Combining Quantitative and Qualitative Reasoning in Defeasible Argumentation

Carlos Iván Chesñevar **Guillermo Ricardo Simari**

Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
Av.Alem 1253 – B8000CPB Bahía Blanca – República Argentina
Tel/Fax: (+54) (291) 459 5135/5136 – Email: {cic,grs}@cs.uns.edu.ar
Key words: Defeasible Argumentation, Quantitative Reasoning, Labelled Deductive Systems

Abstract

Labeled Deductive Systems (LDS) were developed as a rigorous but flexible methodology to formalize complex logical systems, such as temporal logics, database query languages and defeasible reasoning systems.

LDS_{AR} is a LDS-based framework for *defeasible argumentation* which subsumes different existing argumentation frameworks, providing a testbed for the study of different relevant features (such as logical properties and ontological aspects, among others).

This paper presents LDS^{*}_{AR}, an extension of LDS_{AR} that incorporates the ability to combine quantitative and qualitative features within a unified argumentative setting. Our approach involves the assignment of *certainty factors* to formulas in the knowledge base. These values are propagated when performing argumentative inference, offering an alternative source of information for evaluating the strength of arguments in the dialectical analysis. We will also discuss some emerging logical properties of the resulting 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 the wff *f*. A labeling language $\mathcal{L}_{Label}$ and knowledge-representation language $\mathcal{L}_{kr}$ can be combined to provide a new, labeled language, in which labels convey additional information also encoded at object-language level. Formulas are labeled according to a family of *deduction rules*, and with agreed ways of propagating labels via the application of these rules.

The study of logical properties of *defeasible argumentation* motivated the development of LDS_{AR} [Che01, SCG01], an LDS-based argumentation formalism. LDS_{AR} provides a useful formal framework for studying logical properties of defeasible argumentation in general, and of DeLP [Gar00] in particular. Equivalence results with other argumentative frameworks were also studied.

Labeled deduction has a number of features which make it suitable for characterizing new ontologies. As discussed in [SCG01], different *variants* of defeasible argumentation can be explored from the original LDS_{AR} formulation by introducing modifications in the object language. Such changes can be introduced in a modular way, without affecting the framework as a whole.

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 [TP01]. Remarkably, numerical reasoning has been long neglected in the defeasible argumentation community. This is maybe due to the historical origins of the discipline, which were more related to legal (qualitative) reasoning rather than to number-based attributes as those used in rule-based production systems.

This paper extends the approach first presented in [CS02] to characterize defeasible argumentation with numerical values. Our motivation is presenting a formal definition of LDS^{*}_{AR}, an framework for defeasible argumentation based on labeled deduction that incorporates the ability to perform argumentative reasoning with numerical values. As it stands, LDS_{AR} provides a sound setting for qualitative reasoning. Incorporating numerical reasoning capabilities would offer an additional and potentially useful source of information in several knowledge domains. Combining both kinds of reasoning into an single argumentation framework turns out to be particularly attractive when considering many real-world applications.

This paper is structured as follows. First, in section 2 we discuss the main features of the LDS_{AR} framework. Then in section 3 we introduce some basic ideas of numerical reasoning and the different possibilities for integrating it with deductive systems. Section 4 introduces the extended framework. The labeling language will be modified in order to incorporate *certainty factors*. New deduction rules for inferring arguments and performing the associated dialectical analysis will be defined. In section 5 we discuss some basic logical properties that hold in the proposed framework. Finally, section 6 concludes.

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

2.1 Knowledge representation in LDS_{AR}

In this section we will first introduce a *knowledge representation language* $\mathcal{L}_{\mathsf{KR}}$ for performing defeasible inference, together with a *labeling language* $\mathcal{L}_{\mathsf{Labels}}$. These languages will be used to define the object language $\mathcal{L}_{\mathsf{Arg}}$. Following Gabbay's terminology [Gab96], the basic information units in $\mathcal{L}_{\mathsf{Arg}}$ will be called *declarative units*, having the form *Label:wff*. In our approach we will restrict wffs in labeled formulas to ground literals. A ground literal h can be understood as a *conclusion* of an *argument*, which will be defined by the label.

A label in a formula $L{:}\alpha$ will provide different elements which are convenient to take into account when formalizing defeasible argumentation. Given a knowledge base Γ and a declarative unit $L{:}\alpha \in \Gamma$, the label L will distinguish whether $L{:}\alpha$ corresponds to *defeasible*

[1] For space reasons we only give a brief summary of the main elements of the LDS_{AR} framework. In order to make the paper self-contained, some formal issues (such as Def. 2.2) were consequently simplified. We assume that the reader has basic knowledge about the basic concepts underlying most defeasible argumentation formalisms (for an in-depth treatment see [Che01, PV99]).

or *non-defeasible* information. When performing the inference of $L{:}\alpha$ from a set Γ of declarative units, the label L will also provide a *trace* of the wffs needed to infer $L{:}\alpha$ from Γ.

