel{|\!\!\sim}_{\mathcal{T}} \mathbf{T}_i^U(\mathcal{A}, \ldots)$:$\alpha$ and $\mathbf{T}_i^U$ is a final label ($i.e.$, it is not the case that $\mathbf{\Gamma} \mathrel{|\!\!\sim}_{\mathcal{T}} \mathbf{T}_j^D(\mathcal{A}, \ldots)$:$\alpha$ and $\mathbf{T}_i^U \sqsubset \mathbf{T}_j^D$). Then α is a warranted literal wrt $\mathbf{\Gamma}$.*

3.3 A Worked Example

Consider an intelligent agent involved in controlling an engine with three switches $sw1$, $sw2$, and $sw3$. These switches

[5]We require arguments to be minimal wrt set inclusion as it is a common requirement in several argument frameworks, starting with [Simari and Loui, 1992].

[6]It can be proven that Def. 5 and 4 are equivalent [Chesñevar and Simari, 2000].

$$
\begin{array}{rl}
\emptyset: & \overline{f} \leftarrow pc \\
\emptyset: & sw1 \leftarrow \\
\emptyset: & sw2 \leftarrow \\
\emptyset: & sw3 \leftarrow \\
\emptyset: & h \leftarrow \\
\{\, pf \leftarrow sw1 \,\}: & pf \leftarrow sw1 \\
\{\, f \leftarrow pf \,\}: & f \leftarrow pf \\
\{\, po \leftarrow sw2 \,\}: & po \leftarrow sw2 \\
\{\, o \leftarrow po \,\}: & o \leftarrow po \\
\{ e \leftarrow f, o \,\}: & e \leftarrow f, o \\
\{ \overline{e} \leftarrow f, o, h \,\}: & \overline{e} \leftarrow f, o, h \\
\{ \overline{o} \leftarrow h \,\}: & \overline{o} \leftarrow h \\
\{ pc \leftarrow pf, l \,\}: & pc \leftarrow pf, l \\
\{ l \leftarrow sw2 \,\}: & l \leftarrow sw2 \\
\{ \overline{l} \leftarrow sw2, sw3 \,\}: & \overline{l} \leftarrow sw2, sw3 \\
\{ f \leftarrow sw3 \,\}: & f \leftarrow sw3
\end{array}
$$

Figure 3: Argumentative theory Γ_{engine}

regulate different features of the engine, such as the pumping system, speed, etc. Suppose we have defeasible information about how this engine works.

- If the pump is clogged (pc), then the engine gets no fuel ($\overline{f}$).
- When $sw1$ is on, fuel is normally pumped properly (pf).
- When fuel is pumped properly (pf), fuel is usually ok (f).
- When $sw2$ is on, oil is usually pumped (po).
- When oil is pumped (po), it usually works ok (o).
- When there is oil and fuel $(o) \wedge (f)$, usually the engine works ok (e).
- When there is fuel, oil, and heat $(o) \wedge (f) \wedge (h)$ then the engine is usually not ok ($\overline{e}$).
- When there is heat (h), normally there are oil problems ($\overline{o}$).
- When fuel is pumped (pf) and speed is low (l), then there are reasons to believe that the pump is clogged (pc).
- When $sw2$ is on, usually speed is low (l).
- When $sw3$ is on, usually fuel is ok (f).

Suppose we also know some particular facts: $sw1$, $sw2$, and $sw3$ are on, and there is heat (h). The knowledge of such an agent can be modeled by the argumentative theory Γ_{engine} shown in Figure 3. From the theory Γ_{engine}, the argument $\mathcal{A}$:e, with

$$
\begin{array}{rcl}
\mathcal{A} & = & \{\, (pf \leftarrow sw1), (po \leftarrow sw2), \\
 & & (f \leftarrow pf), (o \leftarrow po), (e \leftarrow f, o) \,\}
\end{array}
$$

