ECAI 2014
T. Schaub et al. (Eds.)

doi:10.3233/978-1-61499-419-0-585

Comparing Defeasible Logics

Michael J. Maher[1]

Abstract. In this paper we seek to formally establish the similarities and differences between two formalizations of defeasible reasoning: the defeasible logics of Nute and Maier, and defeasible logics in the framework of Antoniou et al. Both families of logics have developed from earlier logics of Nute, but their development has followed different paths and they are formulated very differently. We examine these logics from the standpoint of relative inference strength – how much the logics can infer from a given theory – and relative expressiveness – how well one logic can simulate another. We identify similarities between logics in the two families and pinpoint aspects that distinguish them.

Introduction

Defeasible logics provide facilities for expressing and reasoning with defeasible knowledge. In this paper we focus on two families of defeasible logics. Both families of logics have their root in initial investigations of Nute [15] into defeasible reasoning, but have diverged both in their style of formulation and in specifics of the logics. The logics ADL and NDL were introduced and studied by Nute and Maier in [16, 14, 13]. Proofs in these logics are witnesses to their conclusions, rather than a sequence of inferences. On the other hand, $\mathbf{WFDL}$ is a collection of logics where proofs are sequences of tagged literals, in more traditional style. The $\mathbf{WFDL}$ logics were introduced in [11, 1] as part of a larger framework of defeasible logics, developed from an earlier defeasible logic DL [2]. A more thorough treatment is in development [12].

The closest counterparts to ADL and NDL in $\mathbf{WFDL}$ are the logics $\mathbf{WFDL}(\delta^*)$ and $\mathbf{WFDL}(\partial^*)$, respectively. ADL and NDL differ on how they treat ambiguity in the "truth" status of literals: whether they allow ambiguity to propagate (ADL) or block it (NDL).[2] $\mathbf{WFDL}(\delta^*)$, like ADL, propagates ambiguity and $\mathbf{WFDL}(\partial^*)$, like NDL, blocks ambiguity. In what follows we closely examine the relationship between these pairs of logics in terms of what they can infer from the same theory, and whether one logic can simulate another.

In the following sections we define the logics, and discuss some surface differences. We also show that, despite great differences in formulation, the proof systems underlying each family are essentially the same. The subsequent section establishes the relative inference strength of the logics for purely defeasible reasoning. The final section first shows that NDL and $\mathbf{WFDL}(\partial^*)$ have incomparable expressiveness, in the sense of [8]. However, by allowing some extra flexibility in the definitions we come to a more nuanced view of the relationship between these two logics.

[1] School of Engineering and Information Technology
University of New South Wales, Canberra
ACT 2600, Australia
E-mail: michael.maher@unsw.edu.au

[2] For further discussion of ambiguity and its treatment, see [18, 14, 6].

Defeasible Logics

We are addressing logics from two different families, but both families are based around rules of the same kinds: *Strict rules* express a monotonic inference. They have the form

$$r: \quad b_1, \ldots b_n \to a$$

where $a, b_1, \ldots b_n$ are literals, and r is the name of the rule. We write $A(r)$ for the antecedent of r, that is, $b_1, \ldots b_n$, and call a the consequent. *Defeasible rules* have the form

$$r: \quad b_1, \ldots b_n \Rightarrow a$$

which expresses that if $b_1, \ldots b_n$ have been established then we have reason to believe a. However, these rules are defeasible (i.e. defeatable), meaning that other rules might cast doubt on a and prevent a conclusion being drawn. The logics incorporate a priority (or superiority) relation on rules that may allow these doubts to be overcome and a conclusion to be drawn.

For example, we might encode the *Tweety* problem as follows.

$$\begin{array}{lrcl} r: & bird & \Rightarrow & flies \\ s: & penguin & \Rightarrow & \neg flies \\ & penguin & \to & bird \\ & s > r & & \end{array}$$

The logics also admit *defeater* rules that can prevent the drawing of a conclusion, but cannot be used to draw a conclusion. They will not play a significant role in this paper (because they are treated identically by the logics under discussion).

Defeasible logics draw two kinds of conclusions: positive conclusions express that a literal has been proved, while negative conclusions express that the logic recognises that the literal cannot be proved positively. Of course, it is possible that neither conclusion applies to a literal.

WFDL

This description of $\mathbf{WFDL}$ is based on [11, 10, 12]. A defeasible theory D has the form $(F, R, >)$ where F is a set of facts, R is a set of rules, and $>$ is a priority relation on rules. We write $\Lambda(D)$ for the set of rule names and $\Sigma(D)$ for the set of literals addressed by D. Conclusions that can be drawn from a theory have the form $+d\,q$ or $-d\,q$, where q is a literal and d is a tag denoting the inference rule that was applied to derive the conclusion. For brevity, we write $\pm X$ meaning "$+X$ (respectively $-X$)".