Our knowledge representation language $\mathcal{L}_{\mathsf{KR}}$ is a Horn-like propositional language restricted to *rules* and *facts*. The set of all rules and facts in $\mathcal{L}_{\mathsf{KR}}$ will be denoted $\mathsf{Rules}(\mathcal{L}_{\mathsf{KR}})$ and $\mathsf{Facts}(\mathcal{L}_{\mathsf{KR}})$, resp. We define $\mathsf{ProgClauses}(\mathcal{L}_{\mathsf{KR}}) = \mathsf{Rules}(\mathcal{L}_{\mathsf{KR}}) \cup \mathsf{Facts}(\mathcal{L}_{\mathsf{KR}})$. A modality (label) will be attached to wffs in $\mathcal{L}_{\mathsf{KR}}$, indicating whether they are *defeasible* or *non-defeasible*.

Definition 2.1 (Language $\mathcal{L}_{\mathsf{KR}}$. Wffs in $\mathcal{L}_{\mathsf{KR}}$) *The language $\mathcal{L}_{\mathsf{KR}}$ will be composed of*

1. *A countable set of* propositional atoms*, possibly subindicated. Example: a, b, c, d, e, $\ldots$, a_1, a_2, a_3 are propositional atoms.*

2. *Logical connectives $\wedge$, $\sim$ and $\leftarrow$.*

Wffs in $\mathcal{L}_{\mathsf{KR}}$ will be defined as follows:

1. *If α is an atom in $\mathcal{L}_{\mathsf{KR}}$, then α and $\sim\alpha$ are wffs called* literals *in $\mathcal{L}_{\mathsf{KR}}$.*

2. *If $\alpha_1, \ldots \alpha_k$, β are literals in $\mathcal{L}_{\mathsf{KR}}$, then $\beta \leftarrow \alpha_1, \ldots \alpha_k$ is a wff in $\mathcal{L}_{\mathsf{KR}}$.*

For the sake of simplicity, when referring to the language $\mathcal{L}_{\mathsf{KR}}$ the following conventions will be used: Greek lowercase letters α, β, γ will refer to any wff in $\mathcal{L}_{\mathsf{KR}}$. Lowercase letters (*e.g.* h, q, etc.) will be used for referring to ground literals in $\mathcal{L}_{\mathsf{KR}}$. Greek uppercase letters (*e.g.* Γ, Φ, etc.) will refer to sets of wffs in $\mathcal{L}_{\mathsf{KR}}$. The conjunction $\alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_k$ will be simply written as $\alpha_1, \alpha_2, \ldots, \alpha_k$.

Definition 2.2 (Labeling language $\mathcal{L}_{\mathsf{Labels}}$) *The labeling language $\mathcal{L}_{\mathsf{Labels}}$ is a set of labels $\{L_1, L_2, \ldots L_k\}$, such that every label $L \in \mathcal{L}_{\mathsf{Labels}}$ can be either an* argument label *or a* dialectical label*, defined as follows:*[2]

1. *An* argument label *will be a set $\Phi \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$.*

2. *If Φ is an argument label, then $\mathbf{T}^U_j(\Phi)$, with $j \in \mathsf{Nat}$ and $\mathbf{T}^D_k(\Phi)$, with $k \in \mathsf{Nat}$ are* dialectical labels *in $\mathcal{L}_{\mathsf{Labels}}$. For the sake of simplicity, we will write $\mathbf{T}^D_k$ to denote a generic dialectical label $\mathbf{T}^D_k(\Phi)$ for a given argument label Φ. We will also write $\mathbf{T}_k$ to denote either the functor $\mathbf{T}^D_k$ or the functor $\mathbf{T}^U_k$.*

3. *If $\mathbf{T}_1, \ldots, \mathbf{T}_k$ are dialectical labels, then $\mathbf{T}^U_n(\mathbf{T}_1, \ldots, \mathbf{T}_k)$, with $k \in \mathsf{Nat}$, $n \notin \{1 \ldots k\}$, and $\mathbf{T}^D_m(\mathbf{T}_1, \ldots, \mathbf{T}_k)$, with $k \in \mathsf{Nat}$, $m \notin \{1 \ldots k\}$ will also be dialectical labels in $\mathcal{L}_{\mathsf{Labels}}$.*

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

If $\mathcal{L}_{\mathsf{Labels}}$ is a labeling language, and $\mathcal{L}_{\mathsf{KR}}$ is a knowledge representation language, then 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 $P \subseteq \mathsf{ProgClauses}(\mathcal{L}_{\mathsf{KR}})$, we will write $P \vdash_{SLD} \alpha$ to denote that α follows from P via $\vdash_{SLD}$.

