# Decentralized Semantic Service Discovery in Preferential Attachment Networks

E. del Val, M. Rebollo, and V. Botti

Grupo de Tecnología Informática - Inteligencia Artificial
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vera S/N 46022 Valencia (Spain)
{edelval,mrebollo,vbotti}@dsic.upv.es

**Abstract.** Service discovery plays an important role in large-scale and highly dynamic environments where the most valuable information is not widely available and may not be registered. In this paper, we present a distributed service discovery approach which makes use of decentralized search algorithms and social network models as underlying structure.

## 1 Introduction

In new paradigms for computing, such as peer-to-peer technologies, grid computing or autonomic computing, large systems can be seen in terms of service provider and consumer entities or agents [15]. The main feature of these domains is that they are open and dynamic, where new agents can enter to the system and existing ones leave. If we consider agents as service providers, the available services change dynamically and it is not an easy task to locate a suitable and available service in a crowded environment with services. In this contex, one of the most challenge issue is service discovery. Conventional approaches in multiagent systems such as registries or matchmakers partially address this problem. However, in highly dynamic environments, the most valuable information is not widely available or it is registered in a centralized repository or may not be registered[28]. Much of this information may only be accessed by contacting the right agents. This fact is present in human society. There are scenarios, such as labor market, where the empirical evidences suggest that about half of all jobs are filled through contacts[7]. Recent literature stresses the role of contact networks in other economic phenomena such as buyer-sellers[12] or R&D (Research & Development)[6]. In the provider-consumer scenario, individuals seeking services read yellow-pages, browse in the web and mobilize their local networks of friends and relatives. Networks of personal contacts can mediate in provider-consumer location opportunities which flow through word-of-mouth and constitute a valid alternative source of service information to more traditional methods.

In this paper, we propose a distributed service discovery approach for Open Multi-Agent Systems (Open MAS) using social networks as underlying structure. When an agent asks for another service, a distributed search is made considering only local information associated to its neighbors: degree and service parameters contained in semantic service descriptions offered by each neighbor.

The paper is structured as follows. Section 2 gives an overview of several works in the area of service discovery in MAS. Section 3 describes the structure and the advantages of preferential attachment networks. In section 4, the proposal for distributed semantic service discovery is presented. Section 5 analyzes the performance of the proposal, comparing with other approaches used in distributed search. In section 6 conclusions and future work are presented.

## 2 Related Work

Open and dynamic environments where the scalability and the workload are low make use of middle-agents [26][11][17] to facilitate service discovery. The main advantage is that matchmakers could provide an optimal matching because they consider all the registered services in the system. These middle-agents usually make an efficient search and get a good throughput. Unfortunately, this kind of agents could be a bottleneck when the workload increases. Other drawbacks of middle-agents are their complexity, the huge amount of memory needed to keep service advertisements and the cost of service composition as the number of services grows significantly. Different approaches have been suggested to overcome the above mentioned problems related to the centralized paradigm in service discovery.

*Peer-to-peer* approach takes advantage of the fact that each agent already knows its own capabilities and those of a few peers, and uses peer-to-peer search (recursively) for locating agents with the needed capability [5][23]. An agent broadcasts a query using its local knowledge to its neighbors and the agent that receives such a request either offers its services to the original caller or broadcasts the request to its own neighbors. The drawback of this approach to service discovery is that the communication among agents is essential and the overall communication traffic overhead may be large.

Another distributed way to locate distributed services is to form *coalitions* or clusters[22][18]. Nevertheless, the choice of what coalitions are going to be formed is a difficult task. This entails recursively to calculate the values of the coalitions and later selecting the coalition with the best result. The calculation of the coalition values can be made in parallel, but this phase requires that each agent knows the rest of sytem agents (global knowledge). In addition to determine the best value, they have to use broadcast. Therefore, in some situations, the system could be overload.