can be inferred via $\mathrel{|\!\!\sim}_{Arg}$ by applying the inference rules Intro-NR twice (inferring $sw1$ and $sw2$), then Intro-RE twice (inferring $pf \leftarrow sw1$ and $po \leftarrow sw2$), then Intro-RE twice again to infer $f \leftarrow pf$ and $o \leftarrow po$, and finally Intro-RE once again to infer $e \leftarrow f, o$. In a similar way, arguments $\mathcal{B}$:$\overline{f}$, $\mathcal{C}$:$\overline{l}$, $\mathcal{D}$:f and $\mathcal{E}$:$\overline{e}$ can be derived via $\mathrel{|\!\!\sim}_{Arg}$, with

$$
\begin{array}{rcl}
\mathcal{B} & = & \{\, (pf \leftarrow sw1), (l \leftarrow sw2), \\
 & & (pc \leftarrow pf, l) \,\} \\
\mathcal{C} & = & \{\, (\overline{l} \leftarrow sw2, sw3) \,\} \\
\mathcal{D} & = & \{\, (\overline{l} \leftarrow sw2, sw3) \,\} \\
\mathcal{E} & = & \{\, (pf \leftarrow sw1), (po \leftarrow sw2), \\
 & & (f \leftarrow pf), (o \leftarrow po), \\
 & & (\overline{e} \leftarrow f, o, h) \,\}
\end{array}
$$

Note that the arguments $\mathcal{B}$:$\overline{f}$, and $\mathcal{E}$:$\overline{e}$, are *counterarguments* for the original argument $\mathcal{A}$:e, whereas $\mathcal{C}$:$\bar{l}$ and $\mathcal{D}$:f are counterarguments for $\mathcal{B}$:$\overline{f}$. In each of these cases, these counterarguments are also defeaters according to the specificity preference criterion [Simari and Loui, 1992]. Assuming such defeat relationship among arguments, the following formulæ can be inferred via $\mid\!\!\sim_{\mathcal{T}}$:

$1)\mathbf{T}_1^*(\mathcal{A})$:$e$	Intro-1D
$2)\mathbf{T}_2^*(\mathcal{B})$:$\overline{f}$	Intro-1D
$3)\mathbf{T}_3^*(\mathcal{C})$:$\bar{l}$	Intro-1D
$4)\mathbf{T}_4^*(\mathcal{D})$:$f$	Intro-1D
$5)\mathbf{T}_5^*(\mathcal{E})$:$\overline{e}$	Intro-1D
$6)\mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D}))$:$\overline{f}$	Intro-ND$, 3), 4)$
$7)\mathbf{T}_1^*(\mathcal{A}, \mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D})), \mathbf{T}_5^*(\mathcal{E}))$:$e$	Intro-ND$, 6)$
$8)\mathbf{T}_5^U(\mathcal{E})$:$\overline{e}$	Mark-Atom
$9)\mathbf{T}_1^D(\mathcal{A}, \mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D})), \mathbf{T}_5^U(\mathcal{E}))$:$e$	Mark-1D$, 8)$

Note that the formula obtained in step (7) has a final label associated with it, since it cannot be 'expanded' from previous formulæ. Hence, following Def. 5, we can conclude that e is not warranted.

4 The Power of Labels

Labelled Deduction Systems offer a wide range of possibilities for formalizing different aspects of computational models of natural argument. Next we will focus on those which we consider to be particularly relevant: detecting fallacies, formalizing argumentation in social contexts, weighing arguments in dialectical trees, and characterizing argument-based inference operators.

4.1 Detecting Fallacies

According to Hamblin, the classical definition of a fallacy is *"an argument that appears to be valid, but is not"* [Hamblin, 1970, p.12]. In more general terms, a fallacy is a general type of appeal (or category of argument) that resembles good reasoning, but some of their inference steps are not truth-preserving.[7] As pointed out in [Thompson, 2004], while we may say that an argument is "fallacious", or "commits a fallacy", the term "fallacy" does not refer to an argument, but to an error of some identifiable kind. All of the arguments that are guilty of committing that error may be said to be *instances* of that fallacy, so fallacies are strictly and classically considered to be *types* of arguments.