[2]For space reasons, we do not give a fully formal definition of the labeling language, which involves defining the associated alphabet, constant names, etc. See details in [Che01].

Definition 2.3 (Contradictory set of wffs in $\mathcal{L}_{\mathsf{KR}}$) *Given a set P of wffs in $\mathcal{L}_{\mathsf{KR}}$, P will be called a* contradictory set *(denoted $P \vdash_{SLD} \bot$) iff literals p and $\overline{p}$ can be derived via $\vdash_{SLD}$ from P.*

Some *basic* declarative units will be used to encode defeasible and non-defeasible information available for an intelligent agent to reason from his knowledge base Γ. Formally:

Definition 2.4 (Basic declarative units) *Let $\gamma = \phi{:}\alpha$ be a formula in $\mathsf{Wffs}(\mathcal{L}_{\mathsf{Arg}})$. Then γ will be called a* basic declarative unit *if $\alpha \in \mathcal{L}_{\mathsf{KR}}$. We will distinguish two kinds of basic declarative units:* defeasible formulas*, having the form $\phi{:}\phi$, and* non-defeasible formulas*, having the form $\emptyset{:}\phi$.*

Given a formula $\phi{:}\alpha$, the label ϕ is intended to provide the *set of support* needed for using α when performing inferences using the inference relationship $\mid\!\sim_{Arg}$, as we will see in section 2.2. Some distinguished sets associated with a knowledge base Γ will also be considered: $\mathsf{Strict}(\Gamma)$ is the set of all non-defeasible formulas in Γ; $\mathsf{Defeasible}(\Gamma) = \Gamma - \mathsf{Strict}(\Gamma)$; $\mathbf{\Pi}(\Gamma)$ is the set of all $\mathcal{L}_{\mathsf{KR}}$ formulas in Γ whose support set is empty; $\mathbf{\Delta}(\Gamma)$ is the set of all $\mathcal{L}_{\mathsf{KR}}$ formulas in Γ whose support set is non-empty.

Definition 2.5 (Argumentative theory Γ) *A finite set $\Gamma = \{ \gamma_1, \gamma_2, \ldots, \gamma_k\}$ of basic declarative units will be called an* argumentative theory*. We will assume that the set of non-defeasible information $\mathbf{\Pi}(\Gamma) \subseteq \mathsf{Wffs}(\mathcal{L}_{\mathsf{KR}})$ in any argumentative theory Γ is non-contradictory.*

2.2 Argument construction and warrant in $\boldsymbol{LDS}_{AR}$

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 introduced in a proof. This information is collected into a *set of support*, containing all defeasible information needed to conclude a given formula. Thus, arguments are modeled as labeled formulas $\mathcal{A}{:}h$, where $\mathcal{A}$ stands for a set of (ground) clauses, and h is an extended literal in $\mathcal{L}_{\mathsf{KR}}$. Figure 1 summarizes the natural deduction rules which characterize the inference relationship $\mid\!\sim_{Arg}$.

Definition 2.6 (Generalized argument. Argument) *Let Γ be an argumentative theory, and let $h \in \mathsf{Lit}(\mathcal{L}_{\mathsf{KR}})$ 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*.*

Given an argument $\mathcal{A}{:}h$ derivable from a theory Γ, there may be other conflicting arguments also supported by Γ which *defeat* it according to some preference criterion. Such conflicting arguments are called *defeaters*. A common syntactic preference criterion among arguments is *specificity* [SL92], which prefers those arguments which are more informed or more 'direct'. However, any partial order on the set of all possible arguments could be used. Since defeaters are arguments, they may be on its turn defeated, and so on. This leads to a recursive analysis, in which a tree structure rooted in $\mathcal{A}{:}h$ results. If $\mathcal{A}{:}h$ ultimately

1. $\dfrac{}{\emptyset{:}\alpha}$ for any $\emptyset{:}\alpha \in \mathsf{Strict}(\Gamma)$.

2. $\dfrac{\mathbf{\Pi}(\Gamma) \cup \Phi \not\vdash_{SLD} \bot}{\Phi{:}\alpha}$ for any $\Phi{:}\alpha \in \mathsf{Defeasible}(\Gamma)$.

