Searching and Sharing Information in Networks of Heterogeneous Agents

George A. Vouros
AI-Lab, University of the Aegean
Karlovassi 83200, Samos
georgev@aegean.gr

ABSTRACT
This paper extends existing methods for information searching and sharing in large-scale, dynamic networks of agents, to deal with networks of heterogeneous agents: Agents that do not share a common conceptualization of information categories and agents that have different preferences on specific information categories. The proposed method extends the method proposed in [7] by building overlay networks for specific information categories. The paper demonstrates through performance experiments the effectiveness of the method, even in cases where these agents shift their expertise.

Categories and Subject Descriptors
I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *multiagent systems*

General Terms
Algorithms, Management, Performance, Experimentation

Keywords
Artificial social systems, Performance, Scalability, Heterogeneous agents, Peer-to-Peer systems, ontology alignment.

1. INTRODUCTION
Considering to be a decentralized control problem, information searching and sharing in large-scale systems of cooperative agents is a hard problem in the general case [2]. The problem is even harder when agents are heterogeneous. These facts have resulted to efforts that either require agents to have a global view of the system, to heuristics [3], to pre-computation of agents' information needs and information provision capabilities for proactive communication [13], to localized reasoning processes built on incoming information [9,10,15], and to mathematical frameworks for coordination whose optimal policies can be approximated [8] for small (sub-) networks of associated agents: All these efforts assume that agents share a common conceptualization of their domain, and their preferences on information categories are common.

On the other hand, there is a lot of research on semantic peer to peer search networks and social networks (e.g. [1,4,5,6,12,14,15]). Methods involving the gradual creation of overlay networks via re-wiring, shortcuts creation [1,4,12] or clustering of peers [13,5] are tuning approaches whose aims are closely related to the approach presented in this paper. However, this paper deals with heterogeneous agents: Agents that do not share a common conceptualization of information categories and that have different preferences on information categories. In addition to this, it combines local interactions with the gradual establishment of overlay structures, as queries are being propagated in the network and agents advertise their information provision abilities given the interests of their neighbours.

The main objective of this paper is to study 'tuning' the information searching and sharing process in networks of heterogeneous agents. '*Tuning*' is the task of sharing and gathering the *necessary* knowledge for agents to propagate requests to the *right* acquaintances, minimizing the searching effort, increasing the efficiency and the benefit of the system. This paper extends the method for tuning information searching and sharing in dynamic and large scale networks proposed in [7] as follows: (a) To deal with networks of heterogeneous agents, (b) by establishing logical shortcuts, and thus by imposing overlay structures, (c) in settings where agents shift their expertise. It must be pointed that, according to our knowledge, this paper provides an important first study on the impact of the effectiveness of ontology alignment methods in information searching and sharing in large-scale networks. It does so by abstracting from the specific computations for measuring the semantic closeness of agents' conceptualizations.

This paper is structured as follows: Section 2 states the problem and section 3 presents the individual techniques and the overall proposed method. Section 4 presents the experimental setup and results, and section 6 concludes the paper, sketching future work.

2. PROBLEM STATEMENT
Let $N=\{A_1, A_2,...A_n\}$ be the set of agents in the system. The network of agents is modelled as a graph $G=(N,E)$, where N is the set of agents and E is a set of bidirectional edges denoted as non-ordered pairs (A_i,A_j). The neighbourhood of an agent A_i includes each A_j such that $(A_i, A_j)\in E$ (i.e. its acquaintance agents). The set of acquaintances of A_i is denoted by $N(A_i)$.

Each agent maintains (a) an ontology that represents categories of information (topics), (b) indices of information pieces available to its local database and to other agents, (c) shortcuts to agents out of its neighbourhood, and (c) a profile model for some of its known agents (denoted by $K(A_i)$). Profiles, indices and known agents are specified in section 3. Each agent has a set of information items in its local repository, which are classified under the concepts of its expertise. It must be pointed that agents having the same topics of expertise may not share the same ontology (i.e. these topics may be lexicalized or even axiomatized in different ways). In addition to this, agents may also differ in their preferences to these topics.

