A Framework for Combining Defeasible Argumentation with Labeled Deduction

Carlos Iván Chesñevar[1] **Guillermo Ricardo Simari**[2]

[1]Artificial Intelligence Research Group — Departament of Computer Science
Universitat de Lleida – Campus Cappont – C/Jaume II, 69 – E-25001 Lleida, SPAIN
TEL/FAX: (+34) (973) 70 2764 / 2702 – EMAIL: cic@eup.udl.es

[2]Artificial Intelligence Laboratory — Dep. of Computer Science and Enginering
Universidad Nacional del Sur – Alem 1253 – B8000CPB Bahía Blanca, ARGENTINA
TEL/FAX: (+54) (291) 459 5135/5136 – EMAIL: grs@cs.uns.edu.ar

KEY WORDS: Defeasible Argumentation, Defeasible Reasoning, Labelled Deduction

Abstract. In the last years, there has been an increasing demand of a variety of logical systems, prompted mostly by applications of logic in AI and other related areas. *Labeled Deductive Systems* (LDS) were developed as a flexible methodology to formalize such a kind of complex logical systems.
Defeasible argumentation has proven to be a successful approach to formalizing commonsense reasoning, encompassing many other alternative formalisms for defeasible reasoning. Argument-based frameworks share some common notions (such as the concept of argument, defeater, etc.) along with a number of particular features which make it difficult to compare them with each other from a logical viewpoint.
This paper introduces LDS_{AR}, a LDS for defeasible argumentation in which many important issues concerning defeasible argumentation are captured within a unified logical framework. We also discuss some logical properties and extensions that emerge from the proposed framework.

1 Introduction and motivations

Labeled Deductive Systems (LDS) [Gab96] were developed as a rigorous but flexible methodology to formalize complex logical systems, such as temporal logics, database query languages and defeasible reasoning systems. In labeled deduction, the usual notion of formula is replaced by the notion of *labeled formula*, expressed as *Label*:*f*, where *Label* represents a label associated with a wff *f*. A labeling language $\mathcal{L}_{\mathsf{Labels}}$ and knowledge-representation language $\mathcal{L}_{\mathsf{KR}}$ can be combined to provide a new, labeled language, in which labels convey additional information also encoded at object-language level. Derived formulas are labeled according to a family of *deduction rules*, and with agreed ways of propagating labels via the application of these rules.

In the last decade *defeasible argumentation* [CML00,PV99] has proven to be a successful approach to formalizing commonsense reasoning, providing a suitable formalization that encompasses many other alternative formalisms. Thus, most argument-based frameworks share some common notions (such as the concept of argument, defeater, warrant, etc.) along with a number of particular features which make it difficult to compare them with each other from a logical viewpoint.

The study of logical properties of *defeasible argumentation* motivated the development of LDS_{AR} [Che01,SCG01,CS01], an LDS-based argumentation formalism. In LDS_{AR} two languages $\mathcal{L}_{\text{Labels}}$ (representing arguments and their interrelationships) and $\mathcal{L}_{\text{KR}}$ (representing object-level knowledge) are combined into a single, labeled language $\mathcal{L}_{\text{Arg}}$. Inference rules are provided in $\mathcal{L}_{\text{Arg}}$ to characterize argument construction and their relationships. LDS_{AR} provides thus a common framework for different purposes, such as studying logical properties of defeasible argumentation, comparing and analyzing existing argument-based frameworks and developing extensions of the original framework by enriching the labeling language $\mathcal{L}_{\text{Labels}}$.

This paper is structured as follows. First, in section 2 we discuss the main definitions and concepts associated with the LDS_{AR} framework. In Section 3 we present some logical properties that hold in LDS_{AR}, and show how different alternative argument-based formalisms can be seen as particular instances of the proposed framework. Then in Section 4 we discuss some particular issues relating LDS_{AR} to modeling scientific reasoning, such as comparing top-down vs. bottom-up computation of warrant, and the combination of qualitative and quantitative reasoning by incorporating numerical attributes. Finally Section 5 summarizes related work as well as the main conclusions that have been obtained.

2 The LDS_{AR} framework: fundamentals[1]

2.1 Knowledge representation

We will first introduce a *knowledge representation language* $\mathcal{L}_{\text{KR}}$ together with a *labeling language* $\mathcal{L}_{\text{Labels}}$. These languages will be used to define the object language $\mathcal{L}_{\text{Arg}}$. Following [Gab96], labeled wffs in $\mathcal{L}_{\text{Arg}}$ will be called *declarative units*, having the form *Label:wff*.

Definition 1 (Language $\mathcal{L}_{\text{KR}}$. Wffs in $\mathcal{L}_{\text{KR}}$). *The language $\mathcal{L}_{\text{KR}}$ will be composed of propositional atoms (a, b, ...) and the logical connectives $\wedge$, $\sim$ and $\leftarrow$. If α is an atom in $\mathcal{L}_{\text{KR}}$, then α and $\sim\alpha$ are wffs called* literals *in $\mathcal{L}_{\text{KR}}$. If α_1, $\ldots\alpha_k$, β are literals in $\mathcal{L}_{\text{KR}}$, then $\beta \leftarrow \alpha_1, \ldots \alpha_k$ is a wff in $\mathcal{L}_{\text{KR}}$ called* rule.

The language $\mathcal{L}_{\text{KR}}$ is a Horn-like propositional language restricted to *rules* and *facts*.[2] Labels in the language $\mathcal{L}_{\text{Labels}}$ can be either *basic* or *complex*. Basic labels distinguish between defeasible and non-defeasible information, whereas complex labels account for *arguments* (a tentative proof involving defeasible information) and *dialectical trees* (a tree-like structure rooted in a given argument).

[1] For space reasons we only give a summary of the main elements of the $LDS_{\boldsymbol{AR}}$ framework; for an in-depth treatment see [Che01,CS01]). We also assume that the reader has a basic knowledge about defeasible argumentation formalisms [CML00,PV99].

[2] The language $\mathcal{L}_{\text{KR}}$ is similar to the language of *extended logic programming* in a propositional setting.