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

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

Figure 1: Inference rules for $|\!\sim_{Arg}$: deriving generalized arguments in LDS_{AR}

prevails over its conflicting defeaters, then $\mathcal{A}{:}h$ is called a *warrant*. In LDS_{AR} this situation is formalized in terms of an inference relationship $|\!\sim_{\mathcal{T}}$.[3]

Definition 2.7 (Warrant –sketch) *Let* Γ *be an argumentative theory, such that* $\Gamma |\!\sim_{Arg} \mathcal{A}{:}h$*, and* $\mathcal{A}{:}h$ *is an argument such that:*

1. *it has no defeaters; or*

2. *every defeater for* $\mathcal{A}{:}h$ *is ultimately defeated.*

Then $\mathcal{A}{:}h$ *is a warranted argument. In that case, we will also say that* h *is* warranted*.*

3 Handling uncertainty in commonsense reasoning

As Judea Pearl points out in [Pea88], commonsense reasoning involves *summarizing exceptions* at a given stage. In defeasible argumentation this is done by providing defeasible rules "$p(X) \prec q(X)$", which provide a symbolic way of specifying "not every $q(X)$ is $p(X)$". Another way of summarizing exceptions is to assign to each proposition a numerical measure of uncertainty, and then combine these measures according to uniform syntactic principles.

When introducing numerical values for modeling uncertainty, *extensional* and *intensional* approaches can be distinguished. Extensional approaches treat uncertainty as a generalized truth value attached to formulas. Computing the uncertainty of any formula is a function of the uncertainties of its subformulas. Intensional approaches, on the other hand, are model-based: uncertainty is attached to "states of affairs" or subsets of "possible worlds". Typical examples of this extensional approaches are production systems and rule-based systems.

Extensional approaches are computationally attractive, but their semantics may be sometimes 'sloppy'. Intensional approaches are semantically clear but computationally clumsy. Most research has been directed to find a trade-off between these two ways of formalizing uncertainty. In order to incorporate numerical attributes in our formalism, we will adopt an extensional approach as it can be easily integrated in the existing ontology, as we will see

[3] For space reasons we just give a brief sketch of the notion of warrant. We will discuss this notion in more detail in section 2.2.

in the next section. Although semantical issues are not discussed in this paper, it must be noted that LDS provide a sound generic basis for defining *fibered semantics* associated with arbitrary logical systems using labeled deduction [Gab96].

4 LDS^*_{AR}: extending LDS_{AR} with numerical attributes

In order to introduce an uncertainty measure in LDS_{AR}, we will extend labels by adding a *certainty factor cf* to every wff in the object language $\mathcal{L}_{\text{Arg}}$. This approach resembles the one used in MYCIN [Sho76], which has been extensively used in many commercial expert systems. As a result, we will obtain LDS^*_{AR}, an extended version of LDS_{AR} which incorporates *cf*'s along with the traditional inference process described in Section 2.2.

A certainty factor v is a numerical value in the range $[0, 1]$. Given a wff f, we will consider $cf(f) = 1$ whenever f corresponds to *non-defeasible* knowledge, and $0 \leq cf(f) < 1$ whenever f stands for *defeasible* knowledge. Thus, given a piece of defeasible knowledge p, there is a range of possible certainty values $0 \leq cf(p) < 1$. This is intended to represent different 'degrees of acceptability' for that wff p.

Definition 4.1 (Labeling language $\mathcal{L}^*_{\text{Labels}}$ in LDS^*_{AR}) *The labeling language $\mathcal{L}^*_{\text{Labels}}$ is a set of labels $\{L_1, L_2, \ldots, L_k\}$. Every $L_i \in \mathcal{L}^*_{\text{Labels}}$ is a tuple $[v, \Phi]$, where v is a certainty factor and $\Phi \in \mathcal{L}_{\text{Labels}}$ (as specified in def. 2.2).*

Definition 4.2 (Object language $\mathcal{L}^*_{\text{Arg}}$ in LDS^*_{AR}) *Given the labeling language $\mathcal{L}^*_{\text{Labels}}$, the* object (labeled) language *in LDS^*_{AR} is defined as $\mathcal{L}^*_{\text{Arg}} = (\mathcal{L}^*_{\text{Labels}}, \mathcal{L}_{\text{KR}})$.*

It should be noted that we will maintain the knowledge representation language $\mathcal{L}_{\text{KR}}$ as well as the rest of the concepts and definitions (such as the notion of argument, knowledge representation language, warrant, etc.) introduced in section 2. In this new setting, basic declarative units of the form $[\alpha, cf(\alpha)]{:}\alpha$ such that $0 \leq cf(\alpha) < 1$ would stand for *"α is a defeasible formula which has the certainty factor $cf(\alpha)$"*.[4] Similarly, the formula $[\emptyset, 1]{:}\alpha$ stands for *"α is a non-defeasible formula"*.