Detecting logical fallacies plays an important role in computational models of argument. In this context the most basic fallacy involves *"circular reasoning"*, or repetition of arguments in a dialogue (as this leads to infinite branches in dialectical trees). Such situation is explicitly avoided in most formal approaches to defeasible argumentation ($e.g.$, [García and Simari, 2004; Hunter, 2004]) by imposing this as a constraint in the definition of argument trees. Other approaches (e.g., [García and Simari, 2004]) consider avoiding those dialogue lines whenever conflict arises among arguments advanced by the proponent (resp. opponent) in a given dialogue line. In that context, the advanced argument provoking such conflict is considered fallacious. In other cases (such as [Kakas and Toni, 1999]), analogous situations are obtained as a by-product of the framework under certain constraints ($e.g.$, when characterizing well-founded semantics using an argument-based approach to logic programming, all proponent arguments in an argument tree turn out to be non-conflictive).

In our approach to argumentation using LDS, constraints upon formation of dialogue lines are given by the special condition VSTree($\mathcal{A}$, $\mathbf{T}_i^*$), which takes into account if a given dialectical label $\mathbf{T}_i^*$ can be used as a sub-tree in a more complex dialectical label rooted in set $\mathcal{A}$ of wffs, corresponding to the main argument at issue. Cycle detection as well as ill-formed dialogue lines (as defined in [García and Simari, 2004]) are captured by this condition VSTree. Formal results concerning which dialectical trees are valid in a given argumentative framework can also be better analyzed by different characterizations of this condition.

4.2 Formalizing Arguments in Social Contexts

LDS also provide a sound framework for modelling multiagent societies. As Gabbay points out [Gabbay, 1996, p.311], a label could be the name of a person (source) who put some proposition forward, along with some indicator of the reliability of that person as a source of data. In this context, LDS play a role in formalizing *source-based arguments* [Walton, 1998], $i.e.$ arguments whose evaluation depends not only on the structure of the inference used, but also on some assessment of the sources of the premises. Evaluation of source-based arguments is clearly important in the context of computational models of argumentation for multiagent systems. In a very interesting paper [Walton, 1999] Walton shows how LDS and multi-agent systems can be combined to evaluate argumentation that is source-based and depends on a credibility function. He also remarks that two of the most common forms of source-based arguments are appeal to expert opinion (or *ad verecundiam* argument) and personal attacks (or *ad hominem* argument). Although such types of argumentation have been acknowledged as informal fallacies, Walton states that both of them can be "quite reasonable in many cases", particularly in legal argumentation contexts. As Walton points out [Walton, 1999, p.66] *"LDS is a big step forward in the evaluation of* ad hominem *and* ad verecundiam *arguments, because it enables us to base our evaluation of such arguments on a label indicating a comparative assessment of the source of the propositions that were put forward"*.

The *LDS*$_{AR}$ framework can be naturally extended to formalize Walton's proposal, keeping at the same time the expressivity to capture the information involved in the dialectical analysis performed by a single agent. The labelled language $\mathcal{L}_{\mathsf{Arg}}$ in *LDS*$_{AR}$ can in turn be labelled ($e.g.$, with a label (Ag_i, c_i) denoting an agent's name Ag_i and some associated credibility degree c_i), defining a new labelling language $\mathcal{L}_{\mathsf{Ag}}$. Thus a labelled formula in the new language $(\mathcal{L}_{\mathsf{Ag}}, \mathcal{L}_{\mathsf{Arg}})$ could be as follows:

[7] Some authors ($e.g.$, Johnson [Johnson, 1995]) suggest that a fallacy should occur *"with sufficient frequency in discourse to warrant being baptized."*. An in-depth treatment of fallacies is outside the scope of this paper.

$$(john, 0.7){:}(\mathbf{T}_i^U(\mathcal{A}, \ldots){:}\alpha)$$

denoting that agent $john$ with a credibility degree of 0.7 has performed some dialectical analysis concluding that α is currently assumed as warranted belief, on the basis of a dialectical analysis stored in the label $\mathbf{T}_i^U(\mathcal{A}, \ldots)$. Suitable deduction rules could be defined in order to characterize conflicts among several agents in which their credibility could be a factor to consider in assessing the final outcome of a dialogue among them.

