Logical Properties in Defeasible Logic Programming – a preliminary report –

A. G. Stankevicius[*] M. Capobianco C. I. Chesñevar

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
Bahía Blanca - Buenos Aires - ARGENTINA
e-mail: {ags,mc,cic}@cs.uns.edu.ar

1 Introduction

Logics for nonmonotonic reasoning have often been described by the property they lack—that is, monotonicity—instead of by those they do enjoy. These theories flourished in the early '80s in response to the inconveniences incomplete and changing information posed to classic, monotonic approaches. Several nonmonotonic formalisms were introduced in the literature: inheritance networks, default logic, preferential entailment, autoepistemic logic, and defeasible argumentation among others. The introduction of these proposals in a short span of time made it difficult to decide which approach is best suited for a given context.

In a landmark paper, Dov Gabbay [3] pioneered the comparison of nonmonotonic theories with respect to a set of desirable properties. He endorsed focusing our attention on the properties of the inference relation induced by each system, that is, the relation between a conclusion and the set of premises supporting it. Further pursuing this approach, first Kraus *et al.* [5], and later Makinson [6], studied which set of core properties every nonmonotonic theory must have. These properties can be roughly divided in the so-called *pure* conditions, that solely depend on the inference relation, and those that also interact with the logical connectives. For any inference relation $\mid\!\sim$ its corresponding *inference operator*, noted C, is defined as $C(\Phi) = \{\phi \,|\, \Phi \mid\!\sim \phi\}$. Assuming that Φ and Ψ are sets of premises, the pure conditions can be defined in terms of this operator as follows:

Inclusion: $\Phi \subseteq C(\Phi)$

Idempotence: $C(\Phi) = C(C(\Phi))$

Cut: $\Phi \subseteq \Psi \subseteq C(\Phi)$ implies that $C(\Phi) \subseteq C(\Psi)$

Cautious monotonicity: $\Phi \subseteq \Psi \subseteq C(\Phi)$ implies that $C(\Psi) \subseteq C(\Phi)$

Inclusion is clearly desirable in the context of any sensible inference relation, as it stands for accepting the premises upon which we reason. Idempotence shows we have inferred as much

[*] Becario de la Comisión de Investigaciones Científicas de la Provincia de Buenos Airea.

as possible, in the sense that no new conclusions can be obtained from Φ through the operator C. Cut ensures that adding known consequences into our set of premises does not give us new conclusions. Cautious monotonicity, the dual of cut, states the opposite: adding known consequences into our set of premises does not make us loose conclusions. Those inference relations satisfying all these properties are called *cumulative.*[1]

Some authors have suggested that every nonmonotonic theories should be engineered around a cumulative inference relation. For instance, Kraus *et el.* [5] pushed this stance further claiming that *"...[these properties] are rock-bottom properties without which a system should not be considered a logical system."* In what follows we study whether the entailment relations induced by a particular nonmonotonic theory are cumulative.

2 Our Study Case

Defeasible Logic Programming (DeLP, henceforth), combines a language similar to the one of logic programming with an argumentative inference engine [10, 9, 4]. DeLP structures knowledge using a set of facts, denoting the evidence at hand, also complemented with two meta-linguistic relations written as rules. On the one hand, *strict rules* represent incontrovertible information (*e.g.*, being a penguin is an incontrovertible reason for being a bird). On the other hand, *defeasible rules* represent disputable information (*e.g.*, being a bird is a disputable reason for being able to fly). Thus, a defeasible logic program $\mathcal{P}$ is just a tuple (Π, Δ) where Π is a set of facts and strict rules, and Δ a set of defeasible rules. The conclusions endorsed by a given a program $\mathcal{P}$ are obtained through a dialectical process based on three auxiliary inference relations. In the remainder of this section we consider what properties are met by each of these relations.