4.1 Building arguments in LDS^*_{AR}

In LDS^*_{AR} every basic declarative unit γ in a knowledge base Γ is attached with a certainty factor, indicating whether the formula corresponds to non-defeasible or defeasible knowledge. Performing an inference from Γ (*i.e.*, building a generalized argument for a given literal h) should consequently result in inferring a formula $[\Phi, cf(\Phi)]{:}\alpha$, standing for *"The set Φ provides an argument for α with a certainty factor $cf(\Phi)$"*. Natural deduction rules should propagate certainty factors as inference steps are carried out. Next we summarize some considerations related to argument construction in this new setting.

- **Propagating certainty factors:** In extensional systems as MYCIN, uncertainty is usually treated as a generalized truth value, *i.e.* the certainty of a formula is defined as a unique function from the certainties of its subformulas. Labeled Deductive Systems

[4] Note that in this setting it would not be strictly necessary to distinguish between defeasible and non-defeasible formulas as this qualification can be inferred from their associated certainty factors.

1. Intro-N:

$$\frac{}{[\emptyset, 1]{:}\alpha} \text{ for any } [\emptyset, 1]{:}\alpha \in \mathsf{Strict}(\Gamma).$$

2. Intro-D:

$$\frac{\mathbf{\Pi}(\Gamma) \cup \Phi \not\vdash_{SLD} \bot}{[\Phi, cf(\Phi)]{:}\alpha} \text{ for any } [\Phi, cf(\Phi)]{:}\alpha \in \mathsf{Defeasible}(\Gamma).$$

3. Intro-$\wedge$:

$$\frac{[\Phi_1, cf(\Phi_1)]{:}\alpha_1 \quad [\Phi_2, cf(\Phi_2)]{:}\alpha_2 \quad \ldots \quad [\Phi_k, cf(\Phi_k)]{:}\alpha_k \qquad \mathbf{\Pi}(\Gamma) \cup \bigcup_{i=1}^{k} \Phi_i \not\vdash_{SLD} \bot}{[\bigcup_{i=1}^{k} \Phi_i, f_\wedge(cf(\Phi_1), cf(\Phi_2), \ldots, cf(\Phi_k))]{:}\alpha_1, \alpha_2, \ldots, \alpha_k}$$

4. MP:

$$\frac{[\Phi_1, cf(\Phi_1)]{:}\beta \leftarrow \alpha_1, \ldots, \alpha_k \qquad [\Phi_2, cf(\Phi_2)]{:}\alpha_1, \ldots, \alpha_k \qquad \mathbf{\Pi}(\Gamma) \cup \Phi_1 \cup \Phi_2 \not\vdash_{SLD} \bot}{[\Phi_1 \cup \Phi_2, f_{mp}(cf(\Phi_1), cf(\Phi_2)){:}\beta}$$

Figure 2: Rules for deriving generalized arguments in LDS^*_{AR}

allow us to proceed the same way: when performing an inference, a new label is defined in terms of existing (already inferred) labels. Therefore propagating cf's turns out to be natural in our framework. As an example, consider two formulas $[\Phi, cf(\Phi)]{:}\alpha$ and $[\Psi, cf(\Psi)]{:}\beta$. If α, β could be derived (introducing conjunction), the resulting formula would have the form $[\Phi \cup \Psi, f_\wedge(cf(\Phi), cf(\Psi))]{:}\alpha, \beta$, where $f_\wedge$ is a function which computes the cf of $\alpha \wedge \beta$. Similarly, a function f_{mp} can be defined to compute the cf of β after applying modus ponens from the formulas α and $\beta \leftarrow \alpha$.[5] Rules Intro-$\wedge$ and MP in figure 2 illustrate this situation.

- **Handling consistency:** In the original LDS_{AR} formulation, consistency checking of a wff f wrt a set of arbitrary wffs Φ stands for $\Phi \cup \{f\} \not\vdash_{SLD} \bot$. Note that consistency checking in defeasible argumentation involves the set of strict knowledge $\mathsf{Strict}(\Gamma)$, where all wffs are non-defeasible. Using the approach discussed above, when a new defeasible formula f is inferred, it will have the form $[\Phi, cf(\Phi)]{:}f$. Consistency checking can be handled by assuming that f is "locally non-defeasible" (*i.e.*, $[\emptyset, cf(1)]{:}f$) and non-contradictory wrt $\mathsf{Strict}(\Gamma)$ when building an argument.