Definition 2 (Labeling language $\mathcal{L}_{\mathsf{Labels}}$). *The labeling 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 $\emptyset$. This is a* basic label *is associated with every wff which corresponds to non-defeasible knowledge.*
2. *A single wff f in $\mathcal{L}_{\mathsf{KR}}$. This is a* basic label *which corresponds to f as a piece of defeasible knowledge.*
3. *A set $\Phi \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$. This is a complex label called* argument label.
4. *A tree-like structure $\mathbf{T}$ is a complex label called* dialectical label*, being 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 $\mathcal{L}_{\mathsf{Labels}}$. For the sake of simplicity, we will write $\mathbf{T}_k$ to denote an arbitrary dialectical label.*
   (b) *If $\mathbf{T}_1, \ldots, \mathbf{T}_k$ are dialectical labels, then $\mathbf{T}^U_n(\mathbf{T}_1, \ldots, \mathbf{T}_k)$, $\mathbf{T}^*_n(\mathbf{T}_1, \ldots, \mathbf{T}_k)$ and $\mathbf{T}^D_m(\mathbf{T}_1, \ldots, \mathbf{T}_k)$ will also be dialectical labels in $\mathcal{L}_{\mathsf{Labels}}$.*
5. *Nothing else is a label in $\mathcal{L}_{\mathsf{Labels}}$.*

The *object (labeled) language* in LDS_{AR} is defined as $\mathcal{L}_{\mathsf{Arg}} = (\mathcal{L}_{\mathsf{Labels}}, \mathcal{L}_{\mathsf{KR}})$. Since $\mathcal{L}_{\mathsf{KR}}$ is a Horn-like logic language, we will assume an underlying inference mechanism $\vdash_{SLD}$ equivalent to SLD resolution [Llo87], properly extended to handle a negated literal $\sim p$ as a new constant name *no_p*. Given $P \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$, we write $P \vdash_{SLD} \alpha$ to denote that α follows from P via $\vdash_{SLD}$.

Definition 3 (Contradictory set of wffs in $\mathcal{L}_{\mathsf{KR}}$). *Given a set S of wffs in $\mathcal{L}_{\mathsf{KR}}$, S will be called a* contradictory set *(denoted $S \vdash_{SLD} \bot$) iff complementary literals p and $\sim p$ can be derived from S via $\vdash_{SLD}$.*

Basic declarative units will be used to encode defeasible and non-defeasible information available for an intelligent agent to reason from a set Γ of labeled wffs. Such a set will be called *argumentative theory.* Formally:

Definition 4 (Basic declarative units. Argumentative theory). *A labeled wff $\psi{:}\alpha$ such that α is a basic label (either (1) $\psi = \emptyset$ or (2) $\psi = \alpha$) will be called a* basic declarative unit *(bdu). In case (1), the wff $\emptyset{:}\alpha$ will be called a* non-defeasible *bdu; in case (2), the wff $\alpha{:}\alpha$ will be called a* defeasible *bdu. A finite set $\Gamma = \{\ \gamma_1,\ \gamma_2,\ \ldots,\ \gamma_k\}$ of bdu's will be called an* argumentative theory*. For every argumentative theory Γ we will assume that the set of non-defeasible formulas $\mathbf{\Pi}(\Gamma) = \{\emptyset{:}\alpha \mid \emptyset{:}\alpha \in \Gamma\}$ is non-contradictory.*

Formulas with an empty label correspond to 'strict' knowledge. Thus, $\emptyset{:}p$ and $\emptyset{:}p \leftarrow q$ stand for a fact p and a logic programming clause $p \leftarrow q$. Defeasible facts (also known as *presumptions*) and defeasible rules are represented by formulas $\{p\}]{:}p$ and $\{p \leftarrow q\}{:}p \leftarrow q$. Thus, the classical default "Birds typically fly" will be represented in LDS_{AR} as $\{fly \leftarrow bird\}{:}fly \leftarrow bird$, whereas the strict rule "Penguins don't fly" will be represented in LDS_{AR} as $\emptyset{:}\sim fly \leftarrow penguin$. Intuitively, the label of a bdu stands for an initial set of support associated with a formula in the argumentative theory, and is used for consistency check when performing inferences, as discussed later in Sec. 2.2.

$\emptyset$:$\sim$*fuel_ok* $\leftarrow$ *pump_clog*
$\emptyset$:*sw1* $\leftarrow$
$\emptyset$:*sw2* $\leftarrow$
$\emptyset$:*sw3* $\leftarrow$
$\emptyset$:*heat* $\leftarrow$
$\{$*pump_fuel_ok* $\leftarrow$ *sw1*$\}$:*pump_fuel_ok* $\leftarrow$ *sw1*
$\{$*fuel_ok* $\leftarrow$ *pump_fuel_ok*$\}$:*fuel_ok* $\leftarrow$ *pump_fuel_ok*
$\{$*pump_oil_ok* $\leftarrow$ *sw2*$\}$:*pump_oil_ok* $\leftarrow$ *sw2*
$\{$*oil_ok* $\leftarrow$ *pump_oil_ok*$\}$:*oil_ok* $\leftarrow$ *pump_oil_ok*
$\{$*engine_ok* $\leftarrow$ *fuel_ok*, *oil_ok*$\}$:*engine_ok* $\leftarrow$ *fuel_ok*, *oil_ok*
$\{\sim$*engine_ok* $\leftarrow$ *fuel_ok*, *oil_ok*, *heat*$\}$:$\sim$*engine_ok* $\leftarrow$ *fuel_ok*, *oil_ok*, *heat*
$\{\sim$*oil_ok* $\leftarrow$ *heat*$\}$:$\sim$*oil_ok* $\leftarrow$ *heat*
$\{$*pump_clog* $\leftarrow$ *pump_fuel_ok*, *low_speed*$\}$:*pump_clog* $\leftarrow$ *pump_fuel_ok*, *low_speed*
$\{$*low_speed* $\leftarrow$ *sw2*$\}$:*low_speed* $\leftarrow$ *sw2*
$\{\sim$*low_speed* $\leftarrow$ *sw2*, *sw3*$\}$:$\sim$*low_speed* $\leftarrow$ *sw2*, *sw3*
$\{$*fuel_ok* $\leftarrow$ *sw3*$\}$:*fuel_ok* $\leftarrow$ *sw3*

Fig. 1. Argumentative theory Γ_{engine} (example 1)

Example 1. Consider an agent involved in controlling an engine with three switches $sw1$, $sw2$ and $sw3$. These switches regulate different features of the engine, such as pumping system, speed, etc. Suppose we have defeasible information about how this engine works.