$\mathbf{WFDL}$ is a family of logics within the framework of [1]. An individual logic is named via the principal tag d (and thus, the principal inference rule) used in the logic. Thus $\mathbf{WFDL}(d)$ denotes the logic employing the $\pm d$ inference rules (as well as monotonic inference rules). There are four main logics in $\mathbf{WFDL}$, corresponding to the tags ∂, δ, ∂^*, and δ^*. A proof in $\mathbf{WFDL}(d)$ is a sequence of conclusions such that each conclusion of the form $+d\,q$ can be inferred by the $+d$ inference rule from the preceding conclusions, and each conclusion of the form $-d\,q$ occurs in a d-unfounded set wrt the set of preceding conclusions. However, rather than dealing with proofs directly, we will aggregate all the conclusions that can be inferred in one step. Given a theory D, $\mathcal{W}_D$ denotes the function that maps a current set of conclusions to the set of all conclusions that can be inferred in one inference step. The function $\mathcal{W}_D$ decomposes into two parts: $\mathcal{T}_D$, which infers positive conclusions, and $\mathcal{U}_D$, which infers negative conclusions. For a set of conclusions, or extension, E, we write $\pm d_E = \{q \mid \pm d\,q \in E\}$.

The tag Δ is used for monotonic inference. All logics in $\mathbf{WFDL}$ are assumed to contain this tag. In the following definitions, R_s (R_{sd}) is the set of strict rules (respectively, strict or defeasible rules) in R. For any set of rules S and literal q, $S[q]$ denotes the set of rules in S with consequent q. The $+\Delta$ inference rule is simply rule application, while a Δ-unfounded set is essentially the unfounded set of [7].

$+\Delta$) $+\Delta q \in \mathcal{T}_D(E)$ iff either
.1) $q \in F$; or
.2) $\exists r \in R_s[q]$ such that
.1) $\forall a \in A(r), a \in +\Delta_E$

A set S of literals is Δ-*unfounded* with respect to an extension E if: For every literal s in S, $s \notin F$ and for every strict rule $B \rightarrow s$ either

- $B \cap -\Delta_E \neq \emptyset$, or
- $B \cap S \neq \emptyset$

We now present the definitions for the defeasible inference rules in $\mathbf{WFDL}(\partial^*)$. Implicit in these definitions is that a literal q can only be contradicted by its negation $\sim q$.

$+\partial^*$) $+\partial^* q \in \mathcal{T}_D(E)$ iff either
.1) $q \in +\Delta_E$; or
.2) $\exists r \in R_{sd}[q]$ such that
.1) $\forall a \in A(r), a \in +\partial^*_E$, and
.2) $\sim q \in -\Delta_E$, and
.3) $\forall s \in R[\sim q]$ either
.1) $\exists a \in A(s), a \in -\partial^*_E$; or
.2) $r > s$.

A set S of literals is ∂^*-*unfounded* with respect to an extension E if: For every literal q in S, $q \in -\Delta_E$ and for every strict or defeasible rule $r : A(r) \hookrightarrow q$ in D either

- $A(r) \cap -\partial^*_E \neq \emptyset$, or
- $A(r) \cap S \neq \emptyset$, or
- $\sim q \in +\Delta_E$, or
- there is a rule $s : A(s) \hookrightarrow \sim q$ in D such that $A(s) \subseteq +\partial^*_E$ and $r \not> s$

For any logic $\mathbf{WFDL}(d)$, let $\mathcal{U}_D(E)$ denote the greatest d-unfounded set wrt E. We define $\mathcal{W}_D(E) = \mathcal{T}_D(E) \cup \mathcal{U}_D(E)$ and the least fixedpoint of $\mathcal{W}_D$ is the set of all conclusions from D in $\mathbf{WFDL}(d)$, denoted by $WF_d(D)$. If $c \in WF_d(D)$ then we write $D \vdash c$.[3]

There is no room here to fully define inference for $\mathbf{WFDL}(\delta^*)$. It involves inference rules for $+\delta^*$ and $+\sigma^*$, and the definitions of δ^*-unfounded and σ^*-unfounded sets, defined mutually recursively. The inference rules can be found in the appendix of [6]. The definition of δ^*-unfounded set can be obtained from the definition of ∂^*-unfounded set by replacing $-\partial^*_E$ with $-\delta^*_E$ and replacing $+\partial^*_E$ with $+\sigma^*_E$. Similarly, the definition of σ^*-unfounded set is obtained by omitting $\sim q \in +\Delta_E$ and replacing $-\partial^*_E$ with $-\sigma^*_E$ and $+\partial^*_E$ with $+\delta^*_E$.

[3] In [11, 10] we used $\vdash_{WF}$, to distinguish inference in well-founded defeasible logics from inference in other defeasible logics, but there is no need to make this distinction here.

ADL and NDL

This description of ADL and NDL is based on [14]. A defeasible theory in ADL or NDL has the form $\langle R, C, \prec \rangle$ where R is a set of rules (as defined above), C is a set of subsets of Σ, and $\prec$ is a priority relation on rules. C is a set of *conflict sets*: sets of literals that are to be considered inconsistent. C is required to contain $C_{MIN} = \{\{q, \sim q\} \mid q \in \Sigma\}$. This ability to nominate inconsistent sets beyond C_{MIN} potentially adds greater expressiveness to ADL and NDL but, for this paper, we assume that the conflict sets of a theory are exactly C_{MIN} so that we can identify similarities.