In [Amgoud *et al.*, 2000] it was underscored the importance of having a formal model of inter-agent dialogues for argument exchange by providing a precisely defined protocol for interaction. In [Rahwan *et al.*, 2003] it was also emphasized that an important challenge facing future research is the understanding of 'social' aspects of argument-based negotiation in agent societies, as *"there is still no generic formal theory that establishes a precise relationship between normative social behavior and the outcomes of communication processes."* Given its expressive power, we think that LDS could provide an adequate formal tool in the context of formalizing protocols and norm adoption, helping to achieve the above goals.

4.3 Pruning and Weighing Arguments in Dialectical Trees

Dialectical trees provide a way of exhaustively analyzing arguments and counterarguments. A problem with this setting is that dialectical trees can often be "too big" [Hunter, 2004] so that the use of some kind of pruning strategy is in order. There are several approaches to pruning the search space in dialectical trees. The most basic approach consists in applying $\alpha - \beta$ pruning, as illustrated in Figure 4. When analyzing a given argument, instead of computing all possible defeaters ($\star$) only a part of the dialectical tree needs to be explored in order to determine whether the root node (main argument at issue) is defeated or not. It must be noted that the rules that characterize the "$\underset{\mathcal{T}}{\mid\!\sim}$" relationship (Figure 2) are also based on this strategy, used when propagating marking in labels in a bottom-up fashion.

Recent research [Hunter, 2004] has been focused on analyzing the *impact* of argumentation. Such an impact depends on what an agent regards as important, which allows to characterize the *resonance* and *cost* of producing arguments and argument trees. To measure resonance in argument trees, the sum of the resonance of the arguments in the tree is taken into account, scaled by a discount function which increases going down the tree, so that arguments at a greater depth have a reduced net effect on the resonance of the tree. The first ideas underlying this approach can be found in [Besnard and Hunter, 2001], where the notion of *categoriser* is introduced. A categoriser is a mapping from dialectical trees to numbers. The resulting number is intended to capture the relative strength of an argument taking into account its defeaters, the defeaters for those defeaters, and so on. An example of categoriser provided in [Besnard and Hunter, 2001] is the following:

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

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

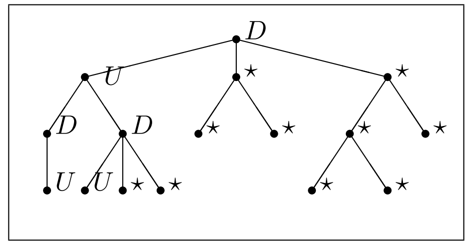

Figure 4: Labeling a dialectical tree with $\alpha - \beta$ pruning.

In the context of LDS_{AR}, assessing a weight to an argument on the basis of its defeaters can be performed in a natural way by suitably extending the criteria for labelling propagation. A function f could be defined to assign numbers to dialectical labels according to some particular criterion. Given a dialectical label, if $\mathcal{A}$ corresponds to an argument without defeaters it would be assigned a particular value $f(\mathcal{A})$. Otherwise, if $\mathcal{A}$ is the root node in a dialectical tree with label $\mathbf{T}(\mathcal{A}, \mathbf{T}_1, \ldots, \mathbf{T}_k)$, having as defeaters arguments $\mathcal{B}_1 \ldots, \mathcal{B}_k$, then f could be recursively defined as $f(\mathcal{A}){=}f(f(\mathbf{T}_1), \ldots f(\mathbf{T}_k))$ where $\mathbf{T}_1, \ldots, \mathbf{T}_k$ are the immediate subtrees (dialectical labels) associated with $\mathbf{T}$. In other words, numbers assigned to dialectical labels would be propagated bottom-up. Computing f can be thus defined in several ways ($e.g.$, as suggested in [Besnard and Hunter, 2001]). Such a setting allows to model a number of typical problems in defeasible argumentation, such as the the notion of *accrual of arguments* [Vreeswijk, 1997; Verheij, 1996], where arguments with many defeaters would be deemed weaker as those which have only one defeater.

4.4 Formalizing Argument-Based Consequence

As we have discussed before, LDS provide a way of formalizing complex frameworks as logical systems with labelled deduction capabilities. Different levels of inference ($e.g.$, restricted to certain kinds of labels) can be captured in terms of suitable non-monotonic *inference operators* which turn out to be a useful tool for both theoretical and practical goals. From a theoretical viewpoint logical properties of defeasible argumentation can be easier studied and formalized with such operators at hand. On the other hand, actual implementations of argumentation systems could benefit from emerging logical properties for more efficient argument-based computation in the context of real-world applications.

