

Self-Organizing Referral Networks: A Process View of Trust and Authority

Pinar Yolum
pyolum@eos.ncsu.edu

Munindar P. Singh
singh@ncsu.edu

Department of Computer Science
North Carolina State University
Raleigh, NC 27695-7535, USA

ABSTRACT

Agents must decide with whom to interact, which is nontrivial when no central directories are available. We are developing a decentralized approach to trust based on referral systems, where agents adaptively give referrals to one another. Interestingly, referral systems provide us with a useful and intuitive model of how links may be generated: a referral corresponds to a customized link generated on demand by one agent for another. This gives us a basis for studying the processes underlying trust and authority, especially as they affect the structure of the evolving social network of agents. We explore key relationships between the policies and representations of the individual agents on the one hand and the aggregate structure of their social network on the other.

1. INTRODUCTION

This paper considers two problems, a practical and a theoretical, and comes up with an interesting interrelationship between them. The practical challenge is how to engender trust in decentralized settings consisting of autonomous parties. In such settings, we cannot rely on a centrally trusted authority. In fact, current approaches based on certificate authorities are fundamentally impoverished because they do not take into account actual evidence about the behavior of a given party, i.e., whether it has been observed to be trustworthy or not by the parties that have interacted with it. Reputation mechanisms, such as the one on eBay, address the question of evidence directly, but they are still centralized. They assume that accurate ratings can be assumed and effectively aggregated. Such ratings are typically just scalars (possibly augmented with free text comments). However, trust is often multidimensional. Moreover, the best ratings are those that are obtained privately and from parties that you know personally. We take this idea and formalize it into the notion of referrals, where a referral can be viewed as an endorsement.

On the theoretical side, studies of Web structure have also assumed the links from one to page to another indicate some sort of an endorsement. This assumption leads to the heuristic that a met-

ric such as the PageRank of a page measures its authoritativeness [Brin and Page, 1998; Henzinger, 2001]. Informally, a Web page has a high PageRank only if it is pointed to by Web pages with high PageRanks, i.e., if other authoritative pages view this page as authoritative. Current work on Web structure generally mines the Web and then studies graphs induced by links among Web pages. It offers statistical models of the Web and its evolution, e.g., [Barabási et al., 2000; Pennock et al., 2002], but doesn't capture the reasoning by each creator of a link, through which a link can be taken to be an endorsement.

Fortuitously, our referral-based approach for finding trustworthy service providers gives us a direct means to model the evolution of a "social" network of agents. We can consider the various representations and strategies or policies used by the agents along with distributions of true authoritativeness and relate these to each other. Our study of the processes of linkage and referrals offers some key benefits:

- It applies in helping determine trust in dynamic settings.
- It gives us a basis for understanding how trust and authoritativeness emerge, potentially leading to mechanisms that cannot be easily violated.
- The above mechanisms can be used to develop measures of authoritativeness and trust that apply to settings such as the Deep Web, where Web sites respond to queries, but do not carry statically linked pages.

Our study is conducted as a series of simulation experiments where we can control the policies of the agents. These experiments consider up to several hundred agents. Although this is far from the size of the Web, the simulations can still offer some understanding of the concepts and processes involved. The immediate practical applications will relate to enterprise knowledge management and virtual communities, where sizes of a few hundred to a few thousand are common.

Organization. Section 2 gives additional details on our model of referrals among autonomous agents, possible applications domains, and our experimental setup. Section 3 studies factors that affect the distribution of PageRanks. Section 4 studies the ways PageRank distributions can evolve based on neighbor selection policies. Section 5 discusses the relevant literature and motivates directions for further work.

Please note that Section 2 is the technical background and is largely based on our previous papers [Yolum and Singh, 2003]. The new results of this paper are given in Section 3 and Section 4 and are previously unpublished.

2. TECHNICAL FRAMEWORK

We consider multiagent systems consisting of autonomous agents. Our basic model is intended to mirror interactions in real life. For this purpose, imagine if you will that our agent represent *principals* who could be people or businesses providing and consuming *services*. The services are construed abstractly, i.e., not limited to current Web services standards. Specifically, the services could involve serving static pages, processing queries, or carrying out e-commerce transactions, but their details are not represented in this paper. Our study will concentrate on agents and services.