Rather than an inference process, NDL and ADL use an argument tree as a witness that a conclusion can be drawn. An *argument tree* over Σ is a finite tree where each node is labelled with $+q$ or $-q$, where q is a literal in Σ. Nodes labelled with $+q$ are *positive* and with $-q$ are *negative*. The following definition simplifies the definition of [14] under our assumption that the conflict sets are exactly C_{MIN}.

Definition 1 *An argument tree is an* NDL-proof *iff for every node n in the argument tree either*

1. *The label of n is $+q$ and either*
   (a) *there is a rule $r \in R_s[q]$ with body B_r and for every $p \in B_r$, $+p$ labels a child of n*
   (b) *there is a rule $r \in R_d[q]$ with body B_r such that*
      i. *for every $p \in B_r$, $+b$ labels a child of n, and*
      ii. *for every rule $s \in R[\sim q]$ either $s \prec r$ or there is a child of n labelled $-p$ and $p \in B_s$*
2. *The label of n is $-q$ and*
   (a) *for every rule $r \in R_s[q]$ with body B_r there is a child of n labelled $-p$ and $p \in B_r$*
   (b) *for every rule $r \in R_d[q]$ with body B_r either*
      i. *there is a child of n labelled $-p$ and $p \in B_r$, or*
      ii. *there is a rule $s \in R[\sim q]$ such that $s \not\prec r$ and for every $p \in B_s$ there is a child of n labelled $+p$*
3. *The label of n is $-q$, and n has an ancestor m with label $-q$, and all nodes between m and n are negative.*

Condition 3 is called *failure-by-looping*.

An *ADL-proof* is similar to an NDL-proof: it differs only in the last clause in the second condition. That clause for ADL is:

ii there is a rule $s \in R[\sim q]$ such that for every $p \in B_s$ there is a child of n labelled $+p$, and either s is strict or $r \prec s$

For both NDL and ADL proofs, we introduce the notion of *degree* of a node r in a proof as follows:

- if r is positive,
  $degree(r) = 1 + max\{degree(n) \mid n \text{ is a child of } r\}$

- if r is negative,
$degree(r) = 1+max\{degree(n) \mid n$ is a positive descendent of $r\}$

where the maximum of the empty set is taken to be 0.

It is straightforward to show that, for any node n, a descendent has a lower or equal degree than n, with equality possible when n and the descendent are negative nodes, with no intervening positive nodes. The degree of a node will be used as the basis for induction proofs.

Syntactic and Notational Differences and Similarities

There are some cosmetic differences between theories in the two families, and some other obvious differences.

Theories in $\mathbf{WFDL}$ contain a set of facts. There is no corresponding element in most-recent presentations of theories in NDL and ADL [14, 13] although there was in an earlier presentations [16]. Facts can be expressed as strict rules with an empty body [2] so, in a minor abuse of terminology, we will refer to such strict rules as facts.

In ADL and NDL, priorities between rules are expressed by a relation $\prec$, while in $\mathbf{WFDL}$ the relation is $>$. In both cases, if $s \prec r$ or $r > s$ then r has priority over s. We will use $s < r$ and $r > s$, interchangeably, in place of $s \prec r$.

Both families identify both literals that can be inferred, and those literals for which it can be proved that they cannot be inferred, but again the notation differs. ADL and NDL use the notation $D \mid\!\sim q$ and $D \sim\!\mid q$, while $\mathbf{WFDL}$ logics use $D \vdash +d\,q$ and $D \vdash -d\,q$. To keep a uniform notation, we will write $D \vdash +\nu\, q$ and $D \vdash -\nu\, q$ for inference in NDL and $D \vdash +\alpha\, q$ and $D \vdash -\alpha\, q$ for inference in ADL.

An important difference, noted earlier, is that ADL and NDL permit a theory to nominate sets of literals to be considered inconsistent, whereas in $\mathbf{WFDL}$ the inconsistencies arise solely from a literal and its negation. However, in this paper we consider only theories where the conflict sets are precisely C_{MIN}.

Finally, inference in $\mathbf{WFDL}$ distinguishes between strict conclusions (expressed $+\Delta q$), derived from facts and strict rules, and defeasible conclusions (expressed $+d\,q$, where d is a tag identifying the specific type of defeasible inference). ADL and NDL do not make this distinction. This difference is important in addressing relative expressiveness, because ADL and NDL will need to also simulate strict conclusions to simulate logics in $\mathbf{WFDL}$.

Both the inference rules of $\mathbf{WFDL}$ and the proof conditions for ADL and NDL are *shallow* in the following sense: the inference rules rely only on the existence of certain conclusions in E, while the proof conditions only rely on the existence of appropriately labelled children of a node n. How the conclusions in E are proved, and what structure the subtrees of n have, is irrelevant. In this, the two families are similar to most logics.

However, the use of d-unfounded sets and the failure-by-looping condition are unusual, and both have been linked to the well-founded semantics of logic programs [7], in [11] and [13] respectively. We now show that these two approaches to defining logics are essentially the same.