2.1 Strict and defeasible inference

The first inference relation we consider is called *strict inference* as it only concerns strict rules. In this relation, a conclusion ϕ is inferred from a program $\mathcal{P}$, noted $\mathcal{P} \vdash \phi$, when it can be derived from $\mathcal{P}$ simply by interpreting its strict rules as inference rules. The second relation, called *defeasible inference*, allows the use of both strict and defeasible rules in order to form conclusions. We write $\mathcal{P} \mid\!\sim \phi$ when a conclusion ϕ is defeasibly inferred from a program $\mathcal{P}$. Even though these two relations look rather simple-minded, they play a fundamental role within DeLP: they are the building blocks of the more elaborate notions of argument and warrant.

Let $C_{\vdash}$ and $C_{\mid\sim}$ be the consequence operators associated respectively to strict and defeasible inference, that is to say, $C_{\vdash}(\Phi) = \{\phi \mid \Phi \vdash \phi\}$ and $C_{\mid\sim}(\Phi) = \{\phi \mid \Phi \mid\!\sim \phi\}$. The following proposition summarizes our findings about these relations:

Proposition 2.1 *The consequence operators $C_{\mid\sim}$ and $C_{\vdash}$ satisfy inclusion, idempotence, cut, and cautious monotonicity.*

It is straightforward to see that $C_{\mid\sim}$ and $C_{\vdash}$ are both monotonic, and therefore cumulative. Any standard proof of monotonicity from classical logic also applies here.

It should be mentioned that the strict part of any defeasible logic program is by definition bound to be consistent with respect to strict inference, *i.e.*, no pair of contradictory literals can be derived.

[1] for an excellent survey of these and other abstract properties we refer the interested reader to [1].

2.2 Argument-based inference

The last auxiliary inference relation concerns the construction of *arguments*. Simply put, an argument is a tentative piece of reasoning supporting a certain conclusion. In this system, an argument is the subset of defeasible rules that allows to defeasibly derive a given conclusion. We say that ϕ is an *argument-based inference* from a program $\mathcal{P}$, noted as $\mathcal{P} \mid\!\sim_{\mathcal{A}} \phi$, if it is possible to build an argument for ϕ using the defeasible rules in $\mathcal{P}$. The corresponding consequence operator for this relation is defined as $C_{\mathcal{A}}(\Phi) = \{\phi \mid \Phi \mid\!\sim_{\mathcal{A}} \phi\}$. This inference relation greatly differs from the previous one.

Proposition 2.2 *The consequence operator $C_{\mathcal{A}}$ satisfies inclusion, but fails idempotence, cut, and cautious monotonicity.*

Inclusion trivially holds, given that every fact is supported by the empty argument (note that facts can be strictly derived). The failure of idempotence is shown next:[2]

Example 2.1 Let $\mathcal{P} = (\Pi, \Delta)$ be a DeLP program, where $\Pi = \{a\}$ and $\Delta = \{b \prec a; \sim b \prec a\}$. Note that in this case $C_{\mathcal{A}}(\{a\}) = \{b; \sim b\} \neq \emptyset = C_{\mathcal{A}}(C_{\mathcal{A}}(\{a\}))$.

Finally, the following counterexamples show that argument-based inference does not satisfy cut nor cautious monotonicity.

Example 2.2 Consider the DeLP program $\mathcal{P} = (\Pi, \Delta)$, where $\Pi = \{b;\ d;\ \sim c \leftarrow a\}$ and $\Delta = \{a \prec b, d;\ b \prec c;\ c \prec d\}$. In this setting, it is possible to build the argument $\mathcal{A}_1 = \{b \prec c;\ c \prec d\}$ for b. However, should b be added to $\mathcal{P}$ as a fact, we now can obtain a new argument $\mathcal{A}_2 = \{a \prec b, d\}$ for a, previously unavailable.