Figure 2 summarizes the rules for (generalized) argument construction in LDS^*_{AR}, characterizing formulas of the form $[\Phi, cf(\Phi)]{:}f$ (standing for "Φ provides a tentative proof for f with certainty factor $cf(\Phi)$") which are non-contradictory wrt the strict knowledge $\mathsf{Strict}(\Gamma)$ for a given knowledge base Γ. Rules Intro-N and Intro-D introduce non-defeasible and defeasible information, respectively. Rules Intro-$\wedge$ and MP account for introducing conjunction and *modus ponens*. As discussed in previous sections, these natural deduction rules propagate labels when performing inference.

[5]In the case of MYCIN, $f_\wedge(\alpha_1, \alpha_2, \ldots, \alpha_k) = \mathsf{min}(\alpha_1, \ldots, \alpha_k)$, and $f_{mp}(\alpha, \beta \leftarrow \alpha) = cf(\beta \leftarrow \alpha) * cf(\alpha)/100$.

4.2 Conflict among arguments in $\boldsymbol{LDS}^*_{AR}$

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. 2.3). Defeat among arguments involves a partial order which establishes a preference criterion on them (*e.g.* specificity [SL92]).

Definition 4.3 (Counterargument) *Let Γ be an argumentative theory, and let $\mathcal{A}$:h and $\mathcal{B}$:q be arguments in 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*.*

Definition 4.4 (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 4.5 (Defeater) *Let Γ be an argumentative theory, such that $\Gamma|\!\sim_{Arg} \mathcal{A}$:$h$ and $\Gamma|\!\sim_{Arg} \mathcal{B}$:$q$. 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. *(a) It holds that $\mathcal{B}'$:$q' \preceq \mathcal{A}$:h, and $\mathcal{A}$:$h \not\preceq \mathcal{B}'$:$q'$.*
   (b) Either $\mathcal{A}$:$h \preceq \mathcal{B}'$:q' and $\mathcal{B}'$:$q' \preceq \mathcal{A}$:h, or both arguments cannot be compared.

In case 2a, we will say that $\mathcal{A}$:h is a proper *defeater for $\mathcal{B}$:q. In case 2b, we will say that $\mathcal{A}$:h is a* blocking *defeater for $\mathcal{B}$:q.*

Note that the counterargument relationship (def. 4.3) is defined in terms of *contradiction*. Therefore certainty factors should play no role in determining whether an argument counterargues another. In LDS_{AR}, the usual criterion for defeat among arguments is *specificity* [SL92]. However, in LDS^*_{AR} defeat can rely on the numerical weight of the arguments in conflict. For example, an argument $\mathcal{A}$:h could be deemed as a proper defeater $\mathcal{B}$:q if $cf(\mathcal{A}{:}h) > cf(\mathcal{B}{:}q)$. Similarly, a blocking defeat situation would arise if $cf(\mathcal{A}{:}h) = cf(\mathcal{B}{:}q)$, or alternatively $\mid cf(\mathcal{A}{:}h) - cf(\mathcal{B}{:}q) \mid \leq \epsilon$, for ϵ arbitrarily small. It is interesting to note that an *aggregated preference criterion* $\preceq^* = \{\preceq_{spec}, \preceq_{num}\}$ can be defined, in which a structural (*e.g.* specificity-based) preference criterion $\preceq_{spec}$ is first used. Should $\preceq_{spec}$ lead to a blocking situation (as in case 2b), then a numeric criterion $\preceq_{num}$ based on certainty factors is applied.

4.3 Dialectical analysis in $\boldsymbol{LDS}^*_{AR}$

As detailed in section 2.2, the whole process of determining whether a given argument is *warranted* (*i.e.* ultimately accepted) or not relies on defeat relationships between arguments. Given an argument $\mathcal{A}$:h, such relationships allow to build a *dialectical tree* rooted in $\mathcal{A}$:h, denoted $\mathcal{T}_{\mathcal{A}:h}$. Conflicting arguments correspond to nodes in the tree

- If $\mathcal{A}$:h. is an argument with no defeaters, then the dialectical $\mathcal{T}_{\mathcal{A}:h}$ has a single root node $\mathcal{A}$:h.