– If the pump is clogged, then the engine gets no fuel.
– When $sw1$ is on, normally fuel is pumped properly.
– When fuel is pumped properly, fuel usually works ok.
– When $sw2$ is on, usually oil is pumped.
– When oil is pumped, usually it works ok.
– When there is oil and fuel, usually the engine works ok.
– When there is fuel, oil, and heat, then the engine is usually not ok.
– When there is heat, normally there are oil problems.
– When fuel is pumped and speed is low, there are reasons to believe that the pump is clogged.
– When $sw2$ is on, usually speed is low.
– When $sw3$ is on, usually fuel is ok.

Suppose we know $sw1$, $sw2$ and $sw3$ are on, and there is heat. This situation can be modeled by the argumentative theory Γ_{engine} shown in figure 1. □

2.2 Argument construction

Given an argumentative theory Γ, and a wff $p \in \mathcal{L}_{\mathsf{KR}}$, the inference process in LDS_{AR} involves first obtaining a tentative proof (or *argument*) for p. A consequence relation $\mid\!\sim_{Arg}$ propagates labels, implementing the SLD resolution procedure along with a consistency check every time new defeasible information is

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

2. Intro-RE: $\dfrac{\mathbf{\Pi}(\Gamma) \cup \Phi \not\vdash_{SLD} \bot}{\Phi{:}\alpha}$ for any $\Phi{:}\alpha$

3. Intro-$\wedge$: $\dfrac{\Phi_1{:}\alpha_1 \quad \Phi_2{:}\alpha_2 \quad \ldots \Phi_k{:}\alpha_k \quad \mathbf{\Pi}(\Gamma) \cup \bigcup_{i=1\ldots k} \Phi_i \not\vdash_{SLD} \bot}{\bigcup_{i=1\ldots k} \Phi_i{:}\alpha_1, \alpha_2, \ldots, \alpha_k}$

4. Elim-$\leftarrow$: $\dfrac{\Phi_1{:}\beta\leftarrow\alpha_1, \ldots, \alpha_k \quad \Phi_2{:}\alpha_1, \ldots, \alpha_k \quad \mathbf{\Pi}(\Gamma) \cup \Phi_1 \cup \Phi_2 \not\vdash_{SLD} \bot}{\Phi_1 \cup \Phi_2{:}\beta}$

Fig. 2. Inference rules for $\mid\!\sim_{Arg}$: deriving (generalized) arguments in LDS_{AR}

introduced in a proof. Figure 2 summarizes the natural deduction rules which characterize the inference relationship $\mid\!\sim_{Arg}$. Rules Intro-NR and Intro-RE allow the introduction of non-defeasible and defeasible information in a proof, respectively. Rules Intro-$\wedge$ and Elim-$\leftarrow$ stand for introducing conjunction and applying modus ponens. In the last three rules, a consistency check is performed in order to ensure that the label $\mathcal{A}$ together with $\mathbf{\Pi}(\Gamma)$ does not derive complementary literals, avoiding logical contradiction. Note that the label $\mathcal{A}$ associated with a formula $\mathcal{A}{:}h$ contains all *defeasible* information needed to conclude h from Γ. Thus, arguments in LDS_{AR} are modeled as labeled formulas $\mathcal{A}{:}h$, where $\mathcal{A}$ stands for a set of (ground) defeasible rules that along with $\mathbf{\Pi}(\Gamma)$ derive h.

Definition 5 (Argument. Subargument). *Let Γ be an argumentative theory, and let h be a literal such that $\Gamma \mid\!\sim_{Arg} \mathcal{A}{:}h$ Then $\mathcal{A}$ will be called a* generalized argument *for h. If it is not the case that $\Gamma \mid\!\sim_{Arg} \mathcal{B}{:}h$, with $\mathcal{B} \subset \mathcal{A}$, then $\mathcal{A}{:}h$ is called a* minimal argument *or just* argument. *If $\Gamma \mid\!\sim_{Arg} \mathcal{A}{:}h$, and $\mathcal{A}{:}h$ is an argument, we will also say that $\mathcal{A}{:}h$ is an argument* based on Γ
An argument $\mathcal{A}{:}h$ is a subargument *of another argument $\mathcal{B}{:}q$ if $\mathcal{A} \subset \mathcal{B}$.*

Example 2. Consider the argumentative theory Γ_{engine} from example 1. Then the argument $\mathcal{A}{:}engine\_ok$, with $\mathcal{A} = \{(pump\_fuel\_ok \leftarrow sw1), (pump\_oil\_ok \leftarrow sw2), (fuel\_ok \leftarrow pump\_fuel\_ok), (oil\_ok \leftarrow pump\_oil\_ok), (engine\_ok \leftarrow fuel\_ok, oil\_ok)\}$ can be inferred via $\mid\!\sim_{Arg}$ by applying the inference rules Intro-NR twice (inferring $sw1$ and $sw2$), then Intro-RE twice (inferring $pump\_fuel\_ok \leftarrow sw1$ and $pump\_oil\_ok \leftarrow sw2$), then Intro-RE twice again to infer $fuel\_ok \leftarrow pump\_fuel\_ok$ and $oil\_ok \leftarrow pump\_oil\_ok$, and finally Intro-RE once again to infer $engine\_ok \leftarrow fuel\_ok, oil\_ok$. Similarly, arguments $\mathcal{B}{:}{\sim}fuel\_ok$, $\mathcal{C}{:}{\sim}low\_speed$, $\mathcal{D}{:}fuel\_ok$ and $\mathcal{E}{:}{\sim}engine\_ok$ can be derived via $\mid\!\sim_{Arg}$, with

