Artificial Intelligence Research and Development
R. Alquézar et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-643-0-49

Comparison of Topologies in Peer-to-Peer Data Sharing Networks

Jordi CAMPOS [a,1], Nuria PIQUÉ [a], Maite LOPEZ-SANCHEZ [a], Marc ESTEVA [b]
[a] *WAI, MAiA, Universitat de Barcelona, e-mail: {jcampos, maite}@maia.ub.es*
[b] *Artificial Intelligence Research Institute (IIIA) CSIC, e-mail: marc@iiia.csic.es*

Abstract. Interactions within Multi-agent systems can be structured in different ways depending on the application scenario and its environmental restrictions. In previous work we have developed a multi-agent system within a Peer-to-Peer (P2P) sharing data network scenario. Its design includes an organisation that structures agents' interactions and an abstract architecture (2-LAMA) that helps to improve its performance. The focus of this paper is to first characterise the environment of this scenario in terms of network topologies and, secondly, to study how it affects system's performance in relation to the proposed architecture. In order to do that, we have set up a series of experiments that consider different network topologies and evaluate their performance. Results show that our architecture effectively helps to improve system's performance in despite of physical network topology.

Keywords. MAS organizations, Network topologies, MAS adaptation, Performance

Introduction

Multi-agent systems can be defined as systems where a set of autonomous software entities (namely agents) interact among them within an environment. Interactions within Multi-agent systems can be structured in different ways depending on the application scenario and its environmental restrictions. When taking an organisation centred perspective, it is necessary to consider the application scenario in the very early stages of the organisation design process, since it will drive the whole modelisation. This is so because main components in the organisation (such as enacted roles or protocols to follow) are highly dependent on both the considered domain and the overall design objective (that is, the general purpose for which the multi-agent system has been designed). Additionally, the application environment may also pose restrictions over agent capabilities and interactions. Therefore, when studying the performance of a multi-agent system, it becomes crucial to consider those environmental restrictions.

In previous work [3] we have proposed a multi-agent system designed to share data in a Peer-to-Peer (P2P) network scenario where agents correspond to peer computers. As next sections describe, its design includes an organisation that structures agents' interactions and an abstract architecture that helps to improve its performance. Nevertheless,

[1]Corresponding Author: Jordi Campos, Dep. Matemàtica Aplicada i Anàlisi, Facultat de Matemàtiques, Universitat de Barcelona, Gran Via de les Corts Catalanes 585, 08007 Barcelona; E-mail: jcampos@maia.ub.es.

as we have previously stated, environmental conditions may restrict this performance, and thus, we need to first characterise the environment in the P2P sharing data network scenario and, secondly, to study how it affects system's performance. In order to do that we have set up a series of experiments that consider different network topologies and evaluate and compare their performance.

In our P2P scenario, network topologies represent the physical Internet connections among peer computers (i.e., agents). These links in the topology can pose constraints on different features such as the transmission capacities or the cost associated to message sending. Moreover, if the number of messages sent trough a link surpass its transmission capacities, it will introduce a delay on these transmissions. The worst problem is encountered when links become saturated, since delays increase abruptly. Our proposed abstract architecture (2-LAMA) tries to minimise these problems by providing assistance services to participant agents. More specifically, within this 2-LAMA architecture, some *meta-level* agents assist *domain-level* agents in charge of sharing the data. Therefore, the aim of this paper is to define different network topologies and to compare their performances so to asses if the *meta-level* agents are able to successfully assist *domain-level* agents when performing their activity.

1. Proposed 2-LAMA Abstract Architecture

Organisational entities are used by some MAS to regulate their participants' coordination. We regard these entities as a *Coordination Support* to agents [2]. Moreover, we propose to assist coordination further than just enabling it. In particular, we suggest to add a new layer (*Assistance layer*) that provides assistance services to participants. This layer provides two main types of services: assisting individual agents to pursue their goals taking into account current organisational regulations (Agent assistance); and adapting those regulations to varying circumstances (Organisational assistance). In this paper, all experiments use the latter, as a distributed pro-active service that adapts an organisation –i.e. P2P sharing regulations– depending on varying circumstances —i.e. network saturations. We define such an organisation as $Org = \langle SocStr, SocConv, Goals\rangle$, where $SocStr$ stands for a social structure (roles and their relationships), $SocConv$ stands for social conventions (that agents should conform and expect others to conform, i.e. protocols and norms) and $Goals$ stands for the organisation design's purpose.