The following aspects of real systems are relevant here:

- The agents offer varying levels of trustworthiness and are interested in finding other trustworthy agents. They track each other's trustworthiness. The agents can judge the quality of a service obtained and adaptively select their neighbors in order to improve their local performance. Importantly, by giving and taking *referrals*, agents can cooperate with one another to find trustworthy agents with whom to interact. An agent may provide a service to another agent or give a referral that leads it in the right direction. Notice that trust applies both to the ultimate service provider and to the agents who contribute to referrals to that provider.
- The agents are autonomous. That is, an agent may or may not respond to another agent by providing a service or a referral. When an agent does respond, there are no guarantees about the quality of the service or the suitability of a referral. Likewise, we do not assume that any agent should necessarily be trusted by others: an agent unilaterally decides how to rate another principal.

The above model addresses the important challenge of finding trustworthy agents, which is nontrivial in open systems. One, referrals can apply even in the absence of centralized authorities and even when regulations may not ensure that services are of a suitable quality. Two, because service needs are often context-sensitive, a response from an agent can potentially benefit from the knowledge that the agent has of the other's needs.

The agents act in accordance with the following abstract protocol. An agent begins to look for a trustworthy provider for a specified service. The agent queries some other agents from among its *neighbors*. A queried agent may offer to provide the specified service or, based on its *referral policy*, may give referrals to other agents. The querying agent may accept a service offer, if any, and may pursue referrals, if any. Each agent maintains models of its acquaintances, which describe their *expertise* (i.e., quality of the services they provide) and *sociability* (i.e., quality of the referrals they provide). Both of these elements are learned based on service ratings from its principal. Using these models, an agent applies its *neighbor selection policy* to decide on which of its acquaintances to keep as neighbors. Key factors include the quality of the service received from a given provider, and the resulting value that can be placed on a series of referrals that led to that provider. In other words, the referring agents are rated as well. An agent's own requests go to some of its neighbors. Likewise, an agent's referrals in response to requests by others are also given to some of its neighbors, if any match. This, in a nutshell, is our basic social mechanism.

Together, the neighborhood relations among the agents induce the structure of the given society. In general, as described above, the structure is adapted through the decisions of the different agents. Although the decisions are autonomous, they are influenced by various policies.

2.1 Applicable Domains

The above framework enables us to represent different application domains naturally. Two important domains are commerce and knowledge management, which differ in their notions of service and how the participants interact.

In a typical commerce setting, the service providers are distinct from the service consumers. The service consumers lack the expertise in the services that they consume and their expertise doesn't get any better over time. However, the consumers are able to judge the quality of the services provided by others. For example, you might be a consumer for auto-repair services and never learn enough to provide such a service yourself, yet you would be competent to judge if an auto mechanic did his job well. Similarly, the consumers can generate difficult queries without having high expertise. For example, a consumer can request a complicated auto-repair service without having knowledge of the domain.

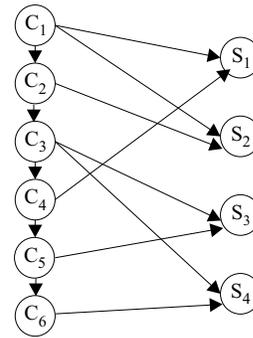


Figure 1: A schematic configuration for e-commerce

By contrast, in knowledge management, the idea of “consuming” knowledge services would correspond to acquiring expertise in a given domain. A consumer might lack the ability to evaluate the knowledge provided by someone who has greater expertise. However, agents would improve their knowledge by asking questions; thus their expertise would increase over time. Following the same intuition, the questions an agent generates would also depend on its expertise to ensure that the agent doesn't ask a question whose answer it already knows.

Figure 1 is an example configuration of service consumers and providers that corresponds to a commerce setting. The nodes labeled C denote service consumers and the nodes labeled S denote service providers. Consumers are connected to each other as well as to the service providers. These links are essentially paths that lead to service providers with different expertise. In this model, the service providers are dead ends: they don't have outgoing edges, because they don't initiate queries or give referrals. Thus, their sociability stays low. Their true and modeled expertise may of course be high.