- If $\mathcal{A}$:h. is an argument with defeaters $\mathcal{B}_1$:h_1, ..., $\mathcal{B}_k$:h_k, then the dialectical $\mathcal{T}_{\mathcal{A}:h}$ is rooted in $\mathcal{A}$:h and has as immediate subtrees the dialectical trees for $\mathcal{B}_1$:h_1, ..., $\mathcal{B}_k$:h_k, *i.e.* $\mathcal{T}_{\mathcal{B}_1:h_1}$, ..., $\mathcal{T}_{\mathcal{B}_k:h_k}$

Given a dialectical tree $\mathcal{T}_{\mathcal{A}:h}$, a labeling process can be carried out. Leaves in $\mathcal{T}_{\mathcal{A}:h}$ are labeled as *undefeated* nodes. An inner node N in $\mathcal{T}_{\mathcal{A}:h}$ is labeled as *undefeated* iff every children of N in $\mathcal{T}$ is labeled as *defeated*; otherwise N is labeled as *defeated*. This situation is captured by the definition of *warrant* (see def. 2.7) and formalized in terms of the inference rules shown in figure 3. Rule Intro-1D specifies that any minimal argument $\mathcal{A}$:h constitutes an atomic *dialectical formula*. Rule Intro-ND indicates how to build a new dialectical formula $\mathbf{T}$ by introducing $\mathbf{T}_1^*$, ..., $\mathbf{T}_k^*$ as immediate sublabels such that they do not violate the VSTree precondition for *non-fallacious* argumentation.[6] Finally, rules Mark-Atom, Mark-1D and Mark-ND allow to perform the marking procedure on any dialectical tree. Note that these rules have attached another certainty factor whose role is discussed in the next section.

4.4 Introducing certainty factors in dialectical trees

The introduction of certainty factors does not affect the process of building a dialectical tree $\mathcal{T}$. However, cf's could be incorporated along the *labeling* of the tree, so that the certainty factor associated with a given argument $\mathcal{A}$:h is strengthened / weakened depending on the dialectical analysis rooted in $\mathcal{A}$:h.

- Given an argument $\mathcal{A}$:h which is a leaf node L in T, then $cf(L)$=$cf(\mathcal{A})$.
- Given an argument $\mathcal{B}$:q which is an inner node in T, then

$$cf(\mathcal{B}{:}q){=}f_{tree}(cf(\mathcal{B}), cf(T_1), ...cf(T_k))$$

  where $T_1...T_k$ are immediate subtrees of $\mathcal{B}$:q.

In other words, certainty factors can be propagated bottom-up according to some function f_{tree}. Computing $f_{tree}(cf(\mathcal{B}), cf(T_1), ...cf(T_k))$ can be thus defined in several ways. Such a setting allows to model a number of typical problems in defeasible argumentation, such as the the notion of *accrual of arguments* [Vre93, Ver96], where arguments with many (ultimately accepted) defeaters would be deemed weaker as those which have only one (ultimately accepted) defeater.

4.5 The resulting framework

From the discussion of the preceding sections we can now give a formal characterization of an extended logical framework for argumentation as follows:

Definition 4.6 (The LDS^*_{AR} framework) *Let Γ be a knowledge base in $\mathcal{L}^*_{\mathsf{Arg}}$. Then the LDS^*_{AR} framework is characterized by the consequence relationship $\mathrel{|\!\!\sim}_{Arg}{}^*$ =*{Intro-N, Intro-D, Intro-$\wedge$, MP}* for argument construction and the consequence relationship $\mathrel{|\!\!\sim}_{\mathcal{T}}{}^*$ = *{ Intro-1D, Intro-ND, Mark-Atom, Mark-1D, Mark-ND}.

[6] Fallacies are outside the scope of this paper. For details see [Che01, SCG01].

1. Intro-1D:

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

2. Intro-ND:

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

3. Mark-Atom:

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

4. Mark-1D:

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

for some $\mathbf{T}_i^*$, $i = 1 \ldots k$

5. Mark-ND:

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

$\forall\ \mathbf{T}_i^*$, $i = 1 \ldots k$

Figure 3: Rules for building dialectical trees in LDS^*_{AR}

5 Some logical properties of LDS^*_{AR}

Next we will discuss some logical properties of the proposed framework. First we will show that empty arguments are as reliable as proofs which do not involve any defeasible information.[7]

Proposition 5.1 *Let $\mathcal{A}$:h be an argument derived from a knowledge base Γ, such that $\mathcal{A}=\emptyset$. Then $cf(\mathcal{A}) = 1$.*

In [CS01] an exhaustive analysis of logical properties for the LDS_{AR} framework was given. Such properties also hold in LDS^*_{AR}, as certainty factors attached to formulas do not restrict the behavior of the original inference mechanism in LDS_{AR}. Formally:

[7] For space reasons we do not include the full proofs of these propositions, limiting ourselves to discuss the basic intuition underlying such proofs.