A third way for agents to discover services in efficiently is the distribution of the *middle agents* or *facilitators*. Jha et al.[8] suggest to split the function of the service facilitator among a group of agents. The system designer assigns a local matchmaker to each host or segment of the system, which provides matchmaking services to agents in its vicinity (its segment). The local matchmaker can consult its peers or a central matchmaker whenever it cannot provide an answer to a local query. This type of solution reduces communication traffic and confines it to network segments (in which communication is fast). Moreover, it reduces message queue sizes, improving scalability and fault tolerance. Sigdel et al. [25] present an adaptative system. The framework suggested allows automatically adaptable matchmaking methods for service localization depending on the network structure and characteristics. This approach is based on two levels: *system adaptation level* and *node adaptation level*. These approaches are applicable in systems that have a hierarchical topology, in which information sharing can be confined to local segments. In systems with very large segments the problems of scalability are only marginally relieved by this approach because the large segments become overloaded systems which have local bottlenecks. Another case in which this approach is not useful is in systems with many crosslinks between segments. In this case the overhead of coordinating tasks among local matchmakers might be greater than the benefit obtained from their distribution.

The main advantage of the presented proposals are fault tolerant and adaptable to changes in the environment. Besides that, they decrease communication time and spread the information among agents. The main drawbacks are that distributed approaches such as coalitions or peer-to-peer have performance problems (network traffic, slow response and congestion) and the coordination effort required is not appropriated for highly dynamic environments. Our proposal tries to overcome these drawbacks through a completely distributed approach based on social networks as underlying structure. In the next section the main features of these networks are presented.

## 3 Social Networks

As MAS continue to grow and migrate to heterogeneous environments, such as the Internet and the Semantic Web, the structure of societies in MAS and the interconnections among the agents in these societies will be fundamental to the effectiveness of service discovery. Social network models are an appropriate representation of agent interconnections, such as friendship, financial exchange, relationships of beliefs, knowledge or prestige. Recent studies using data on communication within organizations[1] and the friendships communities[14] have established the fact that human social networks closely match some mathematical models present in social networks. Another interesting feature of these networks is the property of being *searchable*: *'ordinary people are capable of directing messages through their acquitance networks to reach a specific but distant target person in only a few steps'*[30]. This feature makes social networks not only suitable to model relationships between agents, but also to discover services offered by agents, situated in large networks whose topology is known only locally, in few steps.

To frame the underlying problem, we go back to one of the most well-known social network analysis: *'Six degrees of separation'*[27]. In this experiment Milgram discovered that individuals are connected via short paths, but also that the individuals in these networks, only considering local information about their own neighbors, are able to find these paths. Decentralized search can be classified considering if the network is structured or unstructured. This classification is presented in [31]: *'in structured networks the global position of the target node in the space can guide the search process to reach the target node more quickly. In unstructured networks, the global position of the node is unknown and it is difficult to know whether a step in the search process is towards the target node or away from the target node'*. One of these model sof unstructured networks is the *scale-free*[2].

*Scale-Free Networks.* The scale-free network model is defined as: *'a mathematic model extracted from the real world. The distribution of the number of network neighbors* *(degree distribution) is typically right-skewed with a heavy tail, meaning that a majority of nodes have less-than-average-degree and that a small fraction of hubs are many times better connected than average'*[29]. This qualitative description can be satisfied by several mathematical functions, but the most common in the current literature is a power law [2]:

$$P[k] \sim k^{-\alpha} \tag{1}$$

in which, $k$ is an integer denoting the node degree, $P[k]$ is the probability that a node connects with $k$ other nodes. The parameter $\alpha$ is a scalar coefficient, which usually ranges in:

$$\alpha \in (2, \infty) \tag{2}$$

The power law distribution denotes that some nodes have high degree although most nodes have low degree. This property is called *preferential attachment*: new network members prefer to make a connection to the more popular members in the network.

