Assistance Layer in a P2P Scenario

Jordi Campos[1] and Maite López-Sánchez[1] and Marc Esteva[2]

[1] MAiA Dept., Universitat de Barcelona, {jcampos, maite}@maia.ub.es
[2] Artificial Intelligence Research Institute (IIIA) CSIC, marc@iiia.csic.es

Abstract. Usually, MAS design and implementation involves a coordination model that structures agent interactions and an infrastructure in charge of enacting it. We propose the term Coordination Support to denote the services offered by this infrastructure. Such services can be grouped in different layers. We propose an additional Assistance layer devoted to assist coordination rather than just to enable it. This layer is illustrated by means of a Peer-to-Peer sharing network (P2P) scenario, so that the benefits of our proposal can be empirically evaluated.

1 Introduction

As a general illustration of the proposed *Coordination Support* concept[1], with an accent on the *Assistance Layer*, we present a P2P sharing network scenario. In this scenario, a set of computers connected to the Internet (peers) share some data. The performance is evaluated in terms of time and network consumptions during the sharing process. We model this scenario as a MAS where peers are agents with a certain organisation. Its role capabilities and their net of relationships constitute its *social structure*. Also, we assume they are organised to pursue the following goal: all *peers* get the data by consuming minimum time and network. Moreover, they follow some social conventions. Specifically, peers use a simplified version of the BitTorrent protocol[2] to interact and a norm to limit their network usage. This norm can be expressed as: $norBW_{DL}$ ="a peer cannot use more than $\mathtt{max}_{\mathtt{BW}}$ bandwidth to share data".

2 Assistance Layer

To provide assistance functionalities, we have implemented our proposed 2-LAMA architecture [3], which places a *meta-level* (ML) on top of a previously existing *domain-level* (DL). Accordingly, the P2P MAS constitutes the *domain-level* whilst the *meta-level* consists of a set of agents we call *assistants* organised to aid peers. Each *assistant* provides a subset of peers (cluster) with the following functionalities:

Information. It consists on providing agents with necessary and useful information to participate: the description of current norms and the identifiers of those other peers interested in the same data. An *assistant* supplies this information to new peers joining its cluster and to all peers in its *cluster* whenever this information is updated. Norm descriptions are sent by means of "norm <norm_id> <definition>" messages, whereas information about peers are sent with "interested <peer> [,<peer>]*" messages.

Justification. It justifies the effects of agent actions due to social conventions: involves providing explanations about why some messages have been filtered out due to the application of a norm. We assume Internet Service Providers (ISP) are equipped with an infrastructure mechanism that filters out messages that violate norms. This approach is not unrealistic since, nowadays, there exist ISP initiatives to improve P2P distribution systems. Specifically, *assistants* supply justifications to their peers by sending a "justification '<orig_message>' action 'filtered out' reason 'norm <norm_id>'" message.

Advices. It consists in suggesting plans to contact other *peers*. Based on network communication times, *assistants* recommend the subset of agents to be contacted by a newcomer *peer*. Each *assistant* generates these plans based on partial information it receives from its *clusters*. This information consists of communication times and data possession. At *meta-level*, *assistants* communicate among them to share a summary of this information. Consequently, each *assistant* has detailed information about its *cluster* and an overview about the rest of the *domain-level*. *Assistants* use this information to estimate which are the shortest paths among data sources and destinations. Accordingly, they recommend *peers* to contact those other *peers* that are in these shortest paths [3]. by sending "contact <peer> [,<peer>]*" messages.

Adaptation. In updates system's organisation to improve its design purpose achievement. Our *meta-level* adapts the *domain-level* organisation by changing its bandwidth norm ($norBW_{DL}$). It changes the bandwidth limit ($\mathtt{max}_{\mathtt{BW}}$) at certain time intervals. As this limit increases, the time to share data decreases since more network is used to transmit it. However, a large increase of network usage can saturate it and, as a result, increase time instead of reducing it. *Assistant* agents observe its *cluster* network usage and suggest to vary this limit in order to increase the usage without achieving network saturation (see [3] for further details). Then, *assistants* need to agree on the next value of $\mathtt{max}_{\mathtt{BW}}$. In current implementation, each *assistant* computes the average of all suggested $\mathtt{max}_{\mathtt{BW}}$ and communicates it to its *peers*.

3 Implementation and Results

We have implemented a prototype [3] of our P2P proposal in Repast Symphony [4]. Our actual P2P scenario is composed of 12 *peers* grouped in 3 different *clusters*, each one having its own assistant. Figure 1 depicts the underlying network topology composed by peers (p), assistants (a), routers (r) and links among elements (lines with an associated bandwidth).

We tested three MAS approaches: Brute-force-no-ML, a MAS without ML in which all *peers* contact all the others to request the datum; 2-LAMA-AgAss, our proposed architecture providing *agent assistance* through information, justification, and advice services but without norms; and 2-LAMA-AgAss-OrgAss, previous approach plus an *organisational assistance* to adapt the $norBW_{DL}$ norm. We

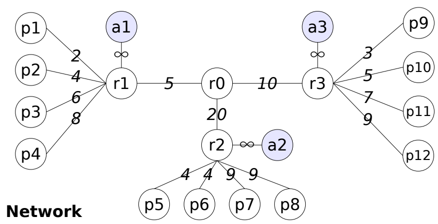

Fig. 1. P2P network topology.

Table 1. Results using different approaches.

MAS approaches	c_t	c_n
`Brute-force-no-ML`	1325.97	69884.3
`2-LAMA-AgAss`	1146.38	25089.2
`2-LAMA-AgAss-OrgAss`	876.43	18854.5

use a simplified version of BitTorrent protocol in each one of them. Table 1 shows the time (c_t) and network (c_n) consumed by these approaches. c_t is the number of time units required to distribute the datum among all *peers*, whereas c_n is the number of network units consumed during this same period (the network units consumed by a single message depends on its length and its transportation latency). Obtained results show that adding a *meta-level* (`2-LAMA-AgAss`) saves network consumption, since it advises *peers* to contact only a subset of them. This reduction also means there is less network saturation, so messages travel faster and c_t is slightly decreased. Moreover, if we add the bandwidth limit norm and let the ML adapt it depending on network status (`2-LAMA-AgAss-OrgAss`) then, the savings become significantly greater for both measures. Over all, simulations show positive results of social structure advices and norm adaptation.

4 Conclusions

This paper suggests to include an additional Assistance layer when developing MAS. We illustrate this new layer in a P2P sharing network scenario. The results show the *Assistance Layer* improves system's performance, especially when adapting MAS' organisation. In fact, we regard run-time assistance as a new research line in MAS. Assisting agents about current coordination model can simplify its development and let them be more effective and efficient. On the other hand, adapting a MAS organisation to varying circumstances can help to keep its original purpose.

Acknowledgements. Work funded by IEA (TIN2006-15662-C02-01), AT (CONSOLIDER CSD2007-0022), and Marc Esteva's Ramon y Cajal contract.

References

1. Campos, J., López-Sánchez, M., Esteva, M.: Assistance layer, a step forward in Multi-Agent Systems Coordination Support. In: AAMAS'09. (2009) 1301–1302
2. Cohen, B.: The BitTorrent Protocol Specification. http://www.bittorrent.org
3. Campos, J., López-Sánchez, M., Esteva, M.: Multi-Agent System adaptation in a Peer-to-Peer scenario. In: ACM SAC09 - Agreement Technologies. (2009) 735–739
4. North, M., Howe, T., Collier, N., Vos, J.: Repast Simphony Runtime System. In: Agent Conference on Generative Social Processes, Models, and Mechanisms. (2005)