A preliminary study of computational complexity in non-monotonic reasoning

M. Capobianco[1,*] C. I. Chesñevar[1,2] G. R. Simari[1]
mc@cs.uns.edu.ar cic@eup.udl.es grs@cs.uns.edu.ar

[1] Dpto. de Cs. e Ing. de la Computación – Universidad Nacional del Sur – Argentina
[2] Departament of Computer Science – Universitat de Lleida – España

Abstract

In this work we analyze existing complexity results in the area of non-monotonic reasoning in general and argumentation in particular. Even though the area of argumentation is based on solid theoretical foundations, its main problems rely on the computational complexity of the system that have so far been developed. In order to use argumentation in real time scenarios we must find an implementation with a reasonable response time. Complexity analysis of argument systems is an indispensable tool for addressing this taks.

We expect that the development of this research line will result in a general analysis of the issues in complexity of argument systems, leading to an efficient implementation of a particular formalism, *observation-based defeasible logic programming*, that could be integrated in an intelligent agent architecture.

1 Introduction

In the last thirty years, several ways to formalize defeasible reasoning have been studied. A particular approach, *defeasible argumentation* [6, 12], has been particularly successful to achieve this goal. Defeasible argumentation is built on the notions of arguments, counterarguments, attack and defeat. The inference process is based on the interaction of arguments for and against certain conclusions. This leads to a natural formalization of common sense reasoning that can be used in a broad set of applications.

The area of argumentation is based on solid theoretical foundations. Recently, argumentation systems have been integrated in prototypes of many real world applications, such as clustering algorithms [7], intelligent web search [5], modelling reasoning for multiagent systems in dynamic environments [11, 4], etc. Argumentation could also be used in the near future in several new applications, such as intelligent interfaces for human-machine communication, e-commerce agents and intelligent web browsing. Nevertheless, the main problem for the practical use of argumentation systems relies on its computational complexity. How can we find an implementation with a response time reasonable for real time scenarios?

Surprisingly, there has not been many useful results regarding the pragmatics of argument-based systems, and in particular, their inherent computational complexity. This may stem from the small amount of practical implementations based on the theoretical formalisms. Toni's research line on

*Partially supported by a postdoctoral fellowship of CONICET

complexity in assumption based argumentation [14] is an exception to this fact. We plan to address this area from a more general perspective, looking for a tractable implementation of an argumentative system. Such implementation will allow the integration of argumentation and *intelligent agents*.

We plan that the development of this research line also leads to an efficient implementation of a particular argumentative system, *observation-based defeasible logic programming* (ODeLP)[4]. The complexity of this implementation will be theoretically and empirically proven, and once a desirable result is obtained, the implemented system will be integrated into an intelligent agent architecture according to the model proposed in [4].

The rest of the paper is structured as follows. First, in Section 2 we present some basic elements of complexity theory. Next, Section 3 surveys related work on the field of complexity in nonmonotonic formalisms in general, focusing on argumentation in particular. Finally Section 4 outlines future work.

2 Background on complexity theory

Complexity theory for reasoning formalisms deals with decision problems, that is to say, problems that return a boolean answer. The decision problem for the class P is the set of problems that can be answered by a *deterministic Turing machine in polynomial* time. The class of problems that can be solved by a *non-deterministic Turing machine in polynomial* time is denoted NP. The class of problems whose answer is always the complement of those in NP is called co-NP. Problems in co-NP can also be solved by a non-deterministic Turing machine in polynomial time, but it is understood that the answer is *yes* if *all* the parallel computations done in the machine end in an accepting state. NP-complete problems are the toughest—with respect to many-one polynomial reducibility—problems in the class NP (or co-NP)[1].

Currently, the best algorithms known for solving NP-complete problems are exponential in the worst case. Event though none of the following relations have been actually proven, most experts conjecture that P $\subset$ NP, P $\subset$ co-NP and NP $\neq$ co-NP.

The complexity of most frameworks for non-monotonic reasoning is located at the lower end of the polynomial hierarchy To analyze these systems a particular type of computation is used, called *computation with oracles*. Intuitively, an oracle is a subroutine or function without cost. Consider a class of decision problems C, then P^C (NP^C) is the class of decision problems that can be solved in polynomial time by a deterministic (non-deterministic) Turing machine that uses an oracle for the problems in C, $i.e$, a function for solving the problems in C that can be called several times at unit cost.