In order to provide these services, we proposed a Two Level Assisted MAS Architecture (2-LAMA [3]). It consists on a distributed *meta-level* (ML) that provides assistance to the part of the system in charge of the domain tasks (i.e. the *domain-level,* DL). It also has an interface (Int) that communicates both levels. Thus, the whole system can be expressed as: $2LAMA = \langle ML, DL, Int\rangle$ —in fact, it is possible to nest subsequent *meta-levels* that update previous level's organisation. Each level has a set of agents (Ag_{ML} and Ag_{DL}) and so they are respectively defined as $ML = \langle Ag_{ML}, Org_{ML}\rangle$ and $DL = \langle Ag_{DL}, Org_{DL}\rangle$. By using the interface, the meta-level can perceive environment observable properties ($EnvP$, e.g. date or temperature) and agents observable properties (AgP, e.g. colour or position). Specifically, we assume each meta-level agent has partial information about them, so it only perceives $EnvP$ and AgP of a subset of the domain-level —in fact, in many scenarios global information is not available. However, these meta-level agents can share part of this information in order to provide better assistance services.

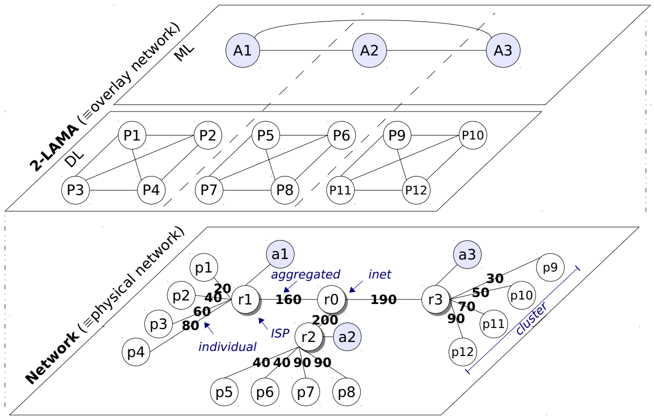


Figure 1. The relationships among agents (P_i, A_i) conform an *overlay network* on top the network links (edges) among terminations (p_i, a_i) and routers (r_i). A *cluster* is a set of peers connected to the same router.

2. P2P Scenario

Our case study is a Peer-to-Peer sharing network (P2P), where a set of computers connected to the Internet (*peers*) share some data. It is worth to apply our model to this scenario because it is a highly dynamic environment due to the very nature of the Internet communications. We regard the overlay network of current contacted peers as its dynamic organisational social structure —see Figure 1. In addition, the performance is evaluated in terms of time consumption during the sharing process: the faster the data is obtained, the better for the users. Notice that a peer usually contacts just a subset of other peers in order to avoid saturating its own communication channel. In fact, if network is saturated, messages are delayed and the time required to share data increases. Thus, there is a trade-off between time and network usage.

2.1. MAS layer

Following our 2-LAMA architecture, we model the P2P scenario as a MAS with two level of organised agents —see top part of Figure 1. We model peers as agents within the domain-level ($Ag_{DL} = \{P_1..P_n\}$). Its single role ($peer$) and the relationships among them conform the social structure of their organisation (Org_{DL}). This organisation also has its own social conventions, like a protocol and some norms. The former is a simplification of the standard BitTorrent protocol as described in [3]. The latter consists in two norms. First norm limits agents' network usage in percentage of its nominal bandwidth[2]. This norm can be expressed as: $normBW_{DL}$ =“a peer cannot use more than $\mathtt{max_{BW}}$ bandwidth percentage to share data”. This way, it prevents peers from massively using their bandwidth to send/receive data to/from all other peers. Second norm limits the number of peers to whom a peer can simultaneously send the data. Analogously to previous norm, we define $normFR_{DL}$ =“a peer cannot simultaneously send the data to more than $\mathtt{max_{FR}}$ peers”. In this paper's experiments, these norms have an initial value for their

[2]The *bandwidth* is the number of data units that can traverse a communication channel in a time unit. The less is used by the peer, the more is left for other purposes.

parameters ($\mathtt{max_{BW}} = 100$, $\mathtt{max_{FR}} = 5$), which is adapted during execution depending on network status as, mentioned below.