Cite as: Searching and Sharing Information in Networks of Heterogeneous Agents (Short Paper), G.A.Vouros, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 1525-1528.

Doing so, even if we assume that two agents A_i, and A_j share a common ontology and a specific topic of expertise c, given their classifiers T_i, T_j and any set of information items I, A_i shall infer that a set I_i of information items are of its interest and A_j shall infer that a set I_j of information items are of its interest, such that $I_i \subseteq I$ and $I_j \subseteq I$. No specific assumption holds for the relation between I_i and I_j. However, even if the above holds for any given set I, for the sake of simplicity in our experiments we assume that the sets of items in agents' local repositories are non-overlapping. The set of information categories in agent's A_i (A_j) ontology is denoted by C_i (respectively C_j). Finally, it is assumed that there is a set of k queries $T=\{t_1,\dots, t_k\}$. Each query is represented by a tuple $<id,a,c,path,ttl>$, where id is the unique identity of the query, a is a non-negative integer representing the maximum number of information pieces requested, c is the specific category to which the requested pieces must belong (c is a topic), $path$ is a path in the network of agents through which the query has been propagated (initially it contains the originator of the query and each agent appends its id in the $path$ before propagating the query), and ttl is a positive integer that specifies the maximum number of hops that the query can reach.

The problem that this article deals with is as follows: Given a network of agents $G=(N,E)$ and a set of queries T, agents must retrieve the pieces of information requested by queries, in concurrent search sessions, and further 'tune' the network so as to answer future similar queries in the more effective and efficient way: The *effectiveness* of the system is measured by its benefit, i.e. the ratio of information pieces retrieved to the number of information pieces requested. The *efficiency* of the system is measured by the number of messages needed for searching and updating the indexes and profiles maintained.

3. INFORMATION SEARCHING AND SHARING

3.1 Indices and Profiles

To capture information about pieces of information accessible by the agents, each agent A_j maintains a routing index that is realized as a set of tuples of the form $<A_i,c_j,s>$. Each such tuple specifies the number s of information items in category c_j that can be reached by A_j *through* A_i, such that $c_j \in C_j$ and $A_i \in K(A_j) \cup \{A_j\}$. This specifies the *information provision abilities* of A_i to A_j with respect to the information category c_j. The routing index is exploited for the propagation of queries to the "right" agents with respect to the known equivalences of information topics and to the agents' satisfaction on the provided answers (this is further explained below). The exploitation of routing indices reported in [7] has been extended to deal with content provider agents as well as with acquaintances (including recommenders would impose a high load on indices' maintenance due to their large number), taking also into account differences between agents' ontologies.

Considering an agent A_j, the profile model of one of its known agents A_i, denoted by P_{ji}, includes: (a) A set eq_{ji} of tuples $<c_{jk},c_{il}>$, denoting ontology concepts' equivalences, such that $c_{jk} \in C_j$ and $c_{il} \in C_i$: any information (queries, indices or query results) sent from agent A_j to agent A_i is translated according to concepts' equivalences eq_{ji}. These equivalencies can be decided by any ontologies' mapping mechanism. Such a mechanism has a specific matching factor (recall), MF, measuring the ratio of the concepts' equivalences computed by the mechanism to the number of the actual equivalences; and a satisfaction factor (SF) measuring the precision of the mapping method with respect to agents' preferences. (b) A set of tuples $<A_i,c_j,p>$ maintained by A_j. Such a tuple specifies the probability p that the agent A_i is interested to pieces of information in category c_j (this probability is denoted by p^{cj}_{ji}). The update of A_i's assessment on $p^{ci}_{i,j}$,caused by any incoming query $<id,a,c,path,ttl>$ from A_j, is computed by leveraging Bayes Rule [9,10] as it specified in [7]. (c) A set of tuples $<A_i,c_j,sf>$, maintained by A_j. Such a tuple specifies the percentage (sf) on the information items provided by A_i concerning the category c_i, such that: (i) c_j is equivalent to c_i, $c_j \in C_j$ and $c_i \in C_i$, (ii) A_j *assesses*, with respect to its own preferences, that only sf% of information items classified by A_i to the category c_i can by correctly classified in c_j by A_j. sf is called the satisfaction factor of A_j *from* A_i *with respect to the category* c_j and is denoted by sf^{cj}_{ji}. (d) A set of tuples $<A_i,c_j,sat>$, maintained by A_j, specifying the actual satisfaction (sat) of A_j, as far as the information items provided *by* A_i in category c_i are concerned (this satisfaction measure is denoted by sat^{cj}_{ji}).
Profile models $P_{ji}=\{<eq_{ij}, p^{cj}_{ji}, sf^{cj}_{ji}, sat^{cj}_{ji}> \mid A_i \in K(A_j) \text{ and } c_j \in C_j\}$ are exploited by the agents to translate information sent to known agents, to decide where to 'advertise' their information provision abilities and record their satisfaction from other peers.

