On the Effect of Dynamic Environments in Defeasible Reasoning

Marcela Capobianco

Laboratorio de Investigación y Desarrollo en Inteligencia Artificial (LIDIA)
Departamento de Ciencias de la Computación
Universidad Nacional del Sur
e-mail: mc@cs.uns.edu.ar

1 Introduction

The design of *intelligent agents* has become a key issue for many interesting applications. Given that there is no universally accepted definition of intelligence, Russell developed the notion of *rational agency* as an alternative for the characterization of *intelligent agency* [11]. In short, an agent is said to be *rational* if it performs the *right* actions according to the information it possesses and the goals it wants to achieve.

Modeling the epistemic state of a rational agent is the most difficult enterprise that must be addressed in its design process. Pereira *et al*. have endorsed the use of *Extended Logic Programming* (ELP) for accomplishing this task [7]. In this setting, the knowledge of the agent is codified by a logic program extended with explicit negation, and its beliefs are set by the well-founded semantics of this program. This approach has a clear advantage: it admits a seamless transition between theory and practice. Nevertheless, ELP cannot deal with incomplete and potentially contradictory information, an essential capability in practical agents. Several *argumentation formalisms* [12,10,8,4,13] have been proposed as knowledge representation and reasoning tools able to handle uncertain information. This property elicits argument-based theories as proper tools for modeling the epistemic state of rational agents. Unfortunately, argumentative systems lack the implementability of ELP.

Defeasible Logic Programming (DeLP) [5] is a new paradigm that interweaves the benefits of logic programming and argument-based systems, thus providing a good trade-off between expressivity and implementability. As a result, DeLP presents attractive properties to model the knowledge of rational agents [1]. DeLP combines a language similar to that of logic programming with an inference process based on argumentation. To answer a query q according to a *defeasible logic program* P the system builds arguments for and against q. These arguments contend together in a dialectical dispute, in which the outcome is established using a *defeat criterion*. The query q can be inferred from P if there exists an argument for q that prevailed in the dispute.

Most applications for rational agents involve interacting with a dynamic world. To properly achieve this interaction, the agent must be continuously adapting to the changes in its environment. In this context, perception is a mandatory issue. We have tailored the DeLP system incorporating perceptions abilities into a new formalism called *Observation based DeLP* (ODeLP). The language of ODeLP is composed by a set of *observations* Ψ, encoding the knowledge the agent has about the world, and a set of *defeasible rules* Δ, representing ways of extending the observations with tentative information (i.e., information that can be used if nothing is posed against it). The ODeLP program P structuring the knowledge of the agent is able to express the following doxastic attitudes with respect a query q:

- Believe that q is true. In this case q belongs to the set of agent's beliefs.
- Believe that q is false, i.e., believe in the contrary of q.
- Neither believe that q is true nor that it is false.

Both the language and the inference process of ODeLP have been formally defined. We have also developed an optimization of the ODeLP system that uses precompiled knowledge to speed up the agent's reasoning. Since the cognitive process of rational agents is complex and computationally expensive, the proposed optimization helps the agent to achieve a timely interaction with its environment. This is essential for perceiving and acting on the world. In what follows we briefly describe how to incorporate perception mechanisms and precompiled knowledge into the ODeLP system.

2 Perception

John Pollock pioneered in the problem of perception in artificial agents. In *Taking Perception Seriously* [9], Pollock claims that an agent residing in a complex dynamic environment cannot be provided from its creation with all the information it needs. Therefore, the agent must be able to perceive.

In our system, the task of perceiving can be carried out by any mechanism that detects the changes in the world and reports the literals representing those changes. The specification of this mechanism depends on the particular application domain and it is not addressed in our work. We only assume that it cannot produce false observations. The perceived literals are added to the knowledge of the agent, into the set of observations Ψ.

If new facts are carelessly added, Ψ may become inconsistent. To avoid this we have defined an updating process [6] that removes any element of Ψ contradicting the new observation. Note that according to our criterion, new perceptions are always preferred over older ones. There is a simple reason behind this policy: given our initial assumption, both of the observations in disagreement were correct at the time of their assimilation. As a result, the only explanation for the conflict is a change in the state of world, and the new fact should be favored since it reflects the actual state. It is worth mentioning that by updating the set of observations the agent can modify its beliefs, changing its previous picture of the world when faced with new information.