Example 2.3 Let $\mathcal{P} = (\Pi, \Delta)$ be a DeLP program, where $\Pi = \{a\}$ and $\Delta = \{b \prec a; \sim b \prec a\}$. In this context, the arguments $\mathcal{A}_1 = \{b \prec a\}$ for b and $\mathcal{A}_2 = \{\sim b \prec a\}$ for $\sim b$ can be constructed from $\mathcal{P}$. However, should b be added to $\mathcal{P}$ as a fact, we can no longer argue $\mathcal{A}_2$ (it now contradicts our strict knowledge). Thus, since $\mathcal{A}_1$ was the only argument supporting $\sim b$, this conclusion is lost.

2.3 Warrant

The whole purpose of a DeLP program is obviously to establish what set of conclusions holds. This purpose is filled by the notion of *warrant*. Given certain program $\mathcal{P}$, a conclusion ϕ is said to be *warranted on the basis of* $\mathcal{P}$, noted $\mathcal{P} \mid\!\sim_{\mathcal{W}} \phi$, if it is supported by an non-defeated argument constructed from $\mathcal{P}$. It should be stressed that in order to establish whether an argument is non-defeated, we must take into account all its potential counter-arguments—one of them might defeat our initial argument. Since these counter-arguments are in fact arguments, there may exist defeaters for the defeaters, and so on, thus requiring a complete recursive analysis. The set of warranted conclusions of a program $\mathcal{P}$ represents its semantics.

In a like manner as before, the consequence operator for $\mid\!\sim_{\mathcal{W}}$ can be defined as $C_{\mathcal{W}}(\Phi) = \{\phi \mid \Phi \mid\!\sim_{\mathcal{W}} \phi\}$. The next proposition summarizes our results in regard with this inference relation.

Proposition 2.3 *The consequence operator $C_{\mathcal{W}}$ satisfies inclusion but fails idempotence, cut, and cautious monotonicity.*

[2] for simplicity sake, we shall adopt propositional defeasible rules, despite the fact that these rules represent general knowledge and should contain variables.

It is easy to see why $C_{\mathcal{W}}$ complies inclusion: just note that every fact in a given program is trivially supported by the empty argument, and that the empty argument cannot be defeated since it is entirely based on strict information. To show the failure of cut and cautious monotonicity we have found the following counter-examples:

Example 2.4 We can use the same counter-example as before (example 2.2) to show that $C_{\mathcal{W}}$ does not satisfy cut. Note that the argument $\mathcal{A}_1 = \{b \mathrel{-\!\!\prec} c;\ c \mathrel{-\!\!\prec} d\}$ for b also warrants it on the basis of $\mathcal{P}$. However, should b be added to $\mathcal{P}$ as a fact, then the new argument $\mathcal{A}_2 = \{a \mathrel{-\!\!\prec} b, d\}$ for a can be built, which now warrants a on the basis of the new program.

Example 2.5 Consider a DeLP program $\mathcal{P} = (\Pi, \Delta)$, where $\Pi = \{a\}$ and $\Delta = \{c \mathrel{-\!\!\prec} b;\ b \mathrel{-\!\!\prec} a;\ \sim c \mathrel{-\!\!\prec} a\}$. The arguments $\mathcal{A}_1 = \{c \mathrel{-\!\!\prec} b;\ b \mathrel{-\!\!\prec} a\}$, $\mathcal{A}_2 = \{\sim c \mathrel{-\!\!\prec} a\}$, and $\mathcal{A}_3 = \{b \mathrel{-\!\!\prec} a\}$, supporting respectively c, $\sim c$, and b, can be all built from $\mathcal{P}$. Yet, only $\sim c$ and b are warranted on the basis of $\mathcal{P}$, since $\mathcal{A}_2$ defeats $\mathcal{A}_1$, but both $\mathcal{A}_2$ and $\mathcal{A}_3$ remain undefeated. Finally, should b be added to $\mathcal{P}$ as a fact, the new argument $\mathcal{A}_4 = \{c \mathrel{-\!\!\prec} b\}$ for c now defeats $\mathcal{A}_2$. Thus, since $\mathcal{A}_2$ was the only argument for $\sim c$, this previously warranted conclusion is now lost.