3.2 Overall Method

Given two known agents A_i and A_j in G, the information searching and sharing process proceeds as it is depicted in Figure 1 (numbers show the sequence of tasks): Initially, (a) agents exchange information concerning their ontologies and decide on ontology concepts' equivalences, and (b) each agent has no knowledge about the information provision abilities of its known agents and also, it possesses no information about their interests.

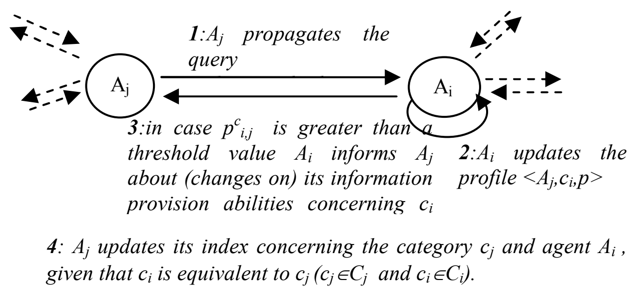


Figure 1. Information sharing between two *known* agents

Furthermore, given two (generally, not-known) agents A_j and A_k such that the later has received a query originated from the former, the information sharing and searching process proceeds as it is shown in Figure 2 (again, numbers show the sequence of tasks). As shown, A_k establishes a "recommendation" link to A_j. Such a link has the form $<A_k,A_j,c_k>$, $c_k \in C_k$. Doing so, the set of known agents of A_k is extended to include A_j. Also, A_j, when it receives the items from A_k, proceeds to establish/maintain a content provider link to A_k. Such a link has the generic form $<A_j, A_k, c_j, \#items_k>$, where $\#items_k$ is the number of items provided by A_k, and c_j in C_j is equivalent to c_k. Similarly to recommendation links, this leads A_j to extend its own set of known agents by including A_k to the set of content providers. For each content provider of A_j, i.e. for each agent A_i for which there is a content provider link $<A_j,A_i,c_j,\#items_i>$, A_j computes its actual satisfaction

(sat^{cj}_{ji}) from A_i concerning the category c_j. This is done according to the following formula: sat^{cj}_{ji}=(#items / $\sum_{X\ s.t\ <A_j,\ X,\ c_j,\ \#i>}$ #i)× sf^{cj}_{ji}, where the denominator sums the number of items in category c_j that have already been provided by all known content providers. *sat* shows the "direct" effectiveness of a content provider, enforcing a myopic policy to information searching.