On top of the described domain-level, we add an Assistance layer to provide assistance to peers. This layer is implemented as a meta-level of agents ($Ag_{ML} = \{A_1..A_m\}$) that play the assistant role ($assistant$). Each assistant is in charge of a disjoint subset of peers (cluster $\subset Ag_{DL}$). Its decisions are based on local information about its cluster and summarised information about other clusters provided by other assistants —we assume assistants are located at ISPs and thus related communications are fast. Both levels share the same goal: that all peers obtain the data by consuming minimum time.

Throuhg the interface (Int) among both levels, assistants collect information about its agents (AgP) and their environment ($EnvP$). On the one hand, assistants collect information about communication latencies among agents and about who has the datum in order to suggest fast contacts to each peer —i.e. social structure adaptation [3]. On the other hand, assistants also collect some measures about the peers serving the datum and the ones that lack it. For instance, measures about how many peers lack the datum or about the bandwidth of peers having it. They use this information in order to adapt domain-level's norms at certain intervals —in current experiments, they are fixed to 50 ticks. They use a heuristic consisting in aligning the amount of served/received data —see norm adaptation in [3]. Finally, meta-level organisation has also a norm. Thus, when an assistant receives the information that one agent in another cluster has become complete, the number of domain-level agents in its cluster it can inform to is limited. This norm is defined as $normHas_{ML}$ =“upon reception of a complete agent (agent $\notin$ cluster) message, inform no more than $\mathtt{max_{Has}}$ agents $\in$ cluster”. In this paper, this norm is fixed for all experiments having the following parameter value $\mathtt{max_{Has}} = 1$.

2.2. Network layer

In our experiments, we use a packet switching network model to simulate the transport of messages among agents. As an illustration, the bottom part of Figure 1 shows one of the network topologies we use. Each peer's network adaptor (p_i) has an *individual* link to its corresponding Internet Service Provider (ISP: $r_{i>0}$). In addition, each ISP has an *aggregated* link among each group of peers –i.e. a *cluster*– and the Internet ($r0$). Links have independent upload and download channels whose bandwidth[2]are considered to be equal, and so they are represented as single numerical labels. Notice that the time required to transmit a message from one peer to another is highly dynamic since it depends on: its length, the bandwidths of the traversed links, and the number of simultaneous messages traversing the same links —a link's bandwidth is divided among the messages that traverse it simultaneously. Regarding message lengths, in this paper we have used the following ones: in general all messages have a single data unit, except for data messages that have 5000 data units and latency measures that have 150 data units. About the traversed links, they depend on network topology. For instance, in the star topology shown in Figure 1, there is only one path from a network termination to another one. On the contrary, in the ring topology shown in Figure 2-0b, there are two alternative paths among network terminations from different clusters. In such a case, the routing algorithm takes into account the number of links traversed and their bandwidth to decide which is the fastest route[3].

[3]Although there are several routing algorithms, their analysis is out of the scope of our research.

3. Experiment design

Our hypothesis is that our architecture is robust when changing the underlying network's topology. In other words, that 2-LAMA is still able to adapt MAS' organisation when agents communicate over different network's topologies. For instance, when participants communicate over a different network layout, when they have a different communication capacity (e.g. when there is a different combination of link's bandwidths), or when their number differs (e.g. when the number of domain-level agents grows). When designing the experiments to test such a hypothesis in current application domain (i.e. P2P data sharing), there are different alternatives to change any of these topology features. For instance, we can increase the number of peers while preserving or changing the number of clusters, or we can add the same number or a different number of peers to each cluster —i.e. there are different alternatives to change the number of participants. Consequently, we have designed seven groups of experiments that contain alternative topologies in order to study the performance of our approach when changing distinct topological features:

- *Group 0*: Basic topology layouts (ring and star) to be compared among them and with subsequent group variations —for example, the bottom part of Figure 1 shows a star layout for aggregated channels (router-router links), whereas Figure 2-0b shows a ring layout for them.
- *Group 1*: Larger bandwidth variations at individual channels (peer-router links, shown as "ind" in tables) as well as for aggregated channels (shown as "agg").
- *Group 2*: Homogeneous bandwidth both at individual and aggregated channels. Assigned values correspond to the mean of bandwidth values in group 0.
- *Group 3*: Increase in the number of peers without increasing the number of clusters. This leads to larger clusters.
- *Group 4*: Bring the number of clusters to its limit, i.e. one cluster per peer.
- *Group 5*: Increase the number of clusters in an homogeneous way, so that all clusters have three peers.
- *Group 6*: Unbalance the distribution of peers among clusters —see Figure 2-6a.