Consider an inference rule $+d$ (in $\mathbf{WFDL}$) which states that $+dq \in \mathcal{T}_D(E)$ iff Ψ, for some condition Ψ, and a condition on an argument tree that permits a node to be labelled $+q$ iff Φ, for some condition Φ. We say an inference rule and a positive labelling condition *correspond* if Φ can be obtained from Ψ by replacing occurrences of $\pm d\,q \in E$ by "$\pm q$ is a child of n", and vice versa. Similarly, consider the definition of a d-unfounded set (in $\mathbf{WFDL}$) which states that S is d-unfounded if, for every $q \in S$ and every strict or defeasible rule r for q, either $A(r) \cap S \neq \emptyset$ or Ψ, for some condition Ψ, and a condition on an argument tree that permits a node to be labelled $-q$ iff Φ, for some condition Φ. We say the definition of d-unfounded set and a negative labelling condition *correspond* if Φ can be obtained from Ψ by replacing occurrences of $\pm d\,q \in E$ by "$\pm q$ is a child of n", and vice versa.

Now, if:

- the inference rule $+d$ (in $\mathbf{WFDL}$) corresponds to a positive labelling condition C_+,
- the definition of a d-unfounded set corresponds to a negative labelling condition C_-, and
- an argument tree is considered a proof iff every node satisfies either C_+, C_-, or the failure-by-looping condition,

then we say that the *proof conditions are equivalent*. If the proof conditions are equivalent then for every defeasible theory D and literal q, $D \vdash \pm d\,q$ iff $\pm q$ has a proof in D. In other words, the two corresponding formulations express the same logic.

Theorem 2 *Consider logic L_1 defined in the $\mathbf{WFDL}$ style and logic L_2 defined in the ADL/NDL style, and suppose their proof conditions are equivalent. Then the two logics infer the same conclusions from the same defeasible theory.*

In this theorem we permit the logic defined in the $\mathbf{WFDL}$-style to violate the Principle of Strong Negation [1], which is usually required in this framework. We also implicitly restrict the logic to using a single tag d. However, if the notation for NDL-style proofs were extended to allow multiple tags, then this theorem could also be extended to multiple tags.

Although the two styles of proof system essentially define the same logics, this theorem does not apply directly to the specific logics ADL, NDL and previously defined logics in $\mathbf{WFDL}$: their proof conditions are not equivalent.

In the next section we investigate their relationship.

Relative Inference Strength

In this section we address the relative inference strength of the logics under discussion.

There are substantial similarities between NDL and $\mathbf{WFDL}(\partial^*)$, as we now see. When a theory contains only strict information, then both logics agree. Furthermore, when a theory and addition contain no strict rules then both logics agree.

Proposition 3 *Let D and A consist only of facts and strict rules. Then*

- $D{+}A \vdash +\nu\, q$ *iff* $D{+}A \vdash +\Delta q$
- $D{+}A \vdash -\nu\, q$ *iff* $D{+}A \vdash -\Delta q$

The same result holds for ADL, since the inference rules of ADL and NDL agree on strict rules. Furthermore, as a consequence of this result, if a theory D only contains strict rules then NDL and ADL are *decisive*, that is, for every literal q either $+q$ or $-q$ is inferred.

Proposition 4 *Let D and A contain no strict rules, except for facts. Then*

- $D{+}A \vdash +\nu\, q$ *iff* $D{+}A \vdash +\partial^* q$
- $D{+}A \vdash -\nu\, q$ *iff* $D{+}A \vdash -\partial^* q$

We now combine the previous two theorems to establish a class of theories on which NDL and $\mathbf{WFDL}(\partial^*)$ agree. We say a theory D over a language Σ is *layered* if Σ can be partitioned into sets Σ_s and Σ_d such that:

- if $q \in \Sigma_d$ then also $\sim q \in \Sigma_d$,
- for every defeasible rule in D with head q, $q \in \Sigma_d$, and
- for every strict rule in D with body B, $B \subseteq \Sigma_s$

If a theory is layered, then strict rules can only be applied to conclusions that are derived using only (facts and) strict rules. Thus conclusions drawn from strict rules are monotonic. The notion of layering has some similarity to stratification in logic programs [5] but layering is based on the kind of rule used, rather than an intervening negation.

Theorem 5 *Let D be a layered theory. Then*

- *if $q \in \Sigma_s$ then $D \vdash \pm\nu\, q$ iff $D \vdash \pm\Delta q$*
- *if $q \in \Sigma_d$ then $D \vdash \pm\nu\, q$ iff $D \vdash \pm\partial^* q$*

For theories that are not layered, the behaviour of NDL and $\mathbf{WFDL}(\partial^*)$ may diverge. Neither logic is stronger than the other.

Example 6 *Consider the theory D consisting of $\{\Rightarrow a; a \rightarrow b; \Rightarrow \neg b\}$. This theory is not layered because a must be in both Σ_d (by the second condition of layering) and Σ_s (by the third condition). We have $D \vdash +\nu\, b$ but $D \vdash -\partial^* b$. If we add $\{b \Rightarrow c; \Rightarrow \neg c\}$ to D then $D \vdash -\nu\, \neg c$ but $D \vdash +\partial^* \neg c$.*

An equivalent result to Proposition 4 does not hold for ADL and $\mathbf{WFDL}(\delta^*)$, as the following example shows.