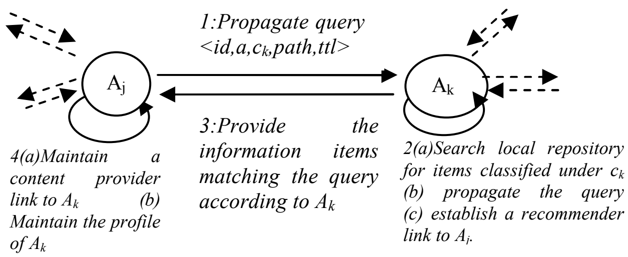


Figure 2. Information sharing between any two agents

Concluding the above, given an agent A_j in *G*, the set $K(A_j)$ of known agents of A_j includes the acquaintances of A_j, $N(A_j)$, the content providers (denoted $CP(A_j)$) and the recommenders (denoted $R(A_j)$) known to A_j, for any information category of its own interest. Formally, $K(A_j) = N(A_j) \cup CP(A_j) \cup R(A_j)$ where,
$CP(A_j)=\{A_i \mid$ there is a content provider link $<A_j, A_i, c_j, \#items_i>$ for any $c_j \in C_j\}$
$R(A_j)=\{A_i \mid$ there is a recommendation link $<A_j, A_i, c_j>$, for any $c_j \in C_j\}$

By establishing content provider and recommendation links, agents dynamically establish overlay structures for efficient and effective information sharing: Content providers for which a high satisfaction (*sat*) has been reported concerning an information category can provide immediate answers to queries, while recommenders are very likely to know providers that can best satisfy a query.

3.3 Tuning

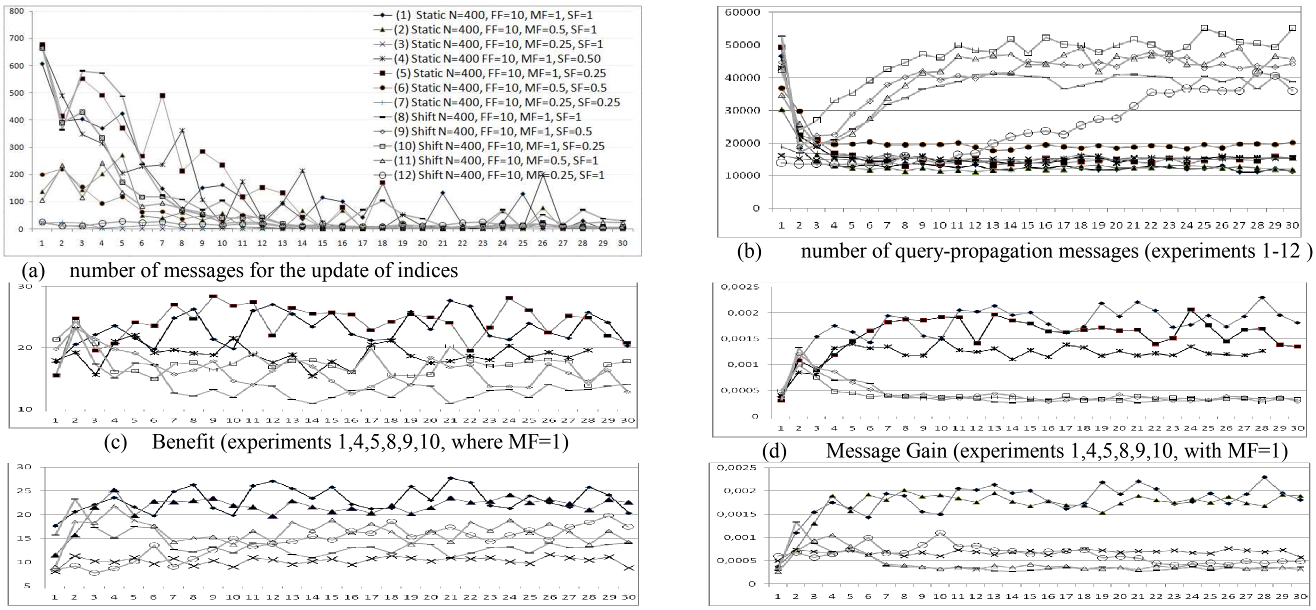
Tuning is performed seamlessly to searching: As agents propagate queries to be served, their profiles are getting updated by the other agents and new logical connections are being established, forming overlay networks which further facilitate queries' propagation. As profiles are getting updated, agents receive the aggregated indices of other agents [7] and measure their satisfaction on the information provision abilities of their acquaintances and content providers. Specifically, given a query, agents propagate it by applying the following rules in order: (a) To any content provider that can serve the query according to the number of provided items and to the satisfaction factor. (b) To a content provider or acquaintance that has the required information provision abilities to serve the query, according to its routing index and satisfaction factor. (c) To content providers with the greater *sat* as well as to any of the recommenders. (d) To those content providers and acquaintances with the highest information provision abilities. An agent may propagate a query to any percentage of its acquaintances. This percentage is called *Flooding Factor* (*FF*).