$\mathcal{A} = \{\ (pump\_fuel\_ok \leftarrow sw1),\ (pump\_oil\_ok \leftarrow sw2),\ (fuel\_ok \leftarrow pump\_fuel\_ok),\ (oil\_ok \leftarrow pump\_oil\_ok),\ (engine\_ok \leftarrow fuel\_ok, oil\_ok)\ \}$

$\mathcal{B} = \{\ (pump\_fuel\_ok \leftarrow sw1),\ (low\_speed \leftarrow sw2),\ (pump\_clog \leftarrow pump\_fuel\_ok, low\_speed)\ \}$

1. Intro-1D:

$$\frac{\mathcal{A}{:}h \qquad \mathsf{Minimal}(\mathcal{A}{:}h)}{\mathbf{T}^*(\mathcal{A}){:}h}$$

2. Intro-ND:

$$\frac{\mathbf{T}^*(\mathcal{A}){:}h \qquad \mathbf{T}_1^*(\mathcal{B}_1,\ldots){:}q_1 \quad \mathbf{T}_k^*(\mathcal{B}_k,\ldots){:}q_k \qquad \mathsf{VSTree}(\mathcal{A},\ \mathbf{T}_i^*),\ i=1\ldots k}{\mathbf{T}^*(\mathcal{A},\mathbf{T}_1^*,\ldots,\mathbf{T}_k^*){:}h}$$

3. Mark-Atom:

$$\frac{\mathbf{T}^*(\mathcal{A}){:}h}{[\mathbf{T}^U(\mathcal{A})]{:}h}$$

4. Mark-1D: for some $\mathbf{T}_i^*$, $i=1\ldots k$

$$\frac{[\mathbf{T}^*(\mathcal{A},\mathbf{T}_1^*,\ldots,\mathbf{T}_i^*,\ldots,\mathbf{T}_k)]{:}h \qquad [\mathbf{T}_i^U(\mathcal{B}_i\ldots)]{:}q_i \qquad \mathsf{VSTree}(\mathcal{A},\mathbf{T}_i^U)}{[\mathbf{T}^D(\mathcal{A},\mathbf{T}_1^*,\ldots,\mathbf{T}_{i-1}^*,\mathbf{T}_i^U,\mathbf{T}_{i+1}^*,\ldots,\mathbf{T}_k^*)]{:}h}$$

5. Mark-ND: For all $\mathbf{T}_i^*$, $i=1\ldots k$

$$\frac{[\mathbf{T}^*(\mathcal{A},\mathbf{T}_1^*,\ldots,\mathbf{T}_i^*,\ldots,\mathbf{T}_k^*)]{:}h \qquad [\mathbf{T}_i^D(\mathcal{B}_i,\ldots)]{:}q_i \qquad \mathsf{VSTree}(\mathcal{A},\mathbf{T}_i^D)}{[\mathbf{T}^U(\mathcal{A},\mathbf{T}_1^D,\ldots,\mathbf{T}_i^D,\ldots,\mathbf{T}_k^D)]{:}h}$$

Fig. 3. Rules for building dialectical trees in LDS^*_{AR}

$\mathcal{C} = \{\ (\sim low\_speed\ \leftarrow\ sw2, sw3)\ \}$
$\mathcal{D} = \{\ (\sim low\_speed\ \leftarrow\ sw2, sw3)\ \}$
$\mathcal{E} = \{\ (pump\_fuel\_ok\ \leftarrow\ sw1),\ (pump\_oil\_ok\ \leftarrow\ sw2),\ (fuel\_ok\ \leftarrow\ pump\_fuel\_ok),\ (oil\_ok\ \leftarrow\ pump\_oil\_ok),\ (\sim engine\_ok\ \leftarrow\ fuel\_ok, oil\_ok, heat)\ \}$

□

2.3 Defeat among Arguments. Warrant

Given an argument $\mathcal{A}$:h based on an argumentative theory Γ, there may exist other conflicting arguments based on Γ that *defeat* it. Conflict among arguments is captured by the notion of contradiction (def. 3).

Definition 6 (Counterargument). *Let Γ be an argumentative theory, and let $\mathcal{A}$:h and $\mathcal{B}$:q be arguments based on Γ. Then $\mathcal{A}$:h* counter-argues *$\mathcal{B}$:q if there exists a subargument $\mathcal{B}'$:s of $\mathcal{B}$:q such that $\mathbf{\Pi}(\Gamma) \cup \{h, s\}$ is contradictory. The argument $\mathcal{B}'$:s will be called* disagreement subargument*.*

Defeat among arguments involves a partial order which establishes a *preference criterion* on conflicting arguments. A common preference criterion is specificity [SL92,SGCS03], which favors an argument with greater information content and/or less use of defeasible rules.

Definition 7 (Preference order $\preceq$). *Let Γ be an argumentative theory, and let* $\mathsf{Args}(\Gamma)$ *be the set of arguments that can be obtained from Γ. A* preference order $\preceq \subseteq \mathsf{Args}(\Gamma) \times \mathsf{Args}(\Gamma)$ *is any partial order on* $\mathsf{Args}(\Gamma)$.

Definition 8 (Defeat). *Let Γ be an argumentative theory, such that $\Gamma \mid\!\sim_{Arg} \mathcal{A}{:}h$ and $\Gamma \mid\!\sim_{Arg} \mathcal{B}{:}q$. We will say that $\mathcal{A}{:}h$* defeats *$\mathcal{B}{:}q$ (or equivalently $\mathcal{A}{:}h$ is a* defeater *for $\mathcal{B}{:}q$) if*

1. *$\mathcal{A}{:}h$ counterargues $\mathcal{B}{:}q$, with disagreement subargument $\mathcal{B}'{:}q'$.*
2. *Either it holds that $\mathcal{A}{:}h \succ \mathcal{B}'{:}q'$, or $\mathcal{A}{:}h$ and $\mathcal{B}'{:}q'$ are unrelated by the preference order "$\preceq$".*

Example 3. Consider the argumentative theory from example 1. Note that the arguments $\mathcal{B}{:}{\sim}fuel\_ok$, and $\mathcal{E}{:}{\sim}engine\_ok$, are counter-arguments for the original argument $\mathcal{A}{:}engine\_ok$, whereas $\mathcal{C}{:}{\sim}low\_speed$ and $\mathcal{D}{:}fuel\_ok$ are counter-arguments for $\mathcal{B}{:}{\sim}fuel\_ok$. In each of these cases, counter-arguments are also defeaters according to the specificity preference criterion [SL92].