Example 7 *Consider the following defeasible theory D:*

$$\begin{array}{rccc} r: & & \Rightarrow & q \\ s: & p & \Rightarrow & \neg q \\ & & \Rightarrow & p \end{array}$$

In $\mathbf{WFDL}$ we can conclude $+\sigma^* p$ and hence $-\delta^* q$ because $r \not> s$. On the other hand, in ADL we cannot conclude $-\alpha\, q$, because we do not have $r < s$, nor can we conclude $+\alpha\, q$. In this example, we see the effect of the use of $r \not> s$ in the inference rule for $-\delta^*$ versus the use of the more restrictive $r < s$ in the ADL inference rule.

If we extend D with $\{\Rightarrow \neg p\}$ and the superiority statement $r < s$, then in $\mathbf{WFDL}$ we can conclude $+\sigma^* p$ and hence $-\delta^* q$ because $r \not> s$. On the other hand, in ADL we cannot conclude $+\alpha\, p$ because the rule for $\neg p$ does not fail and is not inferior to the rule for p. Consequently, we cannot conclude $-\alpha\, q$, even though $r < s$. In this example we see that the weakness of the requirement $+\sigma^* p$ allows $\mathbf{WFDL}(\delta^*)$ to infer that q fails whereas the stronger requirement $+\alpha\, p$ for the ADL inference rule leaves that rule inapplicable.

In these examples we see that ADL is weaker at drawing negative conclusions than $\mathbf{WFDL}(\delta^*)$. However, we can still characterize the conclusions of ADL in terms of that logic.

Theorem 8 *Let D be a defeasible theory containing no strict rules (except facts). Then*

- *$D \vdash +\alpha\, q$ iff $D \vdash +\delta^* q$*
- *$D \vdash -\alpha\, q$ iff $D \vdash -\sigma^* q$*

Thus we see that, for purely defeasible reasoning, inference in ADL is essentially a hybrid of the two inferences in $\mathbf{WFDL}(\delta^*)$: the main inference rules δ^* and the inference rules for support σ^*. $+\delta^*$ is the weakest positive inference rule in $\mathbf{WFDL}$, and $-\sigma^*$ is the weakest negative inference rule. Thus ADL is the weakest logic, in terms of inference strength, of the logics under discussion.

Combining Theorems 5 and 8, and results of [12], we have the following relationship between ADL, NDL and $\mathbf{WFDL}$ for layered theories. For a fixed theory D, let $\pm d = \{q \mid D \vdash \pm d\, q\}$.

Theorem 9 *Let D be a layered defeasible theory. Then*

$$-\alpha = -\sigma^* \subseteq -\nu = -\partial^* \subseteq -\delta^*$$

and

$$+\alpha = +\delta^* \subseteq +\nu = +\partial^* \subseteq +\sigma^*$$

Relative Expressiveness

We now turn to the question of relative expressiveness and simulation. [8, 9] introduced a framework for addressing the relative expressiveness of defeasible logics. The framework identifies the greater (or equal) expressiveness of L_2 compared to L_1 with the ability to simulate any theory D in a logic L_1 by a theory $T(D)$ in the logic L_2, in the presence of an addition.

The *addition* of a theory A to a theory D is denoted by $D + A$. Addition is essentially the union of the theories: Let $D = (F, R, >)$ and $A = (F', R', >')$. Then $D + A = (F \cup F', R \cup R', > \cup >')$. $\Lambda(D{+}A) = \Lambda(D) \cup \Lambda(A)$ and $\Sigma(D{+}A) = \Sigma(D) \cup \Sigma(A)$. The logics under discussion do not have the concept of modules, or any other encapsulation mechanism, so this notion is instead built in to the kind of additions permitted. Given a theory D and a possible simulating theory D', we say an addition A is *modular* if $\Sigma(A) \subseteq \Sigma(D)$, $\Lambda(D) \cap \Lambda(A) = \emptyset$, and $\Lambda(D') \cap \Lambda(A) = \emptyset$. We will consider specific classes of additions but, for each class and any D and D', only the modular additions in the class will be considered.

Since different logics involve different tags, conclusions from theories in different logics cannot be identical. Furthermore, one logic might involve more tags than another. Hence we must allow the use of a term to help represent a tag. A tag representation scheme for the simulation of a logic L_1 by the logic L_2 associates to every tag d_1 of interest in L_1, a tag d_2 in L_1 and a term $t(.)$. Given such a tag representation scheme, we say two conclusions α in L_1 and β in L_2 are *equal modulo tags* if α is $+d_1 q$ and β is $+d_2 t(q)$ or α is $-d_1 q$ and β is $-d_2 t(q)$. This formulation generalizes the formulation in [8].

We now have the following definition of simulation and relative expressiveness. For the motivations behind this definition, see [8].

Definition 10 *Let C be a class of defeasible theories, and fix a tag representation scheme for the simulation of L_1 by L_2.*

We say D_1 in logic L_1 is simulated *by D_2 in L_2 with respect to a class C if, for every modular addition A in C, $D_1 + A$ and $D_2 + A$ have the same conclusions in $\Sigma(D_1 + A)$, modulo tags.*

We say a logic L_1 can be simulated by a logic L_2 with respect to a class C if every theory in L_1 can be simulated by some theory in L_2 with respect to additions from C.