2.2 Evaluation Metrics

The relevant global properties of the system that we study here are formally characterized by some metrics, usually involving vector operations.

Capability. The *capability* of an agent for a query measures how similar and how strong the expertise of the agent is for the query [Singh et al., 2001]. Capability resembles cosine similarity but also takes into account the magnitude of the expertise vector. What this means is that expertise vectors with greater magnitude turn out to be more capable for the query vector. In Formula 1, $Q((q_1 \dots q_n))$ refers to a query vector, $E((e_1 \dots e_n))$ refers to an expertise vector and n is the number of dimensions these vectors have.

$$Q \otimes E = \frac{\sum_{t=1}^n (q_t e_t)}{\sqrt{n \sum_{t=1}^n q_t^2}} \quad (1)$$

PageRank. PageRank is a metric used by Google to rank Web pages that are returned for a query [Brin and Page, 1998]. The PageRank of a Web page measures its authoritativeness. Informally, a Web page has a high PageRank only if it is pointed to by Web pages with high PageRanks, i.e., if other authoritative pages view this page as authoritative. We use the same metric to measure the authoritativeness of agents. The PageRank of an agent is calculated using Equation 2, where $P(i)$ denotes the PageRank of agent i , I_i denotes agents that have i as a neighbor, and N_j denotes the agents that are neighbors of j . In addition to accumulating PageRanks from incoming edges, each node is assumed to get a minimum PageRank of $(1 - d)$, where d is taken to be 0.85 as in the original paper [Brin and Page, 1998]. The calculated PageRanks are not normalized to demonstrate the variance in maximum PageRanks in different setups.

$$P(i) = d \sum_{j \in I_i} \frac{P(j)}{N_j} + (1 - d) \quad (2)$$

2.3 Agent Algorithms

We have implemented a distributed platform using which adaptive referral systems for different applications can be built. However, we investigate the properties of interest over a simulation, which gives us the necessary controls to adjust various policies and parameters.

Consumers have high *interest* in getting different types of services, but they have low expertise, since they don't offer services themselves. Providers have high expertise but low sociability. The interests and expertise of the agents are represented as term vectors from the vector space model (VSM) [Salton and McGill, 1983], each term corresponding to a different domain. The simulation uses these to generate queries and answers for the various agents.

Algorithm 1 Ask-Query()

```

1: Generate query
2: Send query to matching neighbors
3: while (!timeout) do
4:   Receive message
5:   if (message.type == referral) then
6:     Send query to referred agent
7:   else
8:     Add answer to answerset
9:   end if
10: end while
11: for  $i = 1$  to  $|answerset|$  do
12:   Evaluate answer( $i$ )
13:   Update agent models
14: end for

```

Each agent is initialized with the same model for each neighbor; this initial model encourages the agents to both query and generate referrals to their neighbors. An agent that is generating a query follows Algorithm 1. An agent generates a query by slightly perturbing its interest vector, which denotes that the agent asks a question similar to its interests (line 1). Next, the agent sends the query to a subset of its neighbors (line 2). The main idea here is to determine which of its neighbors would be likely to answer the query. We usually determine this through the capability metric.

An agent that receives a query acts in accordance with Algorithm 2. An agent answers a question if its expertise matches a question. If the expertise matches the question, then the answer is the perturbed expertise vector of the agent. When an agent does not answer a question, it uses its *referral policy* to choose some of its neighbors to refer.

Algorithm 2 Answer-Query()

```

1: if hasEnoughExpertise then
2:   Generate answer
3: else
4:   Refer neighbors
5: end if

```

Back in Algorithm 1, if an agent receives a referral to another agent, it sends its query to the referred agent (line 6). After an agent receives an answer, it evaluates the answer by computing how much the answer matches the query (line 12). Thus, implicitly, the agents with high expertise end up giving the correct answers. After the answers are evaluated, the agent uses its *learning policy* to update the models of its neighbors (line 13). In the default learning policy, when a good answer comes in, the modeled expertise of the answering agent and the sociability of the agents that helped locate the answerer (through referrals) are increased. Similarly, when a bad answer comes in, these values are decreased. At certain intervals during the simulation, each agent has a chance to choose new neighbors from among its acquaintances based on its *neighbor selection policy*. Usually the number of neighbors is limited, so if an agent adds some neighbors it might have to drop some neighbors as well.