Each group is composed by different topologies (e.g. $a..d$) in which the corresponding features changes along them. Table 1 shows the resulting 32 different topologies in terms of: number of peers in the system (Peers); number of clusters grouping these peers (Clusters); how peers have being distributed among clusters (Distribution), for homogeneous distributions we specify how many peers has each cluster (thus, for example, "homogeneous (all 4)" in experiment 0a means that all 3 clusters have 4 peers each, whereas "heterogeneous (2,5,5)" in 6a means that first cluster has 2 peers and subsequent clusters have 5 peers each); network topologies where routers are connected following a ring or a star pattern (Layout); and links' bandwidth (BW), which can be homogeneous or heterogeneous for individual/aggregated channels. In our topologies, individual channels' bandwidth range $[10..90]$ whereas aggregated ones range $[50..200]$. Additionally, the value "+heterogeneous (ind)" means that differences have been defined larger at the level of peers, in the sense that some peers have bandwidth of 10 and others of 200 —the same applies for "+heterogeneous (agg)", with differences between 10 and 250.

Table 1. Experimental settings.

Group	Peers	Clusters	Distribution	Layout	BW
0a	12	3	homogenous (all 4)	star	heterogenous (ind & agg)
0b	=	=	=	ring	=
1a	12	3	homogenous (all 4)	star	+heterogenous (ind)
1b	=	=	=	=	+heterogenous (agg)
1c	=	=	=	ring	+heterogenous (ind)
1d	=	=	=	=	+heterogenous (agg)
2a	12	3	homogenous (all 4)	star	homogenous (ind)
2b	=	=	=	=	homogenous (agg)
2c	=	=	=	ring	homogenous (ind)
2d	=	=	=	=	homogenous (agg)
3a	30	3	homogenous (all 10)	star	heterogenous (ind & agg)
3b	=	=	=	ring	=
3c	120	3	homogenous (all 40)	star	=
3d	=	=	=	ring	=
4a	12	12	homogenous (all 1)	star	heterogenous (ind & agg)
4b	=	=	=	ring	=
4c	30	30	=	star	=
4d	=	=	=	ring	=
4e	120	120	=	star	=
4f	=	=	=	ring	=
5a	12	4	homogenous (all 3)	star	heterogenous (ind & agg)
5b	=	=	=	ring	=
5c	30	10	=	star	=
5d	=	=	=	ring	=
5e	120	40	=	star	=
5f	=	=	=	ring	=
6a	12	3	heterogeneous (2,5,5)	star	heterogenous (ind & agg)
6b	=	=	=	ring	=
6c	30	3	heterogeneous (6,12,12)	star	=
6d	=	=	=	ring	=
6e	120	3	heterogeneous (80,20,20)	star	=
6f	=	=	=	ring	=

4. Results

In order to test our 2-LAMA approach over different network topologies, we have implemented a P2P MAS simulator.This simulator is implemented in Repast Simphonyand provides different facilities to execute tests and analyse results. As it simulates both agent and network components, it allows to fairly execute our approach over different networks with identical initial conditions. In particular, we have tested all network topologies described in previous section by varying the agent that initially has the datum. Thus, the results for each network topology correspond to an average of several executions: one for each peer having initially the datum in the given topology. Table 2 shows these averages in terms of the following metrics: *timeCost* corresponds to the total time required to spread the datum among all peers; *netCost* is the network consumed by all messages (each message cost is computed as its length times the number of links it traverses); *netHop* specifies the average number of links (hops) that each message traverses;

netSat represents the overall links' network saturation average (we define saturation as the amount of data waiting to be transmitted over the data that the channel can actually transmit); and, finally, *netCol* is the average of network collisions in all links (a collision occurs when two packets from different messages are transmitted at the same link).