Since defeaters are arguments, there may exist defeaters for the defeaters and so on. That prompts for a complete dialectical analysis to determine which arguments are ultimately defeated.

Definition 9 (Dialectical Tree). *Let $\mathcal{A}$ be an argument for q. The dialectical tree for $\mathcal{A}{:}q$, denoted $\mathcal{T}_{\mathcal{A}:q}$, is recursively defined as follows:*

1. *A single node labeled with an argument $\mathcal{A}{:}q$ with no defeaters is by itself the dialectical tree for $\mathcal{A}{:}q$.*
2. *Let $\mathcal{A}_1{:}q_1, \mathcal{A}_2{:}q_2, \ldots, \mathcal{A}_n{:}q_n$ be all the defeaters for $\mathcal{A}{:}q$. We construct the dialectical tree for $\mathcal{A}{:}q$, $\mathcal{T}_{\mathcal{A}:q}$, by labeling the root node with $\mathcal{A}{:}q$ and by making this node the parent node of the roots of the dialectical trees for $\mathcal{A}_1{:}q_1, \mathcal{A}_2{:}q_2, \ldots, \mathcal{A}_n{:}q_n$.*

Note: in order to avoid *fallacious argumentation* [SCG94], some additional constraints not given in Def. 9 are imposed on every path (e.g. there can be no repeated arguments, as this would lead to circular argumentation).[3]

A dialectical tree resembles a *dialogue tree* between two parties, proponent and opponent. Branches of the tree correspond to exchange of arguments between these two parties. A dialectical tree can be marked as an AND-OR tree [Gin93] according to the following procedure: nodes with no defeaters (leaves) are marked as U-nodes (undefeated nodes). 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 D-node. Formally:

Definition 10 (Marking of the Dialectical Tree). *Let $\mathcal{A}{:}q$ be an argument and $\mathcal{T}_{\mathcal{A}:q}$ its dialectical tree, then:*

[3] An in-depth analysis is outside the scope of this paper. See [CML00,SCG94] for details.

1. *All the leaves in $\mathcal{T}_{\mathcal{A}:q}$ are labeled as U-nodes.*
2. *Let $\mathcal{B}$:h be an inner node of $\mathcal{T}_{\mathcal{A}:q}$. Then $\mathcal{B}$:h will be a U-node iff every child of $\mathcal{B}$:h is a D-node. The node $\mathcal{B}$:h will be a D-node iff it has at least one child marked as U-node.*

After performing the above dialectical analysis, an argument $\mathcal{A}$ which turns to be ultimately undefeated is called a *warrant*. Formally:

Definition 11 (Warrant). *Let $\mathcal{A}$:q be an argument and $\mathcal{T}_{\mathcal{A}:q}$ its associated dialectical tree, such that its root node $\mathcal{A}$:q is marked as U. Then $\mathcal{A}$:q is called a* warranted argument *or just* warrant

In the context of LDS_{AR}, the construction and marking of dialectical trees is captured in terms of *dialectical labels* (Def. 2). Special marks (*, U, D) are associated with the a label $\mathbf{T}(\mathcal{A}, \ldots)$ in order to determine whether $\mathcal{A}$ correspond to an *unmarked*, *defeated* or *undefeated* argument, resp. In the theory of defeasible argumentation, a warranted argument or belief will be that one which is ultimately accepted at some time of the dialectical process. In LDS_{AR} the concept of warrant can be formalized as follows:

Definition 12 (Warrant – Version 1). *Let $Cn_*^k(\Gamma)$ be the set of all dialectical formulas that can be obtained from Γ via $\mid\!\sim_{\mathcal{T}}$ by at most k applications of inference rules ($i <= k$). A literal h is said to be* warranted *iff $\mathbf{T}^U(\mathcal{A}, ...)$:$h \in Cn_*^k(\Gamma)$, and there is no $k' > k$, such that $\mathbf{T}^D(\mathcal{A}, ...)$:$h \in (Cn_*^{k'}(\Gamma) \setminus Cn_*^k(\Gamma))$.*

This approach resembles Pollock's original ideas of (ultimately) justified belief [Pol95]. Note that Def. 12 forces to compute the closure under $\mid\!\sim_{\mathcal{T}}$in order to determine whether a literal is warranted or not. Fortunately this is not the case, since warrant 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 label that cannot be further extended.

Definition 13 (Warrant – Version 2). [4] *Let Γ be an argumentative theory, such that $\Gamma \mid\!\sim_{\mathcal{T}} \mathbf{T}_i^U(\mathcal{A}, \ldots)$:h and $\mathbf{T}_i^U$ is a final label (i.e., it is not the case that $\Gamma \mid\!\sim_{\mathcal{T}} \mathbf{T}_j^D(\mathcal{A}, \ldots)$:h and $\mathbf{T}_i^U \sqsubset \mathbf{T}_j^D$). Then $\mathbf{T}_i^U(\mathcal{A}, \ldots)$:h is a* warrant. *We will also say that h is a warranted literal, or that $\mathcal{A}$:h is a* warrant *in Γ.*

In LDS_{AR}, the construction of dialectical trees is formalized in terms of an inference relationship $\mid\!\sim_{\mathcal{T}}$. Figure 3 summarizes the rules needed for formalizing the above dialectical analysis. Rule Intro-1D allows to generate a tree with a single argument (*i.e.*, a generalized argument which is minimal). 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 (*i.e.* the root of every $\mathbf{T}_i^*$ is a defeater for the root of $\mathbf{T}^*$,

[4] It can be proven that Def. 13 and 12 are equivalent [Che01].

and no fallacious argumentation is present). Rules Mark-Atom, Mark-1D and Mark-ND allow to 'mark' the nodes (arguments) in a dialectical tree as defeated or undefeated. The tree is marked as an AND-OR tree. Nodes with no defeaters are marked as U-nodes (undefeated nodes). 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 D-node.