3. DISTRIBUTION OF PAGERANKS

It has been widely accepted that the in-degree and out-degree distributions on the Web follow a power law distribution [Faloutsos et al., 1999; Huberman, 2001]. That is, the number of pages with k incoming links is inversely proportional to k^m ; Barabási *et al.* estimate that $m = 2.1$ [2000]. Recently, Pandurangan *et al.* [2002] showed that the distribution of the PageRanks on the Web follows a power-law as well. Here, we study four factors that influence the PageRank distributions.

The experiments we report below contain 400 agents, where each agent is neighbors with four other agents. The initial neighbors are picked at random. All the populations contain 400 agents. Each service provider has a high expertise in one domain. The remaining agents are service consumers. The interests of the service consumers can span multiple domains. We tune the simulation so that an agent answers a query only when it is sure of the answer. This ensures that only the providers answer any questions, and the consumers generate referrals to find the providers.

The four factors we study are the percentage of actual experts in the network, the referral policies that the agents follow, adaptability of the agents, and the neighbor selection policies they follow.

3.1 Percentage of Experts

Intuitively, the percentage of agents with high expertise plays a crucial role in the distribution of PageRanks. For example, when there are too many experts in the system, we expect that the PageRanks will tend to be shared among them. Having a small number of experts may ensure that experts with high authoritativeness will emerge. To study this point, we vary the percentage of the experts in the system. We study three populations with 5%, 10%, and 20% experts in them.

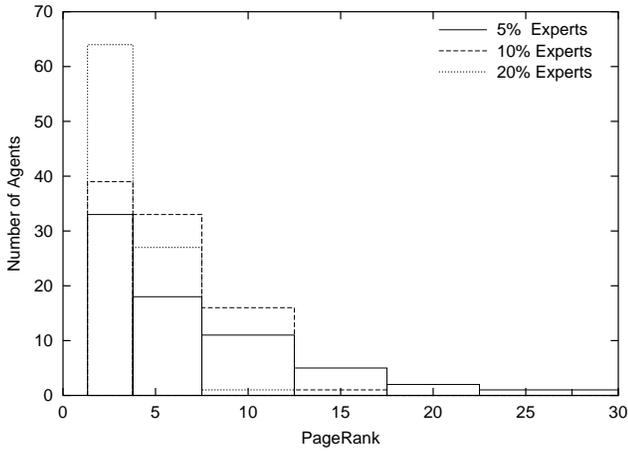


Figure 2: PageRank distributions for percentage of experts

Figure 2 shows a histogram of PageRank distribution for three populations for PageRank values 2.5 and higher. The solid lines denote the population with 5% experts, the dashed lines denote the population with 10% percent experts, and the dotted lines denote the population with 20% experts. When the percentage of experts is high, the PageRanks are clustered for small PageRank values. For example, when the population has 20% experts, the number of agents having PageRank greater than 2.5 is higher than for the other two populations. For the greater values of the PageRank, the converse holds. For example, the only population that allows PageRank greater than 25 is the 5% expert population.

An intuitive explanation for this phenomenon is the implicit competition among the experts. When there are too many of them, they end up sharing the incoming edges thereby only a few reach relatively high PageRanks. When there are a few experts, those experts tend to dominate more clearly. Since the population with 5% percent experts provide a broader distribution of PageRanks, we use this population for the following experiments.

3.2 Referral Policies

A referral policy specifies to whom to refer. We consider some important referral policies. Based on the referral policy, each query results in a different number of agents being contacted. We limit the length of the referral chains to three—similar to Gnutella’s time-to-live value.