These kind of networks have many predominant advantages which can be used to improve the cooperative performance in Open MAS, for instance in service discovery. This model has a robust topology which is immune to random errors such as random removal of links or nodes. Thus, for service location, the preferential attachment network offers a reliable topology to ensure that a service can be found under the condition that certain agents leave the system. The main disadvantage is that this kind of networks are very sensitive to 'sabotage' (attacks to highly connected nodes). Another feature of these networks is the path length between two nodes: *'with most disordered networks, such as the small world network model, the average distance between two vertices in the network is very small relative to a highly ordered network such as a lattice*[21]. More concretely, power-law graphs having $2 < \alpha < 3$ have small diameter $log(n)$ where $n$ is the number of nodes.

Due to all the described features of scale-free networks with preferential attachment, a service discovery system has been proposed based on this this kind of complex networks. In the next section this system is described with detail.

## 4 Social Discovery System

We formulate the service discovery problem in an Open MAS as a probabilistic decision-making task in which the goal is to find an appropriated service minimizing the length of the path travelled by the request message. Our system is based on social networks, therefore agents are situated in a network with preferential attachment. We assume that each agent knows about its immediate neighbors including their identity, degree, and parameters related to the service they offer but it is unaware of the rest of the agents in the network. At the source agent, and at each agent along the path, the optimal decision rule is to send the message to the neighbor from which the message will reach the target agent which offers the desired service in the smallest number of steps, assuming that all future agents will make their decision using the same algorithm and only considering local information related to its neighbors.

If a decentralized search is to succeed, an important point to consider is that the underlying network possess some form of structure that can help to guide the search. There are two features that structure the preferential attachment network: *degree* (it is an intrinsic property of preferential attachment networks) and *homophily*. In the next subsections, the main concepts and components of the service discovery system are described.

### 4.1 Modeling Agent Homophily

*Homophily* is a compact word that expresses the idea that a contact between similar people occurs at a higher rate than among dissimilar people[16]. This is often say with the expression *'Birds of a feather flock together'* - that you tend to be friend, talk to, work with and share ideas with people who share with you a common ethnic, religious and economic background. This word was used by Lazarsfeld and Merton in 1954 in an essay titled *'Friendship as a Social Process'*[13]. Empirical work related to *homophily* within social networks shows that is one of the most robust and pronounced characteristics of social networks. There are two types of homophily: status homophily, *'individuals are considered similar to one to another on the basis of informal, formal or ascribed status'*, and value homophily, *'individuals are considered similar to one another on the basis of shared values, attitudes, and believes'*[3].

There is no global and application independent law on how *homophily* is measured. It is difficult to select an appropriate measure for a particular application area and to compare the existing homophily measures. Although homophily measurement is not restricted to solve a particular task, most homophily measures have been developed for a specific purpose. In our network, the homophily between two agents is based on semantic information contained in the service descriptions.

Given the agents $a_1$ and $a_2$, the homophily between them is calculated as follows. If we consider $s_1$ and $s_2$ as the services offered by the agents $a_1$ and $a_2$ respectively,

$$s_1 =< I_{s_1}, O_{s_1} > s_2 =< I_{s_2}, O_{s_2} > \tag{3}$$

the homophily between $a_1$ and $a_2$ can be computed as:

$$homophily(a_1, a_2) = \alpha sim(I_{s_1}, I_{s_2}) + \beta sim(O_{s_1}, O_{s_2}) \tag{4}$$

in which $\alpha$+$\beta$ =1, 0≤$\alpha$≤1, and the values of $\alpha$ and $\beta$ depends of the number of inputs or outputs of the services. If the number of inputs is higher than the outputs, the value of $\alpha$ will be higher than the $\beta$. The similarity function $sim(X, Y)$, where $X$ and $Y$ represent the input or output parameters of two services, means the degree that $Y$ satisfies $X$ and is defined as a bipartite matching problem for service inputs and outputs.