Proposition 5.2 (Logical Properties in $\boldsymbol{LDS^*_{AR}}$) *Consider LDS^*_{AR} as characterized in def. 4.6. Then the logical properties of inclusion (restricted to non-defeasible information), cummulativity, Horn superclassicality, Horn Right Weakening and subclassical cummulativity hold for* $\mid\!\sim_{Arg}{}^*$ *and* $\mid\!\sim_{\mathcal{T}}{}^*$.

Let us briefly discuss the role of the these properties. Restricted inclusion ensures that non-defeasible facts can be ontologically understood as empty arguments. Cummulativity allows to keep any argument obtained from a theory Γ as an 'intermediate proof' (lemma) to be used in building more complex arguments. Horn supraclassicality indicates that every conclusion that follows via SLD can be considered as a special form of argument (namely, an empty argument), whereas Horn right weakening ensures that strong rules preserve the intuitive semantics of a Horn rule (a strong rule $y \leftarrow x$ makes every argument $\mathcal{A}$ for x also an argument for y) Finally, subclassical cummulativity indicates that two theories Γ and Γ' whose information is a subset of those literals that can be derived via SLD from Γ (or Γ') are equivalent when considering the arguments that can be obtained from them.

Computing warrant, on the other hand, can also be better understood in the light of some logical properties of $\mid\!\sim_{\mathcal{T}}{}^*$. Restricted inclusion ensures that any non-defeasible fact in a theory Γ can be considered as warranted. Idempotence indicates that successive applications of $\mid\!\sim_{\mathcal{T}}{}^*$. on a the set S of warranted literals returns exactly the same set. From Horn supraclassicality it follows that every conclusion obtained via SLD is a particular case of warranted literal, whereas Horn right weakening indicates that non-defeasible rules behave as such in the meta-level (a strong rule $y \leftarrow x$ ensures that every warrant $\mathcal{A}$ for a literal x is also a warrant for y). From subclassical cummulativity it follows that two theories Γ and Γ', whose information is a subset of the conclusions that can be obtained from Γ (or Γ') are equivalent when considering the set of literals that can be warranted from them.

6 Conclusions and future work

In this paper we have outlined LDS^*_{AR}, an extension of the existing LDS_{AR} framework to incorporate numerical attributes. As we have shown, labels provide a flexible tool for including numeric information which can be propagated using suitable deduction rules. As discussed in section 4.1 and 4.3, introducing such numeric information can be performed in a modular way by expanding existing labels.

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 orderings. The same analysis applies to the construction of dialectical trees. In LDS_{AR} the labeling of the tree was performed by considering an AND-OR tree [SCG01]. As discussed in section 4.4, LDS^*_{AR} can incorporate alternative approaches which extend the original labeling criterion, as in the case of considering accrual of arguments when assessing a new cf for a dialectical tree.

Recently, argumentation has been found to be particularly powerful in different areas such as logic programming, legal reasoning, decision, and negotiation. In this context, the definition of LDS^*_{AR} as presented in this paper is a first step in our current research work to enrich traditional argumentation frameworks by incorporating numeric attributes (such as probabilities or certainty values), making them more attractive and suitable for other research and application areas. Even though many argumentation frameworks and their applications have evolved and matured in the last years, we think that many of their most promising results are still ahead.

References

[Che01] Chesñevar, C. I. *Formalización de los Procesos de Argumentación Rebatible como Sistemas Deductivos Etiquetados.* PhD thesis, Departamento de Ciencias de la Computación - U.N.S. - Argentina, January 2001.

[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. Introducing Probabilistic Reasoning in Defeasible Argumentation Using Labeled Deductive Systems. In *Proc. of the Workshop for Researchers in Computer Science* (May 2002), Bahía Blanca, Argentina.

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

[Gar00] García, A. J. *Programación en Lógica Rebatible: Lenguaje, Semántica Operacional y Paralelismo.* PhD thesis, Dep. de Cs. de la Computación, Universidad Nacional del Sur, December 2000.

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

[Pea88] Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, 1988.

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

[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.

[Sho76] Shortliffe, E. H. *Computer-based medical consultation: MYCIN.* New York, Elsevier, 1976.

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

[TP01] Tamma, V., and Parsons, S. Argumentation and qualitative probabilistic resasoning using the kappa calculus. In *Proc. of the 6th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertaintiy – Workshop "Adventures in Argumentation"* (September 2001), Toulouse, France.

[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.