As for rows in Table 2, they correspond to previous topologies in Table 1. A brief analysis on 2-LAMA's performance when having these different physical network topologies can be summarized as follows:

- *Group 0* [basic topologies: ring and star layouts]: Ring (see 0b in Figure 2) performs better than star (see bottom part in Figure 1) both in time and network cost due to less link traversals (netHop). Since ring cost is smaller than the cost obtained for the star topology, we can express that 0b<0a.
- *Group 1* [larger BW variations at individual and aggregated channels]: Although ring layout still performs better than star layout, both perform worse than group 0. Therefore, we can conclude that the performance is very sensitive to unbalanced BWs, and this is specially the case for unbalanced BWs in the individual channels —notice, though, that ring topology is less affected by unbalances in the aggregated channels because messages are routed through alternate paths.
- *Group 2* [homogeneous BW at individual and aggregated channels]: They perform much better that group 1 and very similar to group 0, so we can conclude that homogeneity is desirable. This is especially the case for aggregated BW, since star topology outperforms group 0 (2b<0a). Again, ring performs better, although it takes less profit from homogeneous BW values at aggregated links.
- *Group 3* [more #peers per cluster]: When increasing the number of peers, more time is required to distribute the data among all peers, but the increase is moderated (below linear), so we can conclude that both topologies scale well, although as before, ring outperforms star both in terms of time and network consumption.
- *Group 4* [maximum #clusters: one per peer]: When increasing the number of clusters, the performance changes drastically. On the one hand, star topology improves its reference tests with 12, 30 and 120 peers (that is 0a, 3a and 3c respectively) and scales seamlessly (it takes around 6 times more time when having 10 times more peers) since the number of hops remains almost constant and below 4. On the other hand, ring topology performs much worse than its reference (0b, 3b and 3d). This is due to the fact that having so many clusters, messages have to traverse a large number of links before getting into their destination (in mean, it takes almost 30 times more when having 10 times more peers). This effect is reflected in the number of traversed hops, which also increases drastically. Overall, although star topology improves, it still performs worse than ring for those networks having a moderated number of peers (i.e., 0b with 12 and 3b with 30). It is for large networks (120 peers) where star outperforms any combination of clusters in ring topologies (4e<4f<3d).
- *Group 5* [increase #clusters and their #peers]: An homogeneous increase in the number of clusters (so that all clusters have 3 peers) behaves similarly to previous group 4. Thus, star still improves further than reference and 3 and 4 groups (0a>4a>5a, 3a>4c>5c and 3c>4e>5e). Ring also gets worse than the reference, but the effect here is more moderated so that it improves group 4 (4b>5b>0b, 4d>5d>3b and 4f»5f>3d). So, we can conclude that cluster topologies benefit from populated clusters whereas star topologies perform best for specific distri-

butions (since 3 peers per cluster show better results than 1 peer or 4 peers per cluster).

- *Group 6* [unbalanced #peers per cluster]: The effect of unbalancing the number of peers relates to the importance of the link BW studied in first experiments (i.e., groups 1 and 2). In this manner, for the star topology, when reducing the number of peers in the cluster with smaller BW in its aggregated link (see 6a in Figure 2), performance increases (0a>6a). The opposite effect can be found if most peers belong to the slower cluster (6e has 80 peers in a cluster with an aggregated BW of 50, and the remaining 20 and 20 peers have a BW of 100 and 200 respectively) then, performance decreases dramatically due to saturation problems. In this case, the average number of hops is lower due to the fact that most traffic belongs to the larger cluster. Nevertheless, saturation appears when sending data to smaller clusters since most sources belong to the 80-peer cluster. Additionally, saturation also implies more cancelled data messages, which also increases dramatically the net cost. As for the ring topology, it presents a similar evolution, although alternate message routing paths diminish this effect to some extent.