Example 4. Consider the argumentative theory from example 1 and the arguments and defeat relations from examples 2 and 3. From the argumentative theory Γ_{engine} the following formulas can be inferred via $\mathrel{|\!\sim_{\mathcal{T}}}$:

$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_1^*(\mathcal{A}){:}engine\_ok$	via Intro-1D	(1)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_2^*(\mathcal{B}){:}{\sim}fuel\_ok$	via Intro-1D	(2)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_3^*(\mathcal{C}){:}{\sim}low\_speed$	via Intro-1D	(3)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_4^*(\mathcal{D}){:}fuel\_ok$	via Intro-1D	(4)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_5^*(\mathcal{E}){:}{\sim}engine\_ok$	via Intro-1D	(5)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D})){:}{\sim}fuel\_ok$	via Intro-ND, (3) and (4)	(6)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_1^*(\mathcal{A}, \mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D})), \mathbf{T}_5^*(\mathcal{E})){:}engine\_ok$	via Intro-ND and (6)	(7)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_5^U(\mathcal{E}){:}{\sim}engine\_ok$	via Mark-Atom	(8)
$\Gamma \mathrel{|\!\sim_{\mathcal{T}}} \mathbf{T}_1^D(\mathcal{A}, \mathbf{T}_2^*(\mathcal{B}, \mathbf{T}_3^*(\mathcal{C}), \mathbf{T}_4^*(\mathcal{D})), \mathbf{T}_5^U(\mathcal{E})){:}engine\_ok$	via Mark-1D and (8)	(9)

Note that the formula obtained in step (7) has a final label associated with it, since it cannot be 'expanded' from previous formulas. Hence, following definition 13, we can conclude that *engine_ok* is not warranted.

3 LDS_{AR}: Some relevant logical properties

LDS_{AR} provides a useful formal framework for studying logical properties of argument-based systems in terms of *inference relationships*.[5] Three particular consequence operators can be identified:

- $Th_{sld}(\Gamma)$, $= \{\emptyset{:}h \mid \Gamma \mathrel{|\!\sim_{Arg}} \emptyset{:}h\}$, which denotes the set of non-defeasible conclusions that follow from Γ by using only strict rules.
- $C_{arg} = \{\mathcal{A}{:}\alpha \mid \Gamma \mathrel{|\!\sim_{Arg}} \mathcal{A}{:}\alpha$, where α is a literal in $\mathcal{L}_{\mathsf{KR}}$ $\}$, which denotes the set of all arguments that follow from Γ;
- $C_{war} = \{\emptyset{:}h \mid$ there exists a warranted argument $\mathcal{A}{:}h$ based on Γ $\}$, which denotes the set of all warranted conclusions that follow from Γ;

Cummulativity was proven to hold for argumentative formulae. This allows to think of an argumentative theory as a knowledge base containing 'atomic' arguments (facts and rules), which can be later on extended by incorporating new, more complex arguments. Cummulativity is proven *not* to hold for warranted conclusions, following the intuitions suggested by Prakken & Vreeswijk [PV99].

[5] See [Ant96] for an excellent survey on the role of inference relationships and their properties in nonmonotonic logics.

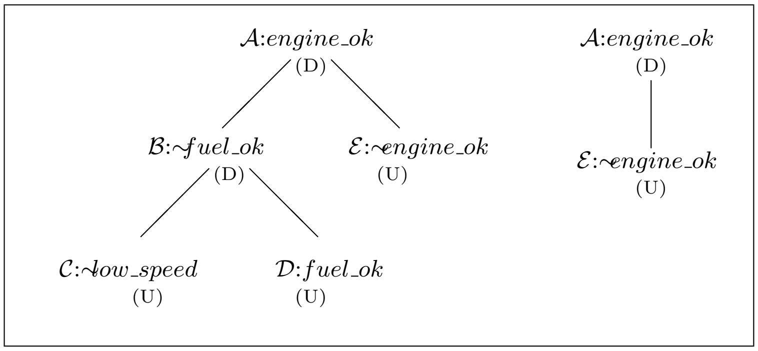

Fig. 4. Dialectical tree $\mathcal{T}_{\mathcal{A}:engine\_ok}$ and associated pruned tree $Pruned(\mathcal{T}_{\mathcal{A}:engine\_ok})$

Lemma 1 (Cummulativity for Arguments). [6] *Let Γ be an argumentative theory, and let α_1 and α_2 be wffs in $\mathcal{L}_{\mathsf{KR}}$. Then $\Gamma \mid\!\sim_{Arg} \mathcal{A}_1{:}\alpha_1$ implies that $\Gamma \cup \{\mathcal{A}_1{:}\alpha_1\} \mid\!\sim_{Arg} \mathcal{A}_2{:}\alpha_2$ iff $\Gamma \mid\!\sim_{Arg} \mathcal{A}_2{:}\alpha_2$*

A special variant of *superclassicality* was shown to hold for both argument construction and warrant wrt SLD resolution: if $Th_{sld}(\Gamma)$ denotes the set of conclusions that can be obtained from Γ via SLD, then it holds that $C_{arg}(\Gamma) \subseteq Th_{sld}(\Gamma)$ and $C_{war}(\Gamma) \subseteq Th_{sld}(\Gamma)$, where C_{arg} and C_{war} stand for the consequence operator for argument construction and warrant, respectively. This implies, among other things, that the analysis of attack between arguments can be focused on literals in defeasible rules. Formally:

Lemma 2 (Horn supraclassicality for C_{arg} and C_{war}). *Operators $C_{arg}(\Gamma)$ and C_{war} satisfy Horn supraclassicality wrt Th_{sld}, i.e. $Th_{sld}(\Gamma) \subseteq C_{arg}(\Gamma)$ and $Th_{sld}(\Gamma) \subseteq C_{war}(\Gamma)$.*

Analogously, a variant of right weakening is proven to hold for both C_{arg} and C_{war}. This implies that (warranted) arguments with a conclusion x account also as (warranted) arguments for y whenever $y \leftarrow x$ is present as a non-defeasible rule. A full analysis of the logical properties of LDS_{AR} is outside the scope of this paper; for an in-depth treatment the reader is referred to [CS01].