We say L_2 is more (or equal) expressive *than L_1 if L_1 can be simulated by L_2 with respect to C.*

The two classes C that we consider are the defeasible theories defined solely by *facts*, and those defined solely by *rules*.

Apart from [8, 9, 10], which compare logics within the framework of [1], work relating defeasible logics to other formalisms corresponds to relative expressiveness without additions. Such work includes [4, 3, 13].

Although NDL and $\mathbf{WFDL}(\partial^*)$ perform identically on layered theories, they differ when *strict* rules are used to derive *defeasible* information. Specifically, in $\mathbf{WFDL}$ the use of a strict rule to draw a defeasible conclusion is no different from the use of a similar defeasible rule and, in particular, a strict rule may be inferior to a defeasible rule, and may be overruled by that defeasible rule when drawing defeasible conclusions from defeasible information. In contrast, in NDL (and ADL) a strict rule is treated as if it is superior to all competing rules when attempting to derive a conclusion, and strict rules do not participate in the superiority relation.

As a result of these different treatments of strict rules, the addition of rules can distinguish the logics from the two families: they have incomparable expressiveness. This result seems obvious when we consider the addition of the rule $a \to b$ to the theory $\{\Rightarrow a; \Rightarrow \neg b\}$, as in Example 6.

Proposition 11 *For every $d \in \{\partial, \partial^*, \delta, \delta^*\}$,*

- *Neither NDL nor ADL can simulate $\mathbf{WFDL}(d)$ with respect to addition of rules*
- *$\mathbf{WFDL}(d)$ cannot simulate either NDL or ADL with respect to addition of rules*

The two families of logic also have incomparable expressiveness with respect to addition of facts. We first show that $\mathbf{WFDL}$ cannot simulate NDL or ADL.

Theorem 12 *For every $d \in \{\partial, \partial^*, \delta, \delta^*\}$, $\mathbf{WFDL}(d)$ cannot simulate either NDL or ADL with respect to addition of facts*

This result arises from the ability of NDL and ADL to derive inconsistent conclusions on the basis of defeasible information, using the strict rules. In contrast, in $\mathbf{WFDL}$ inconsistent conclusions can be derived only as a result of inconsistency in the monotonic part of the theory (the facts and strict rules).

For example, if D is $\{a \Rightarrow a'; a' \to b; c \Rightarrow c'; c' \to \neg b\}$ and we add the facts $\{a; c\}$ then in both families we infer a' and c' defeasibly. However, in NDL and ADL we infer both b and $\neg b$ defeasibly, while in $\mathbf{WFDL}$ neither can be inferred.

As noted earlier, ADL and NDL do not distinguish monotonic consequences from defeasible consequences. Consequently, for these logics to simulate a $\mathbf{WFDL}$ logic they must have a method of representing monotonic conclusions. We will use the proposition $strict(q)$ to refer to the status of the literal q as a monotonic consequence. That is, we will interpret the conclusion $+\nu\ strict(q)$ as representing $+\Delta q$ and interpret $-\nu\ strict(q)$ as representing $-\Delta q$. (More briefly, we say that $\pm\nu\ strict(q)$ represents $\pm\Delta q$.)

This flexibility is still not enough to allow ADL and NDL to simulate $\mathbf{WFDL}$. The problem is that ADL and NDL need to separate the reasoning at the monotonic and defeasible levels from the input data (i.e., the additions). Although $strict$ allows the separation of reasoning at the monotonic level, the input and defeasible conclusions are still not distinguished.

For example, consider the following theory D in $\mathbf{WFDL}$:

$$\begin{array}{ccc} a & \Rightarrow & b \\ b & \to & c \end{array}$$

Any simulation of D must allow $strict(c)$ to be derived from an addition b. However, any simulation must also allow b to be derived from an addition a. Clearly, then, the simulation must allow the derivation of $strict(c)$ from a, which is an invalid conclusion in $\mathbf{WFDL}$. This is the basis of the proof of the following result.

Theorem 13 *For every $d \in \{\partial, \partial^*, \delta, \delta^*\}$, ADL and NDL cannot simulate $\mathbf{WFDL}(d)$ wrt addition of facts.*

However, NDL *can* simulate $\mathbf{WFDL}(\partial^*)$ if we permit it to represent defeasible inference $+\partial^* q$ other than by $+\nu\, q$. In particular, in the following simulation we represent $\pm\partial^* q$ by $\pm\nu\, def(q)$ (as well as representing $\pm\Delta q$ by $\pm\nu\, strict(q)$).