An LDS-based formalization of argumentation such as LDS_{AR} lends itself naturally towards the definition of such operators [Chesñevar and Simari, 2000; Chesñevar *et al.*, 2005]. Given an argumentative theory $\mathbf{\Gamma}$, the notion of "theorem" wrt $\vdash$ can be associated with a particular $C_{\vdash}$ operator, which computes all *empty* arguments, which account for the strict knowledge inferable from $\mathbf{\Gamma}$. Similarly, operators C_{arg} and C_{war} can be defined to compute the set of all arguments and warranted beliefs. Logical properties relevant for the study of non-defeasible inference (such as cummulativity, semi-monotonicity, superclassicality, etc. [Makinson, 1994]) can be better analyzed and contrasted in the light of such operators. Besides, it must be remarked that our LDS approach to argumentation does not stand as a single logic for argument, but rather as a 'family' of logics. Thus, a condition such as VSTree can be used as a parameter for characterizing different, alternative logics within the same logical framework [Chesñevar and Simari, 2001].

5 Related Work

Early work which used some of the principles present in LDS (but not as formally) was Cohen's theory of endorsements [Cohen, 1985]. Endorsements are symbolic representations of different items of evidence, the questions on which they bear, and the relations between them. Endorsements can operate on each other and hence lead to the retraction of conclusions previously reached. Research concerning aggregating arguments by incorporating numerical and symbolic features can be traced back to the work of Krause *et al.* [Elvang-Gøransson *et al.*, 1993; Krause *et al.*, 1995], where a uniform framework for reasoning with different kinds of *strength* in arguments is described. In particular, a characterization of defeasible reasoning using LDS is due to Hunter [Hunter, 1994], who showed how different non-monotonic logics can be characterized in terms of labelling strategies, algebras for labels, proof rules, and preference criteria. In contrast with our approach, Hunter had as a key aim to analyze the notion of preference among different non-monotonic logics. His approach, however, is also argument-based.

Multicontext systems [Giunchiglia *et al.*, 1993] have also been proposed as an alternative framework to LDS to provide contextual reasoning for agents. A context is a triple $C = \langle L, \omega, \Delta \rangle$, where L is a language ($e.g.$, first order logic), ω is the set of axioms for the context and Δ is the set of inference rules associated with the context C. A multicontext system is a pair $\langle C, B \rangle$, where $C = \{c_1, \ldots, c_n\}$ is the set of all contexts, and B is the set of *bridge rules* which have the form $c_1 : \phi_1 \ldots, c_k : \phi_k \rightarrow c_i : \phi_i$ standing for "if the wffs $\phi_1 \ldots, \phi_k$ are known to hold in contexts $c_1 \ldots, c_k$, then the wff ϕ_i will hold in context c_i. In [Parsons *et al.*, 1998] a multicontext approach to modelling argumentation among agents is presented. Different contexts represent different components in the agent architecture, and interactions between such components is specified by means of bridge rules between contexts. In [Grasso, 2004] an interesting approach to modelling rhetorical argument is presented, in which mental states in an arguer are characterized as a multicontext system $\langle B, R \rangle$, where B is a set of attitude contexts and R is a set of bridge rules among them. We contend that similar approaches to the ones mentioned above can be achieved in terms of the logical machinery provided by LDS. It must be noted that contexts themselves can be recast as elements in the labelling language. Thus, the context notation $B_A : X$ suggested in [Grasso, 2004] to denote "agent A believes X" can be seen as a labelled sentence, with B_A as associated label. Nesting beliefs is also possible in the LDS ontology, as suitable functors of the form $B_A(B_B)$ could be defined in the labelling language to denote situations like "agent A believes that agent B believes that...", which in [Grasso, 2004] are defined by nesting contexts.