3 Conclusions

In this work we have studied the different inference relations defined within defeasible logic programming with respect to a set of desirable properties for any nonmonotonic reasoner. The results we have obtained are summarized in the following table:

	Inclusion	Idempotence	Cut	Cautious M.
$\vdash$	$\surd$	$\surd$	$\surd$	$\surd$
$\mid\!\sim$	$\surd$	$\surd$	$\surd$	$\surd$
$\mid\!\sim_{\mathcal{A}}$	$\surd$	$\times$	$\times$	$\times$
$\mid\!\sim_{\mathcal{W}}$	$\surd$	$\times$	$\times$	$\times$

Note that the enhanced inference relations satisfy less properties than the simpler ones. This stems from the fact that each inference relation is included in the previous one, that is to say, for a given DeLP program, $C_{\vdash} \subseteq C_{\mid\sim} \subseteq C_{\mathcal{A}} \subseteq C_{\mathcal{W}}$. Based on these properties, the notion of warrant does not seem to improve argument-based inference. However, the consequence operator $C_{\mathcal{A}}$ fails to preserve consistency. That is to say, for a set of premises Φ non contradictory with respect to strict inference, $C_{\mathcal{A}}(\Phi)$ may become contradictory. On the contrary, $C_{\mathcal{W}}$ does preserve consistency (a fundamental property for any logical system).

It is worth remarking that DeLP semantics (*i.e.*, the set of warranted conclusions) does not verify cumulativity. As we mentioned before, to some authors DeLP would not be deemed as a logical system. Nevertheless, DeLP has been successfully applied to a number of challenging scenarios with encouraging results. In addition, several others nonmonotonic theories do not hold cumulativity as well. For instance, Pollock's Oscar [7], Prakken and Sartor's prioritized argumentation theory [8], or Vreeswijk's abstract argumentation framework [11]. These results show that cumulativity appears to be too restrictive for any sufficiently expressive entailment relation.

References

[1] Antoniu, G. *Nonmonotonic Reasoning.* MIT Press, 1996.

[2] Chesñevar, C. I. *Formalización de los Procesos de Argumentación Rebatible como Sistemas Deductivos Etiquetados.* PhD thesis, Departamento de Ciencias de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, Mar. 2001.

[3] Gabbay, D. Theoretical foundations for nonmonotonic reasoning in expert systems. In *Logics and Models of Concurrent Systems*, K. Apt, Ed. Springer-Verlag, Berlin, Germany, 1985.

[4] García, A. J. *Programación en Lógica Rebatible: Lenguaje, Semántica Operacional, y Paralelismo.* PhD thesis, Departamento de Ciencias de la Computación, Universidad Nacional del Sur, Bahía Blanca, Argentina, Dec. 2000.

[5] Kraus, S., Lehmann, D., and Magidor, M. Non-monotonic reasoning, preferential models and cumulative logics. *Artificial Intelligence 44* (1990), 167–207.

[6] Makinson, D. General patterns in nonmonotonic reasoning. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, H. Gabbay and Robinson, Eds., vol. 3. Oxford University Press, 1994, pp. 35–110.

[7] Pollock, J. L. *Cognitive Carpentry: A Blueprint for How to Build a Person.* The MIT Press, 1995.

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

[9] Simari, G. R., Chesñevar, C. I., and García, A. J. The Role of Dialectics in Defeasible Argumentation. In *Proceedings of the XIV Conferencia Internacional de la Sociedad Chilena para Ciencias de la Computación* (Concepción, Chile, Nov. 1994), Universidad de Concepción, pp. 111–121.

[10] Simari, G. R., and Loui, R. P. A mathematical treatment of defeasible reasoning and its implementation. *Artificial Intelligence 53*, 2–3 (1992), 125–157.

[11] Vreeswijk, G. A. W. Abstract Argumentation Systems. *Artificial Intelligence 90*, 1–2 (1997), 225–279.