Definition 14 *Let Σ be the language of a defeasible theory $D = (F, R, >)$ in $\mathbf{WFDL}$, and define $D' = (R', C_{MIN}, <)$ as follows.*

1. *For each fact $q \in F$ or strict rule in R with empty body for q, R' contains*

$$\to \quad q$$

2. *For each $q \in \Sigma$, R' contains*

$$q \quad \to \quad strict(q)$$

3. *For each strict rule $r = q_1, \ldots, q_n \to q$ in R, with $n > 0$, R' contains*

$$s(r): \quad strict(q_1), \ldots, strict(q_n) \quad \to \quad strict(q)$$

4. *For each proposition $p \in \Sigma$, R' contains*

$$\begin{array}{rcl} strict(p) & \to & d(p) \\ strict(\neg p) & \to & \neg d(p) \\ d(p) & \Rightarrow & def(p) \\ \neg d(p) & \Rightarrow & def(\neg p) \end{array}$$

5. *For each $r = p_1, \ldots, p_n, \neg p_{n+1}, \ldots, \neg p_m \hookrightarrow_r p$ in R that is either a defeasible rule or a defeater, R' contains*

$$r: \quad d(p_1), \ldots, d(p_n), \neg d(p_{n+1}), \ldots, \neg d(p_m) \quad \hookrightarrow_r \quad d(p)$$

where $p_1, \ldots p_n, \ldots, p_m, p$ are propositions.
Similarly, for each $r = p_1, \ldots, p_n, \neg p_{n+1}, \ldots, \neg p_m \hookrightarrow_r \neg p$ in R that is either a defeasible rule or a defeater, R' contains

$$r: \quad d(p_1), \ldots, d(p_n), \neg d(p_{n+1}), \ldots, \neg d(p_m) \quad \hookrightarrow_r \quad \neg d(p)$$

6. *For each strict rule $r = p_1, \ldots, p_n, \neg p_{n+1}, \ldots, \neg p_m \to p$ in R, R' contains*

$$r: \quad d(p_1), \ldots, d(p_n), \neg d(p_{n+1}), \ldots, \neg d(p_m) \quad \Rightarrow \quad d(p)$$

where $p_1, \ldots p_n, \ldots, p_m, p$ are propositions.
Similarly, for each strict rule
$r = p_1, \ldots, p_n, \neg p_{n+1}, \ldots, \neg p_m \to \neg p$ in R, R' contains

$$r: \quad d(p_1), \ldots, d(p_n), \neg d(p_{n+1}), \ldots, \neg d(p_m) \quad \Rightarrow \quad \neg d(p)$$

In this transformation we are using $\pm\nu\ strict(q)$ to represent $\pm\Delta q$ and using $\pm\nu\ def(q)$ to represent $\pm\partial^* q$. We are also using intermediate propositions $d(p)$ for each proposition $p \in \Sigma$. These propositions are used so that competing rules in D are represented by corresponding competing rules in D'. We are using the minimal conflict sets in D', mimicking the implicit conflicts in $\mathbf{WFDL}$, and the same superiority relation in D' as in D, although the rules that that relation refers to have been slightly modified.

Parts 1, 2 and 3 represent $\mathbf{WFDL}$ monotonic reasoning in NDL. Part 4 expresses that all strict conclusions are also defeasible conclusions. It also provides the conversion of the intermediate form of defeasible conclusions (using d) into the final form (using def). Parts 5 and 6 represent the rules of D to be used for defeasible reasoning in NDL. Note that strict rules of D are represented by defeasible rules in D' for defeasible reasoning, so that they do not automatically overrule competing defeasible rules and so that they can participate in the superiority relation.

Theorem 15 *NDL can simulate $\mathbf{WFDL}(\partial^*)$ wrt addition of facts when we permit $\pm\nu\ strict(q)$ to represent $\pm\Delta q$ and $\pm\nu\ def(q)$ to represent $\pm\partial^* q$.*

As observed following Theorem 12, the logics of $\mathbf{WFDL}$ cannot simulate NDL because of the ability of NDL to derive inconsistent conclusions on the basis of defeasible knowledge. $\mathbf{WFDL}$ can only derive inconsistent conclusions when strict knowledge is inconsistent. However, if we restrict our attention to theories and additions that are strongly consistent then we can simulate NDL with $\mathbf{WFDL}(\partial^*)$.

Let D be a defeasible theory for NDL. We say an addition A is *strongly consistent with* D if, for no literal $q \in \Sigma$, $D + A \vdash +\nu\, q$ and $D + A \not\vdash -\nu \sim\!q$. We say that NDL is *simulated wrt strongly consistent addition of facts* if, for any defeasible theory D in NDL, there is a defeasible theory D' in $\mathbf{WFDL}(\partial^*)$ such that, for every addition A of facts strongly consistent with D, and for every literal $q \in \Sigma$,

- $D{+}A \vdash +\nu\, q$ iff $D{+}A \vdash +\partial^* q$
- $D{+}A \vdash -\nu\, q$ iff $D{+}A \vdash -\partial^* q$

We now define a transformation for simulating NDL by $\mathbf{WFDL}(\partial^*)$ in this limited sense. It introduces a new proposition $s(q)$ for each literal $q \in \Sigma$.

Definition 16 *Let Σ be the language of a defeasible theory $D = (R, C_{MIN}, <)$ for NDL, and define $D' = (\emptyset, (R\backslash R_s)\cup R', > \cup >')$ as follows.*

1. *For each strict rule $r = B \to q$ in R, R' contains*

$$r: \quad B \quad \Rightarrow \quad s(q)$$

2. *For each literal $q \in \Sigma$, R' contains*

$$strict(q): \quad s(q) \quad \Rightarrow \quad q$$

3. *For each $q \in \Sigma$ and, for every defeasible rule or defeater r' for $\sim\!q$ in D, $>'$ contains $strict(q) >' r'$. (Note that we do not have $strict(q) > strict(\sim\!q)$, or vice versa, so the resulting superiority relation is acyclic.)*

In this transformation, the implicit priority of strict rules in NDL is represented by the intermediate literals $s(q)$ and the superiority of the rules $strict(q)$ over competing rules. The existing defeasible rules, defeaters and superiority relations are unaltered.