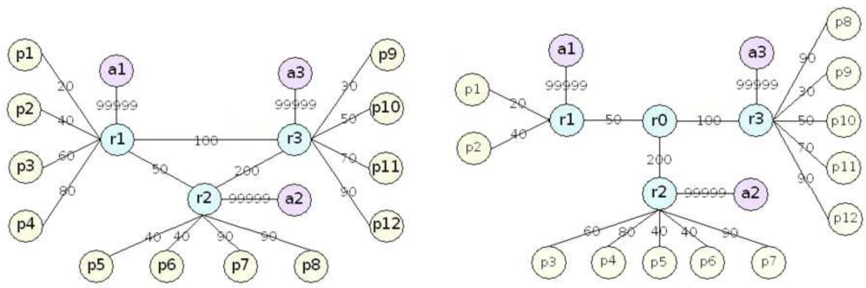
In general, ring layout is better than star layout when the number of peers is moderated, otherwise star layout scales better. Even more, when there are many peers with many clusters, then ring layout requires a larger number of hops whereas star layout gets closer to a constant number of hops. However, notice that a star layout requires a faster router at its central point in order to redirect all traffic. Regarding the bottlenecks coming from low bandwidth, they cannot be avoided at individual channels. In contrast, in the case of aggregated channels, a ring layout provides alternate paths that diminish such bottlenecks. Finally, in despite of network topology, our proposed Assistance layer successfully assists peers in their organisation since they tend to use local communications. Notice that netHop measure shows this behaviour since it reflects a number of hops lower than the required among different clusters —e.g. topology 6a requires 4 hops among two peers in different clusters (see Figure 2) but its average is 2.48 instead. The exception to this behaviour appears when the number of peers per cluster is drastically small (e.g. 4a-4f) or there is a large number of clusters in a ring layout (e.g. 5d). In the former case, the most of communications cannot be local, whereas in the latter, remote communications increase notably the average of hops. Thus, in both cases, it is no possible to decrease netHop. Overall, the results confirm our hypothesis, so 2-LAMA is robust when changing underlying communication network's topology. For instance, the times obtained when increasing the number of peers –in similar topologies– are sub-linear, so 2-LAMA is scalable when the number of agents grows.

5. Related Work and Discussion

Within MAS area, organisation-centred approaches regulate open systems by means of persistent organisations —e.g. Electronic Institutions [7]. Even more, several of these approaches offer mechanisms to update their organisational structures at run-time —e.g. Moise+ [1]. However, most work on adaptation maps organisational goals to tasks and look for agents with capabilities to perform them —e.g. OMACS [6]. Consequently, these approaches cannot deal with scenarios that lack of this goal/task mapping, like our case study. In order to deal with this sort of scenarios, our approach uses norms to influ-

Table 2. Experimental results for Table 1 topologies. Group descriptions outline most outstanding features of these topologies. Diamonds outline best performers within a group and bolds highlight global outperformers.

Group description	timeCost	netCost	netHop	netSat	netCol
0a: 12/3,star	890.58	221705.00	2.56	3.06	108.91
0b: 12/3,ring	**◇776.42**	**193891.67**	**2.37**	3.27	98.13
1a: 12/3,star,+het(ind)	1900.25	233623.33	2.66	8.39	189.44
1b: 12/3,star,+het(agg)	1649.75	228395.00	2.63	17.83	409.22
1c: 12/3,ring,+het(ind)	1888.67	203626.67	2.35	15.30	438.50
1d: 12/3,ring,+het(agg)	◇ 795.58	198182.50	2.44	3.14	99.05
2a: 12/3,star,hom(ind)	907.00	183093.33	2.40	0.24	8.12
2b: 12/3,star,hom(agg)	**837.33**	215486.67	2.55	2.58	91.50
2c: 12/3,ring,hom(ind)	791.67	169206.67	2.25	0.34	118.76
2d: 12/3,ring,hom(agg)	◇784.17	188433.30	2.32	3.22	100.49
3a: 30/3,star	1502.75	650656.67	2.34	4.26	94.66
3b: 30/3,ring	◇1306.83	585746.67	2.19	4.45	83.56
3c: 120/3,star	5762.08	4863316.67	2.08	11.17	145.05
3d: 120/3,ring	◇5674.00	4772626.67	2.05	11.24	131.38
4a: 12/12,star	◇850.25	359206.67	3.89	4.57	397.69
4b: 12/12,ring	1163.92	649653.30	6.03	5.78	374.45
4c: 30/30,star	◇1331.75	891553.30	3.86	2.57	328.91
4d: 30/30,ring	4574.00	2881410.00	10.18	6.22	375.24
4e: 120/120,star	◇2678.75	4712313.30	3.81	2.98	222.52
4f: 120/120,ring	32870.58	34682270.00	22.87	5.30	745.36
5a: 12/4,star	843.08	239436.67	2.78	2.97	94.35
5b: 12/4,ring	◇818.42	204332.50	2.52	2.75	97.86
5c: 30/10,star	**◇1300.30**	662840.00	3.00	3.31	116.13
5d: 30/10,ring	2485.75	2052474.47	8.23	3.68	109.70
5e: 120/40,star	**◇2483.42**	3155935.00	3.08	5.80	155.18
5f: 120/40,ring	9668.67	8829641.67	8.04	5.29	700.05
6a: 12(2,5,5),star	**◇775.25**	219005.00	2.48	2.96	71.87
6b: 12(2,5,5),ring	788.25	190625.83	2.24	1.96	70.94
6c: 30(6,12,12),star	◇1397.42	619868.30	2.24	3.89	75.87
6d: 30(6,12,12),ring	1425.00	605378.30	2.14	4.05	78.34
6e: 120(80,20,20),star	13258.50	6189638.30	2.05	9.93	164.53
6f: 120(80,20,20),ring	◇13181.83	6079686.67	2.03	10.02	160.09