In a dynamic setting, agents may shift their expertise, their interests, they may leave or join the network at will. In this paper, emphasizing on agents' heterogeneity, we study settings where agents may shift their expertise: Agents update their local repository by deleting existing information items from their local repositories, and by adding new information items in their new area of expertise. These changes cause the update of routing indices. Since agents need valid and updated information about their content providers (who may have shifted their expertise) we enforce a simple policy, keeping the most recently used providers. Currently, we have set the number of the most recently used content providers equal to 5.

4. EXPERIMENTAL SETUP

To validate the proposed approach we have built a prototype that simulates large networks. Here we present results from a small-world network with $|N|$=400: These are representatives of other cases as well. Details about experiments', together with experiments' ids, are shown in Figure 3(a). Each experiment ran 30 times. In each run the network is provided with a new set of randomly generated queries. The agents search and gather knowledge that they further use and enrich. Each run lasts a number of rounds that depends on the *ttl* of queries (currently set equal to 6). Information used in the experiments is synthetic and is being classified in 33 distinct categories of each of the 11 ontologies: Each agent is been assigned a specific ontology and a unique information category that represents its expertise. However, there are agents that they may not have assigned an ontology, or they may have not a specific expertise. For each sub-category in its expertise, each agent holds at most 3000 information pieces, the exact number of which is determined randomly. Concerning satisfaction, given an agent A_j, for each information category c_j, and A_i in $K(A_j)$, a specific satisfaction factor sf^{cj}_{ji} is being assigned randomly during runtime. The satisfaction factor must be greater than the *minimum satisfaction factor* (*SF*) provided as input in the simulator. Each query is randomly assigned to an originator agent and is set to request a random number of information items, less than 6000. In such a setting, the demand for information items is higher than the individual agents' information provision abilities, given the *ttl* of queries.

As it is shown in Figure 3, as agents search and share information from run 1 to run 30, they manage to drastically reduce the number of messages. Also (not shown here due to space reasons) the number of unfulfilled queries decrease, while the served queries increase gradually. Experiments show an effective tuning of the networks as time passes and more queries are posed to the network, even if agents maintain the models of a small percentage of their acquaintances (*FF=10*). Also, according to experiments (not shown here due to space reasons) the 'tuning' approach is robust to settings where agents cannot reach an agreement to the mappings of their conceptualizations (MF=0.5 or MF=0.25) and in cases where agents have quite different preferences to specific information categories (SF=0.5 or SF=0.25). In other words, high precision and recall of the ontology mapping task can greatly facilitate information searching and sharing in networks of heterogeneous networks, however, the tuning task is still effective for lower values of mapping precision and recall. Also, the method is quite effective in settings where a great number of highly heterogeneous agents shift their expertise (in average 80 agents per run), to the cost of increasing the number of messages, and thus, reducing the message gain. We have to emphasize that, despite the large variances of the results presented, the tuning task reaches a plateau in the early runs. This is also true for the overlays networks built: These are nearly stabilized in early runs. This shows the effectiveness of the method, even in settings where information is highly distributed and scarce.