4 LDS_{AR}: theoretical considerations and applications

4.1 Computing Warrant: Bottom-up vs. Top-down

As described in Section 2.3, the notion of dialectical tree allows to capture the computation of warranted arguments. This notion is relevant in the context

[6] Proofs of propositions and theorems are not included for space reasons. For details the interested reader is referred to [Che01,CS01,CS02].

of defeasible argumentation in particular, and with respect to scientific reasoning in general. In most implementations of defeasible argumentation (e.g. DeLP [GS03]), computation of warrant is performed in a top-down fashion, based on a depth-first construction of a dialectical tree. As a marked dialectical tree is an AND-OR tree, an additional α-β pruning can be performed as the tree is built, resulting in a smaller tree, *pruned* tree.

Example 5. Consider the dialectical label rooted in $\mathcal{A}$:*engine_ok* associated with the final dialectical label in example 4. This label can be depicted as a *dialectical tree* as shown in figure 4 (left). The root node of $\mathcal{T}_{\mathcal{A}:engine\_ok}$ is labeled as D-node. Note that it is not necessary to compute the whole tree to mark the root node as D. In fact, considering the pruned tree $Pruned(\mathcal{T}_{\mathcal{A}:engine\_ok})$ shown in figure 4 (right), an equivalent answer would have been obtained. Note that $Pruned(\mathcal{T}_{\mathcal{A}:engine\_ok})$ was obtained from $\mathcal{T}_{\mathcal{A}:engine\_ok}$ by applying α-β pruning.

The LDS approach provides a bottom-up construction procedure, as complex labels are built on the basis of more simple ones. It can be proven that warrant can be computed by either of these approaches. In particular, such equivalence result shows that pruning aspects in the top-down approach (commonly used in implemented argument-based systems as [GS03]) correspond to performing a particular selection of inference rules in the bottom-up approach.

Theorem 1. *Given an argumentative theory Γ, the following three cases are equivalent: 1) The root of $\mathcal{T}_{\mathcal{A}:q}$ is marked as U-node; 2) The root of $Pruned(\mathcal{T}_{\mathcal{A}:q})$, is marked as U-node; 3) It is the case that $\Gamma \mid\!\sim_{T} \mathcal{A}{:}h^{U}$.*

4.2 Variants of LDS_{AR}

Another interesting issue concerns the definition of *variants* for LDS_{AR}. Since LDS_{AR} is a logical framework, its knowledge-encoding capabilities are determined by the underlying logical language, whereas the inference power is characterized by its deduction rules. Adopting a different knowledge representation language or modifying some particular inference rules would lead to different variants of LDS_{AR}, resulting in a *family* of argumentative systems. Figure 5 summarizes some of these variants of LDS_{AR} and their relationship to to some existing argumentation frameworks, such as Simari-Loui's [SL92], MTDR (an extension of the original Simari-Loui approach), Defeasible Logic Programming [GS03] and NLP (normal logic programming), conceptualized in an argumentative setting as suggested in [KT99]. Every variant of LDS_{AR} is denoted as AS_x (standing for Argumentative System). Thus, for instance, adopting a restricted first-order language as the knowledge representation language $\mathcal{L}_{\mathsf{KR}}$ leads to AS_{SL}, a particular instance of LDS_{AR} with a behavior similar to the argumentative framework proposed in [SL92]. Similarly, restricting the language $\mathcal{L}_{\mathsf{KR}}$ in LDS_{AR} to normal clauses [Llo87] and incorporating an additional inference rule to handle default negation will result in a particular argumentative system AS_{NLP}, a formulation similar to normal logic programming (NLP) under well-founded semantics as discussed in [KT99].[7] Two distinguished variants of LDS_{AR} deserved particular attention, as they allowed to model two particular cases of defeasible logic programming [GS03], namely $DeLP_{\mathsf{not}}$ and $DeLP_{neg}$ ($DeLP$ restricted to default and strict negation, resp.). Such special cases of DeLP could be better understood and compared in the context of extensions based on LDS_{AR}.

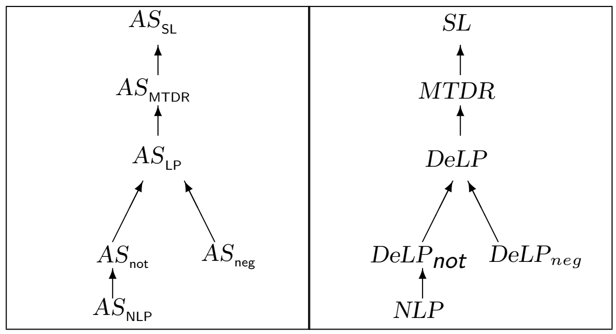

Fig. 5. A taxonomy relating the expressive power of LDS_{AR} and different argumentation systems

4.3 Extending LDS_{AR} to incorporate numerical attributes

The growing success of argumentation-based approaches has caused a rich crossbreeding with other disciplines, providing interesting results in different areas such as legal reasoning, medical diagnosis and decision support systems. Many of these approaches rely on *quantitative aspects* (such as numeric attributes, probabilities or certainty values). As argumentation provides mostly a non-numerical, *qualitative* setting for commonsense reasoning, integrating both quantitative and qualitative features has shown to be highly desirable.

LDS_{AR} can be naturally extended to incorporate such quantitative features, *e.g.* by adding some *certainty factor* cf such that $cf(f) = 1$ whenever f corresponds to non-defeasible knowledge, and $0 < cf(f) < 1$ whenever f stands for defeasible knowledge. A formula of the form $[\alpha, cf(\alpha)]{:}\alpha$ in the knowledge base Γ would therefore stand for "α is a defeasible formula which has the certainty factor $cf(\alpha)$".Similarly, the formula $[\emptyset, 1]{:}\alpha$ would stand for "α is a non-defeasible formula". Finally, performing an inference from Γ (*i.e.*, building a generalized argument) would result in inferring a formula $[\Phi, cf(\Phi)]{:}\alpha$, standing for "The set Φ provides an argument for α with a certainty factor $cf(\Phi)$".