(0b) (6a)

Figure 2. Network topologies 0b (12/3,ring) and 6a (12[2,5,5],star)

ence agent behaviour, instead of delegating tasks. Moreover, we propose a MAS abstract architecture to update such norms based on social power —see norm taxonomy [5]. Besides, the most of norm emergence works are agent-centred approaches that depend on participants' implementation and they rarely update persistent organisations —e.g. [10].

Regarding our P2P case study, there are some network management perspective approaches that also try to promote local communications but they cannot directly act on network consumption to balance net capacity and traffic —e.g. ONO [4] tries to achieve it without ISPs involvement whereas P4P [9] involves them. From a MAS angle, there are some works where agents adapt local norms using local information but they cannot reason/act at an organisational level —e.g. P2P normative system [8].

Our work compares the performance of our proposed abstract MAS architecture (2-LAMA) over different network topologies in such a P2P scenario. In particular, we present different topologies inspired in existing End-User / Internet Service Provider / Internet networks (*non-backbone-net*). Results show that 2-LAMA is able to scale to network sizes, heterogeneity of channel bandwidths, unbalanced clusters and path-layouts (ring/star). Overall, even when changing these features, 2-LAMA is still able to promote local traffic (low-latency) and update norms depending on network status. This means that our adaptive approach is able to improve system's performance in usual *non-backbone-nets*. This is specially relevant in a current P2P sharing network, since our approach does not require to increase current data sharing programs' complexity. Instead, the adaptation complexity is handled at a higher level by assistants residing in ISPs.

As future work, we plan to confront further issues in open MAS such as how the system should react to agents joining or leaving the MAS anytime, or transgressing its organisational restrictions. In fact, we have preliminary results about norm violations that show how system re-adapts to counter violation side effects. Besides, we are improving meta-level agents to use learning techniques in order to perform the adaptation process.

Acknowledgements: Work funded by EVE (TIN2009-14702-C02-01/TIN2009-14702-C02-02) & AT (CONSOLIDER CSD2007-0022) projects, and M. Esteva's Ramon y Cajal contract.

References

[1] O. Boissier and B. Gâteau. Normative multi-agent organizations: Modeling, support and control. In *Normative Multi-agent Systems*, 2007.

[2] Jordi Campos, Maite López-Sánchez, and Marc Esteva. Assistance layer, a step forward in Multi-Agent Systems Coordination Support. In *Autonomous Agents and Multiagent Systems*, pages 1301–1302, 2009.

[3] Jordi Campos, Maite López-Sánchez, and Marc Esteva. Norm Adaptation using a Two-Level Multi-Agent System Architecture in a P2P Scenario. In *Proceedings of COIN at AAMAS'10*, to appear 2010.

[4] David Choffnes and Fabián Bustamante. Taming the torrent: a practical approach to reducing cross-ISP traffic in peer-to-peer systems. *SIGCOMM Comput. Commun. Rev.*, 38(4):363–374, 2008.

[5] B.T.R. Savarimuthu S. Cranefield. A categorization of simulation works on norms. 2009.

[6] Deloach, Oyenan, and Matson. A capabilities-based model for adaptive organizations. *Autonomous Agents and Multi-Agent Systems*, 16(1):13–56, 2008.

[7] Marc Esteva. *Electronic Institutions: from specification to development.* IIIA PhD. Vol. 19, 2003.

[8] A. Grizard, L. Vercouter, T. Stratulat, and G. Muller. A peer-to-peer normative system to achieve social order. *LNCS*, 4386:274, 2007.

[9] H.Xie, Y.Yang, A.Krishnamurthy, Y.Liu, and A.Silberschatz. P4P: provider portal for apps. 2008.

[10] Norman Salazar-Ramirez, Juan A. Rodríguez-Aguilar, and Josep Lluis Arcos. An infection-based mechanism for self-adaptation in multi-agent complex networks. pages 161–170, 2008.