A matching of a bipartite graph $G$=($V$,$E$) is a subgraph $G'$=($V$,$E'$), $E' \subseteq E$, such that no two edges $e_1$,$e_2$ $\in E'$ share the same vertex. Given a bipartite graph $G$=($V_1 \cup V_2$,$E$) and its matching $G'$, the matching is complete if and only if all vertices in $V_1$ are matched.

Let consider $S_{1out}$ and $S_{2out}$ the set of concepts in services $s_1$ and $s_2$ respectively. Consider the graph $G$=($V_1 \cup V_2$,$E$) where $V_1$=$S_{1out}$ and $V_2$=$S_{2out}$. Consider two concepts $o_i \in V_1$ and $o_j \in V_2$. We differentiate among the four degrees of match proposed by Paolucci et al. [24]. We calculate the degree of match of this two concepts using a semantic similarity measure. With the value of this measure we decide the degree of match $R$. $R$ can be one of these values: $Exact$, $Plugin$, $Subsume$ or $Fail$. If $R$ is one of these degrees, an edge is defined between ($o_i$ ,$o_j$), $o_i \in V_1$ $o_j \in V_2$ in the graph and label it with a weight ($\omega_{ij}$). Once we have the weighted bipartite graph, we have to compute a complete matching of the bipartite graph such that the sum of weights of the edges in the matching, $\Sigma\omega_{ij}$, is minimized. For this task we use the Hungarian algorithm [19] which computes it in a polynomial time bound.

### 4.2 Agent Social Network

In this proposal, we use a preferential attachment network $G = (V,E)$ which consists of a set of nodes $V$ and a set of edges $E$ between them. The set of nodes represent agents which offer semantic services. The edges represent a relationship between agents which provide similar services. This network possess some form of structure that can guide the search. Basically, the preferential attachment network has two features that create such structure [4]. The first is *homophily*: agents tend to be linked with other agents that have services with similar category. The second feature is *degree*: some agents have more neighbors than others and may act as hubs that connect agents with different service categories. The consideration of *homophily* favors the neighbors that offer a service more similar to the target service. Consideration of *degree* favors the neighbor with the highest degree.

We create an undirected network with a power-law degree distribution. Each agent in the network offers a semantic service and has defined two vectors: one with the service inputs ($I's$) and the other with the service outputs ($O's$). Each $I/O$ is a semantic concept defined in an ontology. The link between two agents is established considering the ratio preference between agents $a_1$ and $a_2$ to the sum of preferences from $a_1$ to all the agents in the network. To approximate the preference from $a_1$ to all the agents in the network, using only local information, we use the degree of the agent $a_1$ ($k_{a_1}$) and the preference between the agent $a_1$ and its neighbors.

$$q_{a_1,a_2} = f_{a_1,a_2}/k_{a_1} * max(f_{a_1,neighbour}), \tag{5}$$

The preference between two agents $a_1$ and $a_2$ with services $s_1$ and $s_2$ respectively $f_{a_1,a_2}$ is defined as follows:

$$f_{a_1,a_2} = (max\{homophily(s_1, s_2), 0.01\})^r \tag{6}$$

where $homophily(s_1, s_2)$ is the homophily function between the service offered by agent $a_1$ and the service offered by agent $a_2$. The return value of these function is a real number which ranges in the interval [0..1] (1 if the service provided by agent $a_1$ is equal to the service provided by agent $a_2$). The $r$ parameter is a homophily regulator. When $r$ is zero, the graph shows no homophily, agents are not grouped by similar service categories. As $r$ grows, links connect agents with more similar services. Basically $r$ makes the network to show groups of agents (communities) with similar services [9].

