A simulator for organisation-centred MAS adaptation in P2P sharing networks

Jordi Campos and Maite Lopez-Sanchez
Universitat de Barcelona
Gran Via 585
08007 Barcelona, Spain
{jcampos, maite}@maia.ub.es

Marc Esteva and Javier Morales
IIIA, CSIC
Campus de la UAB
E-08193 Bellaterra, Spain
{marc, jmorales}@iiia.csic.es

ABSTRACT
We present a simulator to compare different approaches to organisation-centred MAS adaptation in a peer-to-peer (P2P) scenario. In particular, we describe our approach to MAS adaptation (2-LAMA), the P2P sharing network case study and the software we built to evaluate different alternatives.

Categories and Subject Descriptors
1.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence—Multiagent systems, Coherence and coordination

General Terms
Design, Experimentation, Performance

Keywords
Assistance, Coordination, Organisation, P2P, MAS

1. INTRODUCTION
Organisational centred multi-agent systems (OCMAS) have proved to be effective to regulate agents’ activities. Nevertheless, population and/or environmental changes may lead to a poor fulfilment of the system’s purposes, and therefore, adapting the whole organisation becomes key. In order to endow an OCMAS with self-adaptation capabilities, we propose to incorporate a meta-level in charge of adapting system’s organisation. Hence, we call our approach Two Level Assisted MAS Architecture (2-LAMA) [2]. As a case study, we apply this approach to a P2P sharing network scenario. Moreover, we built a simulator1 to analyse and compare different adaptation alternatives in such a scenario. Thus it can be used as a testbed for comparing them. The simulator also provides different visual tools to analyse the behaviour of the system and its adaptation. In the following sections we present our MAS adaptation approach, the P2P case study and the simulator we built.

1Video: http://namaste.maia.ub.es/2LAMA/video.ogv


Copyright © 2010, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

Figure 1: Two Level Assisted MAS Architecture (2-LAMA): Domain Level (DL), Meta Level (ML) and Interface.

2. 2-LAMA MODEL
Organisational entities are used by some MAS to regulate their participants’ coordination [1]. We describe these entities as \( \text{Org} = \langle \text{SocStr}, \text{SocConv}, \text{Goals} \rangle \), where \( \text{SocStr} \) stands for a social structure (roles and their relationships), \( \text{SocConv} \) stands for social conventions (that agents should conform and expect others to conform, i.e. protocols and norms) and \( \text{Goals} \) stands for the organisational design’s purpose. In order to improve system’s performance under varying circumstances, we suggest to add a meta-level in charge of updating its organisation. To provide such an adaptation, we propose goal fulfilment as its driving force within the context of a rational world assumption. Specifically, we propose a Two Level Assisted MAS Architecture (2-LAMA) [2]. It consists on a distributed meta-level (ML) that provides assistance to a domain-level (DL) in charge of domain-specific tasks. Figure 1 shows them and their communication through an interface (Int). Thus, the whole system can be denoted as \( 2LAMA = \langle ML, DL, Int \rangle \) —it is possible to nest subsequent meta-levels by updating previous level’s organisation. Each level has a set of agents with its own organisation: \( DL = \langle AgDL, OrgDL \rangle \) and \( ML = \langle AgML, OrgML \rangle \). Using the interface, ML agents perceive partial information—in many scenarios global information is not available—about environmental observable properties (e.g. date or temperature) and agents’ observable properties (e.g. colour or position). In particular, a ML agent has partial information about the subset of DL agents it assists. We assume DL agents are grouped into clusters according to a domain-specific criterion—e.g. interaction costs. Therefore, a ML agent—we call it assistant—assists a cluster of DL agents, observes partial information about them, and shares it with
what is happening in the simulation at every moment. The Main layout (5) shows the elements of the simulation and the communications among them. Peers and assistants are drawn according to the network topology, while messages are displayed as arrows among them with the corresponding colour defined in the legend panel. Finally, the Resume layout (6) displays how the data has been distributed among the different peers. It highlights completed peers and displays arrows connecting source and receiver agents. These arrows are labelled with the time step at which the datum was received. In addition, the simulator generates log files containing all occurred events during executions. It includes a module for facilitating the analysis of simulation results.

For this purpose, this module processes the generated logs extracting relevant information, which is later on displayed in different types of graphics. Hence, this can be used to compare the time spent to share the data in different configurations, or using different sharing methods.

The Agent Simulator component represents the conceptual model defined by the 2-LAMA targeted to drive the simulation at agent-level. Among others, it provides facilities to create state-based agents, and to define a problem (number of peers, who has initially the datum, etc.). Finally, the Network Simulator component drives the simulation at network-level, simulating message transport as a packet switching network. It provides facilities to define different network topologies, and to collect statistical information about network status.

Current implementation offers different alternate adaptation mechanisms that can be executed over the same configurations to compare their results. In particular, it includes two 2-LAMA alternatives: one where assistants use an heuristic to decide how to adapt the system, and one where assistants use machine learning (Case Based Reasoning, CBR) to take such decisions. In addition, it also offers an implementation of BitTorrent as a standard P2P protocol reference. Notice, that simulator’s components are easily extensible to implement other adaptation mechanisms or P2P protocols to compare them with current ones.

5. FUTURE WORK

As future work, we plan to extend the simulator with peers that enter or leave at any moment and violate norms. Even more, we plan to let user control a peer agent in order to increase simulator’s interactivity capacity. These features will let us test open MAS issues on our 2-LAMA approach.

Acknowledgements: This work is partially funded by IEA (TIN2006-15662-C02-01), EVE (TIN2009-14702-C02-01 / TIN2009-14702-C02-02) and AT (CONSOLIDER CSD2007-0022) projects, EU-FEDER funds, the Catalan Gov. (Grant 2005-SGR-00093) and Marc Esteva’s Ramon y Cajal contract.

6. REFERENCES