6 Conclusions and Future Work

LDS offer a powerful tool for formalizing different aspects of computational models of natural argument. In particular, as we have outlined in this paper, LDS_{AR} provides a sound formal framework for modelling argument-based dialectical reasoning for an intelligent agent. The underlying argumentative logic for such an agent can be formally analyzed from the natural deduction rules that characterize it, providing a way of studying formal properties associated with such logics. We also showed that such framework can be parametrized with respect to a number of features (knowledge representation language, preconditions in natural deduction rules, etc.) which are unified in a single logical system. We have also shown why labels are a good alternative for coping with several issues relevant in modelling argumentation: detecting fallacies, considering arguments in social contexts, analyzing dialectical trees, and formalizing consequence operators.

We contend that several other issues related to computational models of natural argument which have not been explored in this paper ($e.g.$, argumentation protocols, resource-bounded reasoning, rhetorical capabilities, etc.) can be suitably modelled in terms of LDS by providing an appropriate ontology in which such notions can be 'abstracted away' as labels. We think that labels are also a good tool in the context of Semantic Web applications, as they can be naturally stored as pieces of structured XML code. On the other hand, the semantic annotation of web content may be stored as labels by means of an appropriate LDS. Different levels of granularity can also be better identified ($e.g.$, abstracting away particular sublabels), which might be useful for identifying *argumentation schemes* [Walton and Reed, 2002] as well as for integrating LDS-based knowledge into salient software tools for argument analysis (such as ARAUCARIA [Reed and Rowe, 2004]). In our opinion, exploiting such integration can offer promising results that can help in solving several open problems in computational models of argument. Research in this direction is currently being pursued.

Acknowledgments: The authors would like to thank anonymous reviewers for their suggestions to improve the original version of this paper. This research was funded by Agencia Nacional de Promoción Científica y Tecnológica (PICT 13096, PICT 15043, PAV 076), by CONICET (Argentina), by projects TIC2003-00950 and TIN2004-07933-C03-03 (MCyT, Spain) and by Ramón y Cajal Program (MCyT, Spain).

References

[Amgoud *et al.*, 2000] L. Amgoud, N. Maudet, and S. Parsons. Modeling dialogues using argumentation. In *ICMAS*, pages 31–38, 2000.

[Amgoud *et al.*, 2004] L. Amgoud, C. Cayrol, and M. Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *Proc. of 10th Intl NMR Workshop. Whistler, Canada*, pages 1–9, 2004.

[Besnard and Hunter, 2001] P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artif. Intell.*, 128(1-2):203–235, 2001.

[Bondarenko *et al.*, 1997] A. Bondarenko, P. Dung, R. Kowalski, and F. Toni. An abstract, argumentationtheoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.

[Chesñevar and Simari, 2000] C. Chesñevar and G. Simari. Formalizing Defeasible Argumentation using Labelled Deductive Systems. *Journal of Computer Science & Technology*, 1(4):18–33, 2000.

[Chesñevar and Simari, 2001] C. Chesñevar and G. Simari. Consequence Operators for Defeasible Argumentation. In *Proc. VII Argentinean Conf. in Computer Science*, pages 309–320, October 2001.

[Chesñevar *et al.*, 2000] C. Chesñevar, A. Maguitman, and R. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.

[Chesñevar *et al.*, 2005] C. Chesñevar, G. Simari, L. Godo, and T. Alsinet. Argument-based expansion operators in possibilistic defeasible logic programming: Characterization and logical properties. In *Proc. 8th ECSQARU Conf., Barcelona, Spain (in press)*, 2005.

[Cohen, 1985] P. Cohen. *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach*. London, Pitman, 1985.

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

[Elvang-Gøransson *et al.*, 1993] M. Elvang-Gøransson, P. Krause, and J. Fox. Dialectic reasoning with inconsistent information. In *Proc. of the 9th UAI, Washington, USA*, pages 114–121, 1993.

[Gabbay *et al.*, 2004] D. Gabbay, L. Lamb, and A. Russo. *Compiled Labelled Deductive Systems : A Uniform Presentation of Non-Classical Logics*. Inst of Physics Pub. Inc. Series: Studies in Logic and Computation, 2004.

[Gabbay, 1996] D. Gabbay. *Labelling Deductive Systems (vol.1)*. Oxford University Press (Volume 33 of Oxford Logic Guides), 1996.