### 4.3 Semantic Distributed Searching of Services

Preferential attachment networks grow according to a simple self-organizating process. These networks need efficient search algorithms in order to function well. Algorithms should rely on local information in order to avoid a dependence on a unique point of failure and to avoid the effects of the changes in the network structure. There are several algorithms proposed for decentralized search in networks. Some methods do not consider the special features for the corresponding network models such as breadth-first searching methods based on limited flooding or random walks [32]. By making use of special features of the system topologies the algorithms can be classified in three groups:

- *degree*: the degree-based search methods typically make the assumptions defined in [32]: (i) *'each node knows its own neighborhood network topology'*; and (ii) *'each node can locate the target if and only if the target is within a certain range of its neighborhood'*. Generally speaking, the algorithm navigates through the network selecting in each step the neighbor agent with highest degree. In case that all the neighbor agents have been visited, the algorithm selects one randomly.
- *similarity*: The algorithm basically navigates the network selecting in each step the neighbor agent which has the service more similar to the target service. If all the neighbor agents have been visited, the algorithm selects one randomly [32]. In our case, it navigates using semantic similarities among service description (or parameters) using formula 4 as similarity measure.
- *mixed*: the algorithm navigates through the network selecting the neighbor agent whose service is more similar to the target service. In case that the neighbor agent do not offer the information needed to calculate the similarity between services, the algorithm selects the next agent between its neighborhood considering the degree. In the case that similarity and degree values are available, both parameters can be used to calculate the next neighbor. If all the neighbor agents have been visited, the algorithm selects one randomly.

In the context presented in this paper, the selected algorithm to search in preferential attachment networks is the Expected-Value Navigation(EVN) described in[4] which is a *mixed algorithm*. To apply this algorithm in the agent network, it is necessary to consider degree and the *homophily* between agents which is based on semantic similarity between the semantic services provided by the agents (see formula 4). In our scenario we assume that if the agents do not share the same ontology a previous step of ontology alignment is done. With this information, we can estimate the probability that a link exists from one agent to another. This probability is calculated assuming that each link is placed independently of the others. For a link from agent $a_1$ to agent $a_2$ the probability $p_{a_1a_2}$ can be calculated as the inverse of $q_{a_1a_2}$:

$$p_{a_1a_2} = 1 - (1 - q_{a_1a_2})^k \tag{7}$$

where $q_{a_1a_2}$ is the probability that the first link for $a_1$ ends at $a_2$ (see formula 5), and $k$ is the degree of node $a_2$.