Oracle computations are used to define the *polynomial hierarchy*. Intuitively, this is a presumably infinite hierarchy of complexity classes above NP, defined used *oracle machines*. Formally, the classes Σ_k^p, Π_k^p, Δ_k^p of the polynomial hierarchy are defined as:

$$\Sigma_0^p, \Pi_0^p, \Delta_0^p = P$$

and for all $k > 0$:

$$\Sigma_{k+1}^p = \mathrm{NP}^{\Sigma_k^p}, \Pi_{k+1}^p = \text{co-NP}^{\Sigma_k^p}, \Delta_{k+1}^p = \mathrm{P}^{\Sigma_k^p}$$

Note that all the problems in the polynomial hierarchy can be solved in polynomial time iff P=NP. Further, all these problems can be solved using worst time exponential algorithms. Nevertheless, different levels of the polynomial hierarchy may differ considerably in practice. Methods workings for moderately size instances of NP-complete problems may not work for Σ_2^p-complete problems.

[1] A good source for complexity theory is [2]

Formalism	Credulous reasoning	Sceptical reasoning
Logic programming	NP-complete	co-NP-complete
Theorist	Σ_2^p-complete	Π_2^p-complete
Circumscription	Σ_2^p-complete	Π_2^p-complete
Default logic	Σ_2^p-complete	Π_2^p-complete
Autoepistemic logic	Σ_2^p-complete	Π_2^p-complete

Figure 1: Existing complexity results for non-monotonic formalisms

As we shall see in the following section, many problems in propositional non-monotonic reasoning are Σ_2^p-complete or Π_2^p-complete and algorithms for solving these problems use satisfiability testers as subroutines.

3 Complexity in argumentation: related works

Early works on complexity for nonmonotonic logics and reasoning systems can be found in [3], a survey containing results for the most popular approaches. This work elaborates on the tractability/intractability of default logic, autoepistemic logic, circumscription, closed world reasoning and abduction. It states that notions of higher order complexity must be used in the analysis, since a general property of non-monotonic formalisms is that its computational complexity is higher than the complexity of the underlying monotonic logic. Part of the inherent complexity of these systems can be explained by the fact that their semantic definitions are either based on fix-point constructions or on some form of minimality conditions, and this seems to be a source of high complexity.

As an instance, circumscription (a minimality-based formalism) performs inference with respect to the models of a first order formula in which the extensions of some selected predicates are minimized. In this case the minimality condition gives raise to a a non-determinism form that cannot be reduced unless the polynomial hierarchy collapses.

The situation is no better for systems based on fix-point definitions (such as default or autoepistemic logics) where reasoning is performed by *extensions* or *expansions* that are solutions of fixpoint equations. This calculation requires a non-deterministic choice that is analogous to the case of minimality constructions.

The full results obtained for non-monotonic formalisms are summarized in Figure 3. Note that complexity results are analyzed in the case of credulous and sceptical reasoning.

The analysis of complexity for argument based systems lacks a general work such as Cadoli's survey. In fact, there are only a few scattered approaches to this field. The authors of most argumentative formalisms have not included a complexity analysis as part of their work. An exception in such sense is the work of Dimopoulos $et.al.$ [14] in which the authors study the computational complexity of an abstract framework for default reasoning based on argumentation. This work is based on the research line founded by Bondarenko $et.al.$ [1] where it is shown that many logics for default reasoning (autoepistemic logic [10], modal logic, logic programming[8], default logic[13], and many cases of circumscription[9]) can be understood as special cases of a single abstract framework based on argumentation. The standard semantics of all these logics can then be understood as sanctioning a set of assumptions as an *stable extension* of a given theory, formulated in the monotonic logic underlying the respective framework. A set of assumptions is said to be *stable* iff it does not attack this set and it attacks every other assumption that is not in the set. In abstract terms, an assumption is attacked when its contrary can be proven.

	Admissibility		Preferability	
Formalism	Credulous	Sceptical	Credulous	Sceptical
Logic programming	NP-comp	P-comp	NP-comp	Π_2^p-comp
Theorist	Σ_2^p-comp	co-NP-comp	Σ_2^p-comp	Π_2^p-comp
Circumscription	Σ_2^p-comp	co-NP-comp	Σ_2^p-comp	Π_2^p-comp
Default logic	Σ_2^p-comp	co-NP-comp	Σ_2^p-comp	Π_3^p-comp
Autoepistemic logic	Σ_3^p-comp	Π_3^p-comp	Σ_3^p-comp	Π_4^p-comp

Figure 2: Complexity results for the new (Bondarenko's) semantics

Bondarenko $et.al.$ also propose two new semantics generalizing the existing ones (that is, admissibility semantics and the semantics of preferred extensions for logic programming). In the new semantics, a set of assumptions is said to be an *admissible argument* iff it does not attack itself and it attacks all the set of assumptions that attack it. A set of assumptions is a *preferred argument* iff it is a maximal (with respect to set inclusion) admissible argument. The new semantics are more general than the stability semantics since every stable extension is a preferred (and of course admissible) argument, but not every preferred argument is an stable extension.