In [CS02] this approach was first explored, and an extension of the LDS_{AR} framework was defined in order to incorporate numerical attributes. In this extended framework, deduction rules propagate certainty factors as inferences are

[7] A full discussion of different argumentative frameworks encompassed by LDS_{AR} can be found in [Che01].

carried out both in arguments and dialectical trees.[8] It must be remarked that the combination of qualitative and quantitative reasoning has recently motivated the development of general encompassing frameworks, such as the one proposed in [ADP03], which allows to deal with default, paraconsistency and uncertainty reasoning, and is general enough to capture Possibilistic Logic Programs and Fuzzy Logic Programming, among others.

5 Conclusions

As we have outlined in this paper, Labelled Deductive Systems offer a powerful tool for formalizing different aspects of defeasible argumentation. Many argument-based formalisms exist (e.g. [GS03,PS97,Vre93]), relying on a number of shared notions such as the definition of argument, defeat and warrant. Such formalisms provided the motivation for the definition of LDS_{AR}, in which the above notions could be abstracted away by specifying a suitable underlying logical language and appropriate inference rules.

LDS_{AR} provides a formal framework for argumentative reasoning which can be adapted for different purposes. As we have detailed in section 3, LDS_{AR} makes it easier to analyze, compare and relate alternative argumentative frameworks. Relevant logical properties of argumentation can also be studied and analyzed in a formal setting. Arguments in conflict can be compared and weighed wrt to qualitative features (*e.g.* specificity) or quantitative ones (*e.g.* certainty factors). Aggregated preference criteria can be defined to properly combine these such preference orderings. The same analysis applies to the construction of dialectical trees. Alternative approaches can extend the original labeling criterion, as in the case of considering *accrual of arguments* [Vre93,Ver96] when assessing a new certainty factor for the root of a dialectical tree.

In summary, we contend that a general encompassing framework as LDS_{AR} provides an integrated test-bed for studying different issues and open problems related to computational models of defeasible argumentation (such as argumentation protocols, models of negotiation, resource-bounded reasoning, etc.). Research in this direction is currently being pursued.

Acknowledgments: The authors want to thank one of the anonymous reviewers for his suggestions, which helped improve the final version of this paper.

References

[ADP03] Alcântara, J., Damásio, C., and Pereira, L. An encompassing framework for paraconsistent logic programs. *J. of Applied Logic (to appear)* (2003).

[Ant96] Antoniou, G. *Nonmonotonic Reasoning.* The MIT Press, 1996.

[8] A detailed analysis of this extension of LDS_{AR} is outside the scope of this paper. For details see [CS02].

[Che01] Chesñevar, C. I. *Formalizing Defeasible Argumentation As Labelled Deductive Systems*. PhD thesis, Departament of Computer Science, Universidad Nacional del Sur, Argentina, January 2001.

[CML00] Chesñevar, C. I., Maguitman, A., and Loui, R. Logical Models of Argument. *ACM Computing Surveys 32*, 4 (December 2000), 337–383.

[CS01] Chesñevar, C. I., and Simari, G. R. Consequence Operators for Defeasible Argumentation: Characterization and Logical Properties. In *Proc. of the VII Argentinean Congress in Computer Science, ISBN 987-96 288-6-1* (October 2001), VII CACIC, El Calafate, Argentina, pp. 309–320.

[CS02] Chesñevar, C. I., and Simari, G. R. Combining Quantitative and Qualitative Reasoning in Defeasible Argumentation. In *Proc. of the VIII Argentinean Congress in Computer Science* (October 2002), Univ. de Buenos Aires, Argentina, pp. 272–283.

[Gab96] Gabbay, D. *Labelling Deductive Systems (vol.1)*. Oxford University Press (Volume 33 of Oxford Logic Guides), 1996.

[Gin93] Ginsberg, M. *Essentials of Artificial Intelligence*. Morgan Kaufmann Publishers, Inc., 1993.

[GS03] Garcia, A., and Simari, G. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming (to appear)* (2003).

[KT99] Kakas, A., and Toni, F. Computing argumentation in logic programming. *Journal of Logic Programming 9*, 4 (1999), 515:562.

[Llo87] Lloyd, J. W. *Foundations of Logic Programming*, second edition ed. Springer-Verlag, New York, 1987.

[Pol95] Pollock, J. L. *Cognitive Carpentry: A Blueprint for How to Build a Person*. Massachusetts Institute of Technology, 1995.

[PS97] Prakken, H., and Sartor, G. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics 7* (1997), 25–75.

[PV99] Prakken, H., and Vreeswijk, G. Logics for Defeasible Argumentation. In *Handbook of Philosophical Logic*, D. Gabbay, Ed. Kluwer Academic Publisher, 1999.

[SCG94] Simari, G. R., Chesñevar, C. I., and García, A. J. The role of dialectics in defeasible argumentation. In *Anales de la XIV Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación* (November 1994), Universidad de Concepción, Concepción (Chile).

[SCG01] Simari, G. R., Chesñevar, C. I., and García, A. Modelling Argumentation in a Logic-Programming Setting: Formalization and Logical Properties. In *Proc. of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertaintiy – Workshop "Adventures in Argumentation"* (September 2001), Toulouse, France.

[SGCS03] Stolzenburg, F., García, A., Chesñevar, C. I., and Simari, G. Computing Generalized Specificity. *Journal of Non-Applied Classical Logics 13*, 1 (2003), 87–113.

[SL92] Simari, G. R., and Loui, R. P. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence 53* (1992), 125–157.

[Ver96] Verheij, B. *Rules, Reasons, Arguments: formal studies of argumentation and defeat*. PhD thesis, Maastricht University, Holland, December 1996.

[Vre93] Vreeswijk, G. A. *Studies in Defeasible Argumentation*. PhD thesis, Vrije University, Amsterdam (Holanda), 1993.