Theorem 17 *NDL can be simulated by $\mathbf{WFDL}(\partial^*)$ wrt strongly consistent addition of facts (where $\pm\nu\, q$ is represented by $\pm\partial^* q$).*

Finally, we compare the expressiveness of ADL and NDL.

Theorem 18

- *ADL cannot simulate NDL with respect to addition of rules*
- *NDL cannot simulate ADL with respect to addition of rules*

Thus NDL and ADL have different expressiveness, with respect to addition of rules. In [8] a similar result was proved, showing that the ambiguity blocking and ambiguity propagating logics of $\mathbf{DL}$ have different expressiveness wrt rules. In [9] simulations between those logics were shown wrt addition of facts, but it remains to be seen if similar results can be established for ADL and NDL

Conclusion

We have established close relationships between two families of defeasible logics. First, we showed that differences in the formulation of proof in the two families do not lead to different semantics. We showed that defeasible reasoning in NDL and $\mathbf{WFDL}(\partial^*)$ is identical for layered theories. In contrast, we showed that defeasible reasoning in ADL is weaker than other logics in the two frameworks, but we characterized its inference in terms of $\mathbf{WFDL}(\delta^*)$. For relative expressiveness and simulation the results are more nuanced. In general, neither framework is fully able to simulate the other. However, we demonstrated simulations of $\mathbf{WFDL}(\partial^*)$ by NDL and vice versa, under some restrictions.

Open questions remain concerning the relative expressiveness of ADL compared to NDL and $\mathbf{WFDL}(\delta^*)$, the effect of general conflict sets on expressiveness, and the relation between defeasible logics and concrete argumentation in frameworks such as ASPIC+ [17].

Acknowledgements: The author thanks the referees for their careful reviewing.

REFERENCES

[1] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher, 'A flexible framework for defeasible logics', in *AAAI/IAAI*, pp. 405–410. AAAI Press / The MIT Press, (2000).

[2] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher, 'Representation results for defeasible logic', *ACM Trans. Comput. Log.*, **2**(2), 255–287, (2001).

[3] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher, 'Embedding defeasible logic into logic programming', *TPLP*, **6**(6), 703–735, (2006).

[4] Grigoris Antoniou, Michael J. Maher, and David Billington, 'Defeasible logic versus logic programming without negation as failure', *J. Log. Program.*, **42**(1), 47–57, (2000).

[5] Krzysztof R. Apt, Howard A. Blair, and Adrian Walker, 'Towards a theory of declarative knowledge', in *Foundations of Deductive Databases and Logic Programming.*, 89–148, Morgan Kaufmann, (1988).

[6] David Billington, Grigoris Antoniou, Guido Governatori, and Michael J. Maher, 'An inclusion theorem for defeasible logics', *ACM Trans. Comput. Log.*, **12**(1), 6, (2010).

[7] Allen Van Gelder, Kenneth A. Ross, and John S. Schlipf, 'The well-founded semantics for general logic programs', *J. ACM*, **38**(3), 620–650, (1991).

[8] Michael J. Maher, 'Relative expressiveness of defeasible logics', *TPLP*, **12**(4-5), 793–810, (2012).

[9] Michael J. Maher, 'Relative expressiveness of defeasible logics II', *TPLP*, **13**(4-5), 579–592, (2013).

[10] Michael J. Maher, 'Relative expressiveness of well-founded defeasible logics', in *Proc. Australasian Joint Conf. on Artificial Intelligence*, (2013).

[11] Michael J. Maher and Guido Governatori, 'A semantic decomposition of defeasible logics', in *AAAI/IAAI*, pp. 299–305. AAAI Press, (1999).

[12] Michael J. Maher, Guido Governatori, and Ho Pun Lam. Well-founded defeasible logics, manuscript.

[13] Frederick Maier, 'Interdefinability of defeasible logic and logic programming under the well-founded semantics', *TPLP*, **13**, (2013).

[14] Frederick Maier and Donald Nute, 'Well-founded semantics for defeasible logic', *Synthese*, **176**(2), 243–274, (2010).

[15] Donald Nute, 'Defeasible logic', in *Handbook of Logic in Artificial Intelligence and Logic Programming, Vol. III*, eds., D. Gabbay, C. Hogger, and J. Robinson, 353–395, Oxford University Press, (1994).

[16] Donald Nute, 'Defeasible logic', in *INAP (LNCS Volume)*, eds., Oskar Bartenstein, Ulrich Geske, Markus Hannebauer, and Osama Yoshie, volume 2543 of *Lecture Notes in Computer Science*, pp. 151–169. Springer, (2001).

[17] Henry Prakken, 'An abstract framework for argumentation with structured arguments', *Argument and Computation*, **1**, 93–124, (2010).

[18] David S. Touretzky, John F. Horty, and Richmond H. Thomason, 'A clash of intuitions: The current state of nonmonotonic multiple inheritance systems', in *IJCAI*, pp. 476–482, (1987).