(a) number of messages for the update of indices

(b) number of query-propagation messages (experiments 1-12)

(c) Benefit (experiments 1,4,5,8,9,10, where MF=1)

(d) Message Gain (experiments 1,4,5,8,9,10, with MF=1)

(e) Benefit (experiments 1,2,3,8,11,12, shown where SF=1)

(f) Message Gain (experiments 1,2,3,8,11,12, where SF=1)

Figure 3. Experimentation results grouped per type of experiment

5. CONCLUSIONS

This paper presents a method for tuning large networks of heterogeneous agents search and share information effectively. The proposed method has the following features: (a) It extends the method proposed in [7] by considering agents' heterogeneity that is due to agents' different conceptualizations of information categories, as well as to agents' different preferences. (b) It establishes logical content-provider and recommendation links between agents imposing overlay network structures. (c) It supports the acquisition and exploitation of information that is not only locally available (i.e. information available via agents' immediate acquaintances) but also information available via content providers and recommenders.

6. ACKNOWLEDGMENTS

Many thanks to Ioannis Partsakoulakis. This work was supported by GR Pythagoras grant number 1349 under the Operational Program for Education and Initial Training.

7. REFERENCES

[1] Cooper, B.F., Garcia-Molina, H. Ad hoc, self-supervising peer-to-peer search networks. *ACM TOIS*, Vol.23 ,Issue 2 (April 2005) ,169 - 200

[2] Goldman, C., and Zilberstein, S. Decentralized Control of Cooperative Systems: Categorization and Complexity Analysis. *JAIR,* 22 (2004), 143-174.

[3] Goldman, C., and Zilberstein, S. Optimizing Information Exchange in Cooperative Multi-agent Systems. In *Proc. of AAMAS 2003*.

[4] Loser, A., Staab, S., Tempich, C. Semantic Social Overlay Networks. *IEEE Journal on Selected Areas in Communications*, 25 (1): 5--14. January 2007

[5] Nejdl, W., Wolpers M., Siberski, W., Schmitz, C., Schlosser, M., Brunkhorst, I., Loser, Super-Peer-Based Routing and Clustering Strategies for RDF-Based Peer-to-Peer Networks.*In Proc. of WWW2003,* 536-543.

[6] Tempich, C., Staab, S., Wranik, A. REMINDIN': Semantic Query Routing in Peer-to-Peer Networks Based on Social Metaphors. In *Proc. of the WWW04,* 640-649.

[7] G.A.Vouros, Information Searching and Sharing in Large-Scale Dynamic Networks, In *Proc. of AAMAS'*07.

[8] Xu, Y., Scerri, P., Yu, B., Lewis, M., and Sycara, K. A POMDP Approach to Token-Based Team Coordination. In *Proc. of AAMAS'05*.

[9] Xu, Y., Lewis, M., Sycara, K., and Scerri, P. Information Sharing in Large Scale Teams. In *AAMAS'04 Workshop on Challenges in Coordination of Large Scale MultiAgent Systems*, 2004.

[10] Xu, Y., Liao, E., Scerri, P., Yu, B., Lewis, M., and Sycara, K. Towards Flexible Coordination of Large Scale Multi-Agent Systems. *In Challenges of Large Scale Coordination* Springer, 2005.

[11] Xu, Y., Scerri, P., Yu, B., Okamoto, S., Lewis, M., and Sycara, K. An Integrated Token Based Algorithm for Scalable Coordination. In *Proc. of AAMAS'05,* 407-414

[12] Yu, B., and Singh M. Searching Social Networks, In *Proc. of AAMAS'03*.

[13] Zhang, Y., Volz, R., Ioeger, T.R., Yen, J. A Decision Theoretic Approach for Designing Proactive Communication in Multi-Agent Teamwork. In *Proc of SAC'04,*2004, 64-71.

[14] Zhang, H., Lesser, V. Multi-Agent Based Peer-to-Peer Information Retrieval Systems with Concurrent Search Sessions. In *Proc. of AAMAS'*06, 305-31.

[15] Zhang, H., Lesser, V.A Reinforcement Learning based Distributed Search Algorithm for Hierarchical P2P Information Retrieval Systems. *In Proc of AAMAS'07*, 219-226.