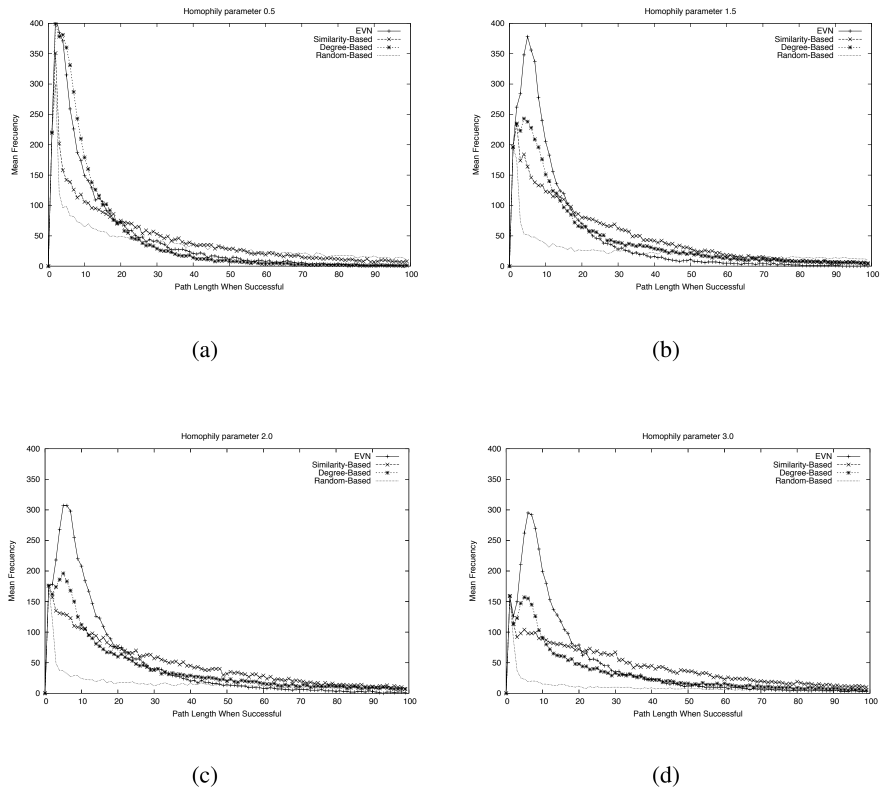



Fig. 1: Path Length When Successful

# 5 Experiments

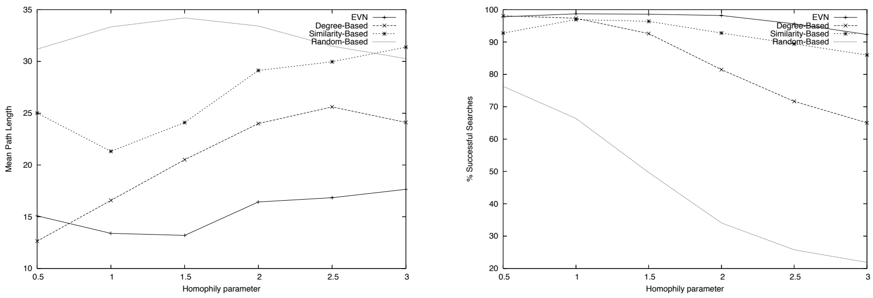
## 5.1 Network Characterization

The experiments have been done in a set of synthetic networks. These networks are preferential attachment networks with the features explained in section 4.2. Each network is composed of 1000 agents with one semantic service each one. There are 100 service categories. The services have been assigned to the agents using a uniform distribution. We have created six sets of 10 random agent social networks. Each set of networks have been generated with different homophily parameter. This degree ranges from 0.5 to 3.0.

## 5.2 Experimental Results

In order to evaluate the proposed service discovery system in preferential attachment networks, we have analyze the behavior of the *EVN* algorithm with respect the other distributed searching algorithms: *random*, *degree* and *similarity*. We have made 5000 searches in each of the previous networks.

In Figures 1 and 2 the data gathered from the previous described experiment is shown. In figure 1 we present the results obtained varying the homphily parameter from 0.5 to 3.0. Each figure indicates the frequency of path lengths for each distributed algorithm (EVN, degree, similarity and random). From these set of graphs we see that the EVN algorithm, in general has a better performance than the other algorithms independently of the homophily parameter. The EVN algorithm has the higher frequency of short paths (around 9 hops). In Figure 1a, the EVN have the same behavior than the degree-based algorithm. This is due to the homophily degree is too low, so the network does not show homophily and the EVN algorithm follows selects the neighbors only considering the degree. When the homophily parameter increases (Fig.1b,1c)), the performance of the EVN becomes better than the other algorithms. This is because the EVN considers both parameters to guide the search: degree and homophily and can take more advantage of the network structure. The best performance of the EVN with respect the others is with the homophily parameter varying from 1.5 to 3.0 (Fig.1b - 1d).

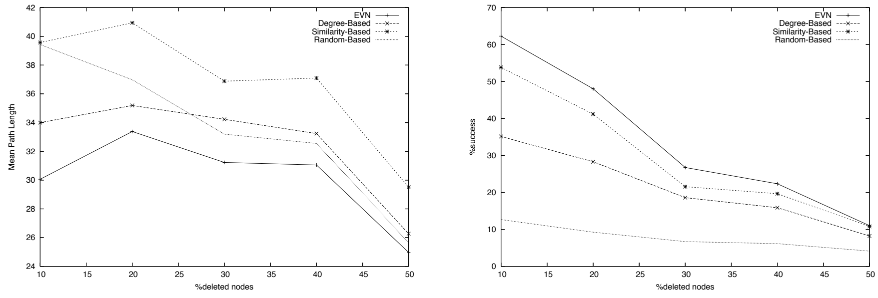



(a) (b)