Even though the theoretic results regarding expressiveness of the new semantics are encouraging, it seems unlikely that they lead to better lower bounds than the standard semantics, given that all the sources of complexity are still present. To clarify this situation Dimopoulos $et.al.$ [14] present complexity results for the propositional variants of circumscription, logic programming, default logic and autoepistemic logic under the new semantics. To do this, they analyze the following problems:

- The credulous reasoning problem: the problem of deciding for any given sentence ϕ in the object language whether it belongs to the deductive closure of *some* assumption set sanctioned by the semantics.
- The sceptical reasoning problem: the problem of deciding for any given sentence ϕ in the object language whether it belongs to the deductive closure of *every* assumption set sanctioned by the semantics.

As a general result, they show that reasoning under the new semantics can be much harder than reasoning under the standard one. Figure 3 summarizes the results obtained for the admissibility and preferability semantics.

Note that for autoepistemic logic in particular, sceptical reasoning is situated (respectively) one and two level higher than under the stability semantics. This increase in complexity also holds for other frameworks, except for the case of admissibility semantics in default logic and logic programming (in the sceptical case) where complexity is one level simpler. Note that in this cases the reasoning process reduces to monotonic reasoning in underlying logic, resulting in a trivial case.

Summing up, the complexity results in Dimopoulo's work show that results under the new argument based semantics is harder. Nevertheless, we must also consider that the new semantics allows us to encode more complex reasoning patterns than under the traditional stability approach.

4 Conclusions and future work

Considering the analysis of reasoning systems and their inherent complexity we can affirm that non-monotonic logics are computationally hard. Argumentation semantics bring more flexibility and expressive power, at the expense of higher complexity. Therefore, studying the properties and complexity of existing formalisms is a key issue for obtaining practical applications. Devising proper restrictions and/or optimizations of argumentation seems to be mandatory in most cases.

As future work, we will analyze the system of observation-based defeasible logic programming, to find an adequate implementation that could be integrated into an agent architecture for intelligent agents. This approach builds on the preliminary design for this architecture that was originally presented in [4].

References

[1] Bondarenko, A. G., Dung, P. M., Kowalski, R. A., and Toni, F. An Abstract, Argumentation-Theoretic Approach to Default Reasoning. *Artificial Intelligence 93*, 1–2 (1997), 63–101.

[2] Bovet, D., and Crescenzi, P. *Introduction to the theory of complexity*. Prentice Hall, 1994.

[3] Cadoli, M., and Schaerf, M. A survey of complexity results for nonmonotonic logics. *Journal of Logic Programming 17* (1993), 127–160.

[4] Capobianco, M., Chesñevar, C. I., and Simari, G. R. An argument-based framework to model an agent's beliefs in a dynamic environment. *Lecture Notes in Artificial Intelligence 3366* (2005), 96–111.

[5] Chesñevar, C., and Maguitman, A. ArgueNet: An Argument-Based Recommender System for Solving Web Search Queries. In *Proc. of Intl. IEEE Conference on Intelligent Systems IS-2004. Varna, Bulgaria (to appear)* (June 2004).

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

[7] Gomez, S., and Chesñevar, C. A hybrid approach to pattern classification using neural networks and defeasible argumentation. In *Procs. of the International FLAIRS 2004 Conference, MIAMI, FL* (2004), pp. 388–393.

[8] Lloyd, J. W. *Foundations of Logic Programming*. Springer-Verlag, 1987.

[9] McCarthy, J. Circumscription – a form of non-monotonic reasoning. *Artificial Intelligence 13*, 1–2 (1980), 27–39.

[10] Moore, R. Possible-world semantics for autoepistemic logic. In *Proceedings of the Non-Monotonic Reasoning Workshop* (New Paltz, USA, 1984), American Association for Artificial Intelligence, pp. 344–354.

[11] Parsons, S., and Fox, J. Argumentation and decision making. In *Proceedings of the IEEE Coloquium on Decision Making and Problem Solving* (1997).

[12] Prakken, H., and Vreeswijk, G. Logical systems for defeasible argumentation. In *Handbook of Philosophical Logic*, D. Gabbay, Ed., vol. 4. Kluwer Academic Publisher, 2002, pp. 219–318.

[13] Reiter, R. A Logic for Default Reasoning. *Artificial Intelligence 13*, 1–2 (1980), 81–132.

[14] Yannis Dimopoulos, Bernhard Nebel, F. T. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence 141*, 1 (2002), 57–78.