[García and Simari, 2004] A. García and G. Simari. Defeasible Logic Programming: An Argumentative Approach. *Theory and Practice of Logic Prog.*, 4(1):95–138, 2004.

[Giunchiglia *et al.*, 1993] F. Giunchiglia, L. Serafini, E. Giunchiglia, and M. Frixione. Non-omniscient belief as context-based reasoning. In *Proc. of 13th IJCAI Conf., Chambery, France*, pages 548–554, 1993.

[Grasso, 2004] F. Grasso. A Mental Model for a Rhetorical Arguer. In R. Young F. Schmalhofer and G. Katz, editors, *Proc. of the European Cognitive Science Society Conf.*, June 2004.

[Hamblin, 1970] C. L. Hamblin. *Fallacies*. Methuen, London, 1970.

[Hunter, 1994] A. Hunter. Defeasible Reasoning with structured information. In E Sandewall J Doyle and P Torasso, editors, *Proc. 4th KR*, pages 281–292. M.Kaufmann, 1994.

[Hunter, 2004] A. Hunter. Towards Higher Impact Argumentation. In *Proc. 19th AAAI Conf.*, pages 275–280, 2004.

[Johnson, 1995] R. Johnson. The blaze of her splendors. In H. Hansen and R. Pinto, editors, *Fallacies: Classical and Contemporary Readings*. Penn State University Press, 1995.

[Kakas and Toni, 1999] A. Kakas and F. Toni. Computing argumentation in logic programming. *Journal of Logic Programming*, 9(4):515:562, 1999.

[Krause *et al.*, 1995] P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox. A logic of argumentation for reasoning under uncertainty. *Comput. Intell.*, 11:113–131, 1995.

[Makinson, 1994] D. Makinson. General patterns in nonmonotonic reasoning. In D.Gabbay, C.Hogger, and J.Robinson, editors, *Handbook of Logic in Art. Int. and Logic Prog.*, volume Nonmonotonic and Uncertain Reasoning, pages 35–110. Oxford University Press, 1994.

[Parsons *et al.*, 1998] S. Parsons, C. Sierrra, and N. Jennings. Agents that Reason and Negotiate by Arguing. *Journal of Logic and Computation*, 8:261–292, 1998.

[Pollock, 1991] J. Pollock. A theory of defeasible reasoning. *Intl. Journal of Intelligent Systems*, 6:33–54, 1991.

[Prakken and Sartor, 1997] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics*, 7:25–75, 1997.

[Prakken and Vreeswijk, 2002] H. Prakken and G. Vreeswijk. Logical Systems for Defeasible Argumentation. In D. Gabbay and F.Guenther, editors, *Handbook of Phil. Logic*, pages 219–318. Kluwer, 2002.

[Rahwan *et al.*, 2003] I. Rahwan, S. Ramchurn, N. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation-based negotiation. *Knowl. Eng. Rev.*, 18(4):343–375, 2003.

[Reed and Rowe, 2004] C. Reed and G. Rowe. Araucaria: Software for argument analysis, diagramming and representation. *Intl. J. of AI Tools*, 14(3-4):961–980, 2004.

[Simari and Loui, 1992] G. Simari and R. Loui. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence*, 53:125–157, 1992.

[Thompson, 2004] B. Thompson. What is a fallacy. http://www.cuyamaca.net/bruce.thompson/, 2004.

[Verheij, 1996] B. Verheij. *Rules, Reasons, Arguments: formal studies of argumentation and defeat*. PhD thesis, Maastricht University, Holland, December 1996.

[Vreeswijk, 1997] G. Vreeswijk. Abstract argumentation systems. *Artif. Intell.*, 90(1-2):225–279, 1997.

[Walton and Reed, 2002] D. Walton and C. Reed. Argumentation schemes and defeasible inferences. In *Proc. ECAI'2002, CMNA Workshop*, pages 45–55, 2002.

[Walton, 1998] D. Walton. *Ad hominem Arguments*. Univ. of Alabama Press, Tuscaloosa, 1998.

[Walton, 1999] D. Walton. Applying labelled deductive systems and multi-agent systems to source-based argumentation. *J. of Logic and Computation*, 9(1):63–80, feb 1999.