Fig. 2: Network with different homophily degrees

Figure 2a shows the mean path length obtained with each algorithm in networks with the homophily parameter varying from 0.5 to 3.0. In general, the EVN always return the shortest path except in graphs with a low value of the homophily parameter. This is because EVN takes more advantage of the network structure. In Figure 2b the success rate of each algorithm is shown. The EVN algorithm in the 90% of searches finds a path between the source agent to the agent that has the service that it was interested in.

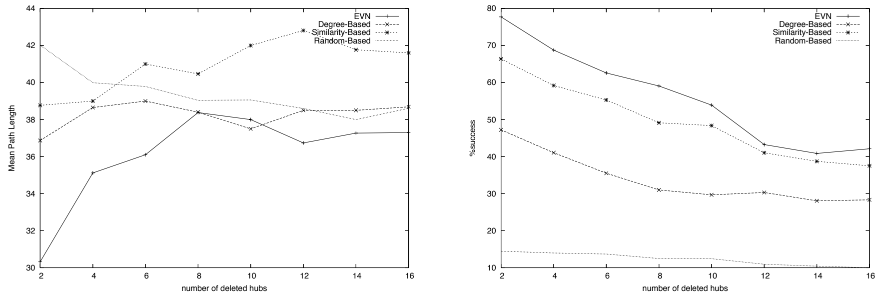
The last and very important check is the behavior of the network under failures. The problem appears when a broken link splits the network into tow isolated parts, since some nodes will no longer be reachable. To analyze it, node failures have been




(a) (b)

Fig. 3: Network with random failures

modelled as a failure of all its connexions. When some links are broken, an alternative path has to be found.




(a) (b)

Fig. 4: Network under 'sabotage' conditions

For random failures (see Fig.3a and Fig.3b), it can be observed that the path length decreases when the number of deleted nodes increases. But this is a consequence of the failure in most of the searches when the percentage of deleted nodes approaches 30% because many searches cannot end successfully.

An interesting case is what happens when a deliberate failure is provoked. In the case of power-law networks, the worst case occurs when nodes with highest degree (hubs) are disconnected. Figure 4a and 4b shows how 'sabotage' affects the performance of the search process. In this case, the path length increases due to only a few highly connected hubs have been deleted and an alternative path exists. The performance attending the number of successful searches decreases considerably as the number of deleted hub increases.

## 6 Conclusions and future work

The aim of this work is to investigate the use of social networks and distributed search algorithms to provide a fully distributed service discovery approach in Open MAS environment. Our proposal tries to overcome drawbacks present in other centralized (bottlenecks, complexity, huge amount of memory needed, global knowledge) and distributed (network traffic, congestion, coordination effort, data consistency between distributed registries, update data) discovery approaches. In our proposal, agents are situated in a social network with homophily factor. Each agent maintains the information about the current available services it offers. Agents in the network act as a 'matchmakers' and make use of a distributed search algorithm (EVN) that only makes use of local information to guide the search. The experimental results show that the EVN can be considered a good algorithm for service discovery domain.

As a future work we consider how does the problem of service discovery changes when the network evolves over time and what happen when agents do not follow a fixed algorithm. Furthermore, we will consider organizational information in the discovery process to guide the search.

## Acknowledgment

This work is supported by TIN2009-13839-C03-01 and TIN2008-04446 projects, CONSOLIDER-INGENIO 2010 under grant CSD2007-00022, FPU grant AP-2008-00601 awarded to E. del Val.

