Available online at www.sciencedirect.com
SCIENCE DIRECT®

Artificial Intelligence 165 (2005) 187–259

Artificial Intelligence

www.elsevier.com/locate/artint

Self-stabilizing defeat status computation: dealing with conflict management in multi-agent systems

Pietro Baroni, Massimiliano Giacomin *, Giovanni Guida

Dipartimento di Elettronica per l'Automazione, Università di Brescia, Via Branze 38, I-25123 Brescia, Italy

Received 5 September 2004; accepted 15 February 2005

Available online 27 April 2005

Abstract

The role of argumentation in supporting various forms of interaction among possibly conflicting autonomous agents has been explicitly recognized in the literature. In argumentation, conflict management is carried out by the formal process of defeat status computation. In this paper we consider the generalization of this process to a distributed setting. We show that significant stabilization problems may arise even in relatively simple cases. A fundamental negative result is then proved: no general self-stabilizing algorithm exists for distributed defeat status computation, indicating that self-stabilizing algorithms for this problem can be defined only under specific conditions. Accordingly, we focus on two cases: an algorithm tailored to a specific family of inference graphs, that include only rebutting defeaters, and an algorithm that applies to any inference graph, also including undercutting defeaters, but may provide (cautiously) incorrect results for some nodes. For both algorithms the worst-case round complexity is analyzed and it is proved that no algorithms with lower complexity exist for the same tasks.

Keywords: Argumentation; Self-stabilization; Multi-agent systems

* Corresponding author.
E-mail addresses: baroni@ing.unibs.it (P. Baroni), giacomin@ing.unibs.it (M. Giacomin), guida@ing.unibs.it (G. Guida).

0004-3702/$ – see front matter
doi:10.1016/j.artint.2005.02.003

1. Introduction

Conflicts arise in everyday life when distinct autonomous subjects disagree, for whatever reasons, about opinions, objectives, possession or use of goods, and many other possible matters of controversy. The ability to manage and (where possible) solve conflicts is in fact among the most important social skills. Similarly, conflicts play an important role in multi-agent systems, where autonomous software components, holding distinct beliefs and endowed with different resources, coexist and pursue individual goals. The existence of conflicts is strictly related to two basic conditions:

- the presence of some form of autonomy of the agents, for example in acquiring information, in making decisions or in executing plans of actions;
- the need for the agents to interact, for example to exchange knowledge or resources or to compete for them, in order to reach their goals.

Conflict resolution models and techniques are receiving an increasing attention in multi-agent literature. In particular, several works have considered the use of *argumentation theory* as a suitable tool to model interaction among autonomous agents [10,35,65].

In classical centralized argumentation systems, two main activities are involved: the production of new, possibly conflicting, arguments by means of some inferential activity (*argument construction*), and the identification of which conclusions are justified, given the developed arguments and their conflict relationships (*defeat status computation*). In a multi-agent setting, this schema needs to be extended to encompass the two basic conditions mentioned above:

- argument construction is distributed among agents, which autonomously produce arguments by means of some internal inferential activity;
- agents interact by exchanging information concerning the produced arguments.

In particular, alternative architectural decisions can be considered as far as information exchange is concerned. On the one hand, the exchanged arguments may be either managed by a centralized entity, which provides a shared store to all the participating agents, or communication may be carried out directly between agents on a peer-to-peer basis. On the other hand, agents may be bounded to always exchange entire arguments, including all the grounds supporting argument conclusion, or may be free to communicate selected information about arguments, e.g., conclusions or attack relationships. These structural features directly affect the way defeat status computation can be carried out: three main cases, corresponding to an increasing level of distribution, can be identified in this respect. First, in the presence of a centralized structure collecting all the exchanged arguments, defeat status can be computed by means of a centralized algorithm operating on them and updating the contents of the structure itself. As a consequence, defeat status computation can be carried out in an analogous way as in the case of conventional centralized argumentation systems. In the second case, no global sharing of arguments is assumed, but agents exchange directly complete arguments among them. Combining locally produced arguments with the ones received from others, each agent achieves a partial view of the whole set