3 Precompiled Knowledge

John Doyle defined the concept of *Truth Maintenance Systems* (TMS) [3] as tools for problems solvers, inaugurating the use of precompiled knowledge. The function of a TMS is to record and maintain the reasons for the agent's beliefs. Doyle describes a series of procedures that determine the current set of beliefs from the recorded information and update it in accord with new reasons. He remarked that the use of precompiled knowledge significantly improves the performance of the system.

> ... the overhead required to record justifications for every program belief might seem excessive. However, the pressing issue is not the expense of keeping records of the sources of beliefs. Rather, we must consider the expense of not keeping these records. If we throw away information about derivations, we may be condemning ourselves to continually re-deriving information in large searches caused by changing irrelevant assumptions ...

Doyle's proposal was a source of inspiration for our work: precompiled knowledge can also be used in ODeLP to prevent the system from building the same argument from scratch several times. The key idea can be resumed as follows. For a ODeLP program $P = \langle \Psi, \Delta \rangle$ we maintain a repository containing, among other things, every argument that could be built from rules in Δ. Using this repository, we have defined a new optimized version of the inference process. For space reasons we refer the interested reader to [2] for a complete description about the use of precompiled knowledge in ODeLP.

4 Concluding Remarks

We have defined an expressive formalism that can be used to describe complex domains. ODeLP provides mechanisms to update the knowledge of the agent from information acquired perceptually, adapting it to the changes in the world. We have also shown that the use of precompiled knowledge can improve the performance of ODeLP in the same way Truth Maintenance Systems optimize problem solvers. Currently, we are researching on two different topics: studying the formal properties of the ODeLP system and developing an efficient implementation of our formalism.

References

[1] **Capobianco, M., and Chesñevar, C. I.** Using Logic Programs to Model an Agent's Epistemic State. In *Proceedings of the 7th Workshop on Aspectos Teóricos de la Inteligencia Artificial (ATIA), 2nd Workshop of Investigadores en Ciencias de la Computación (WICC)* (La Plata, May 2000), Universidad Nacional de La Plata.

[2] **Capobianco, M., and Simari, G. R.** Defeasible Reasoning in Dynamic Domains. In *Proceedings of the 1st Workshop en Agentes y Sistemas Inteligentes (WASI), 6th Congreso Argentino de Ciencias de la Computación (CACIC)* (Ushuaia, Oct. 2000), Universidad Nacional de la Patagonia.

[3] **Doyle, J.** A Truth Maintenance System. *Artificial Intelligence 12*, 3 (1979), 231-272.

[4] **Dung, P. M.** On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning and Logic Programming and n-Person Games. *Artificial Intelligence 77*, 2 (1995), 321-357.

[5] **García, A. J.** *La Programación en Lógica Rebatible: Lenguaje, Semántica Operacional, y Paralelismo*. PhD thesis, Universidad Nacional del Sur, Bahía Blanca, Argentina, Dec. 2000.

[6] **Katsuno, H., and Mendelzon, A.** On the difference between updating a knowledge base and revising it. In *Belief Revision*, P.Gardenfors, Ed. Cambridge University Press, 1992, pp. 183-203.

[7] **Pereira, L. M., and Quaresma, P.** Modelling Agent Interaction in Logic Programming. In *Proceedings of the 11th International Conference on Applications of Prolog* (Sept. 1998), O. Barenstein, Ed.

[8] **Pollock, J. L.** *Cognitive Carpentry: a blueprint for how to build a person*. Bradford/MIT Press, 1995.

[9] **Pollock, J. L.** Taking Perception Seriously. In *Proceedings of the 1st International Conference on Autonomous Agents* (Feb. 1997), pp. 526-527.

[10] **Prakken, H., and Sartor, G.** Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics 7* (1997), 25-752.

[11] **Russell, S.** J. Rationality and Intelligence. *Artificial Intelligence 94*, 1-2 (1997), 57-77.

[12] **Simari, G. R., and Loui, R. P.** A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence 53*, 1-2 (1992), 125-157.

[13] **Vreeswijk, G.** Abstract Argumentation Systems. *Artificial Intelligence 90*, 1-2 (1997), 225-279.