## References

1. L. A. Adamic and E. Adar. How to search a social network, 2004.
2. A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.
3. B. Chaudhry, C. Marton, and H. Shepherd. Homophily and structure in multiplex networks.
4. O. Şimşek and Jensen. Navigating networks by using homophily and degree. *Proceedings of the National Academy of Sciences*, 2008.
5. J. Dang and M. Hungs. *Concurrent Multiple-Issue Negotiation for Internet-Based Services*. Number Vol.10 - 6. 2006.
6. S. Goyal and J. Moraga. R&d networks. Econometric Institute Report 202, Erasmus University Rotterdam, Econometric Institute, 2000.
7. H. Holzer. Search method used by unemployed youth. *Journal of Labor Economics*, 6:1–20, 1988.
8. S. Jha, P. Chalasani, O. Shehory, and K. Sycara. A formal treatment of distributed matchmaking. In *Proc. of the 2nd Int. Conference on Autonomous Agents*, number Vol.3, pages 457–458, 1998.

9. J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *in Proceedings of the 32nd ACM Symposium on Theory of Computing*, pages 163–170, 2000.
10. J. M. Kleinberg. Navigation in a small world. *Nature*, 406(6798), 2000.
11. M. Klusch, B. Fries, and K. Sycara. Automated semantic web service discovery with owls-mx. In *Proceedings of 5th AAMAS*, Hakodate, Japan, 2006.
12. R. E. Kranton and D. F. Minehart. A theory of buyer-seller networks. *American Economic Review*, 91(3):485–508, 2001.
13. P. Lazarsfeld. Friendship as a social process: A substantive and methodological analysis. *Freedom and Control in Modern Society*, 1954.
14. D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins. Geographic routing in social networks. *Proceedings of the National Academy of Sciences of the USA*, 102(33):11623–11628, 2005.
15. M. Luck, P. McBurney, O. Shehory, and S. Willmott. *Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing)*. AgentLink, 2005.
16. M. Mcpherson, L. S. Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.
17. I. Mecar. Agent-oriented semantic discovery and matchmaking of web services. In *8th International Conference on Telecommunications*, pages 603–607, 2005.
18. M. Moore and T. Suda. A decentralized and self-organizing discovery mechanism. In *Proc. Of the First Annual Symposium on Autonomous Intelligent Networks and Systems*, 2002.
19. K. Nedas. Implementation of munkres-kuhn (hungarian) algorithm, 2005.
20. M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.
21. http://en.wikipedia.org/wiki/Scale−free−network.
22. E. Ogston and S. Vassiliadis. Local distributed agent matchmaking. In *Proceedings of the 9th International Conference on Cooperative Information Systems*, 2001.
23. A. Ouksel, Y. Babad, and T. Tesch. Matchmaking software agents in b2b markets. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04)*, 2004.
24. M. Paolucci, T. Kawamura, T. R. Payne, and K. Sycara. Semantic matching of web services capabilities, 2002.
25. K. Sigdel, K. Bertels, B. Pourebrahimi, S. Vassiliadis, and L. Shuai. A framework for adaptive matchmaking in distributed computing. In *In proceeding of GRID Workshop*, 2005.
26. K. Sycara and M. Klusch. Brokering and matchmaking for coordination of agent societies: A survey. *Coordination of Internet Agents: Models, Technologies and Applications*, pages 197–224, 2001.
27. J. Travers and S. Milgram. An experimental study of the small world problem. *Sociometry*, 32(4):425–443, 1969.
28. E. Del Val and M. Rebollo. A survey on web service discovering and composition. In *Webist*, volume I, pages 135–142, 2008.
29. D. J. Watts. The new science of networks. *Annual Review of Sociology*, 30, 2004.
30. D. J. Watts, P. S. Dodds, and M. E. J. Newman. Identity and search in social networks, May 2002.
31. H. P.Thadakamalla, R. Albert, S. R.. TKumara. Search in spatial scale-free networks *New Journal of Physics, Volume 9, Issue 6, pp. 190 (2007).*
32. S. Xiao and G. Xiao. On degree-based decentralized search in complex networks. *CoRR*, abs/cs/0610173, 2006.