1. *Refer all neighbors.* Agents refer all of their neighbors. This policy resembles Gnutella’s search process where each server forwards an incoming query to all of its neighbors if it doesn’t already have the requested file [Kan, 2001].
2. *Refer all matching neighbors.* The referring agent calculates how capable each neighbor will be in answering the given query (based on the neighbor’s modeled expertise). Only neighbors scoring above a given capability threshold are referred.
3. *Refer the best neighbor:* Refer the best matching neighbor. This is similar to Freenet’s routing of request messages, where each Freenet client forwards the request to an agent that is the likeliest to have the requested information [Langley, 2001].

Some agents are identified as authoritative as a result of their being chosen as neighbors by other authoritative agents. Presumably,

authoritative agents are the most desirable to interact with. In general, agents with high expertise or high sociability would be candidates for becoming authorities. We measure the authoritativeness of each agent using the PageRank metric (Equation 2) and study the effect of referral policies in the emergence of authorities.

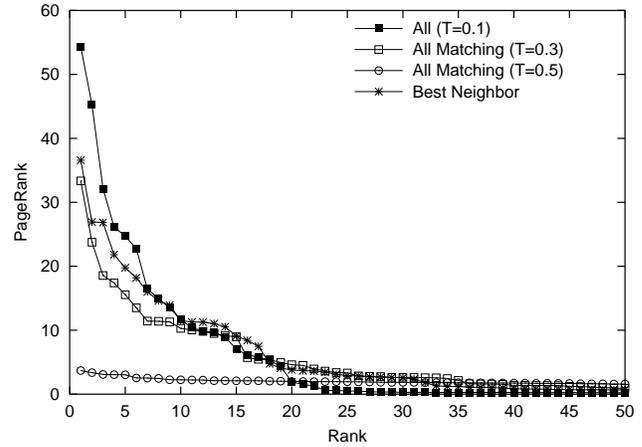


Figure 3: PageRank distributions for referral policies

After each simulation run, the agents are ranked based on their final PageRank. Figure 3 shows the PageRank distribution of the top 50 agents (out of a total of 400). If the agents use the *Refer all* policy, few authorities with high PageRanks emerge. For example, the 10th agent in the *Refer all* policy gets a PageRank greater than the first agent in two instances of the *Refer all matching* policy (with thresholds 0.4 and 0.5). Further, the *Refer all* policy creates a large variance among the PageRank distributions. For example, while the first agent gets a PageRank of 54.00, the 50th agent gets a PageRank of only 0.23.

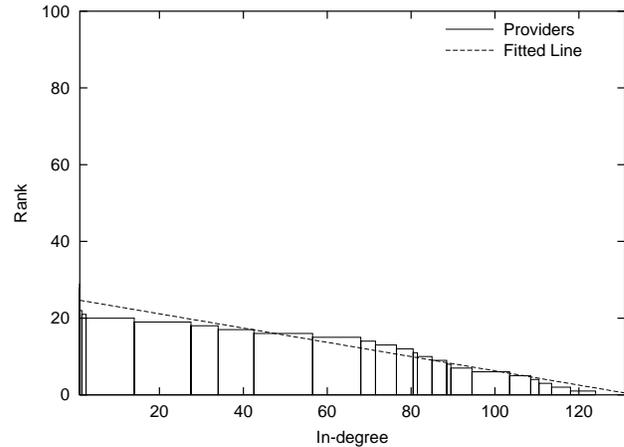


Figure 4: In-Degree vs rank (Policy = All)

Contrast this with *Refer all matching* policy with a threshold of 0.5, where the first agent gets a PageRank of 3.68 whereas the 50th agent gets a PageRank of 1.58. The distribution of PageRanks using the *Best neighbor* policy falls between the distributions for *Refer all* and *Refer all matching* with high thresholds. In other words, when agents use the *Best neighbor* policy, the highest PageRank is not as high as the *Refer all* policy (36.00) but the PageRank variance between the first and the 50th agent is still quite large.

Intuitively, the explanation for the above is that the *Refer all* policy is highly effective in disseminating information about the experts. With *Refer all* policy, the consumers locate the providers easily. When this policy is used, most consumers have in-degree zero, and the possible in-degrees are shared among the providers with a linear distribution. Figure 4 shows the in-degree distribution after this occurs. Hence, agents can easily recognize these providers' authoritativeness, thereby yielding high PageRanks for some of them.

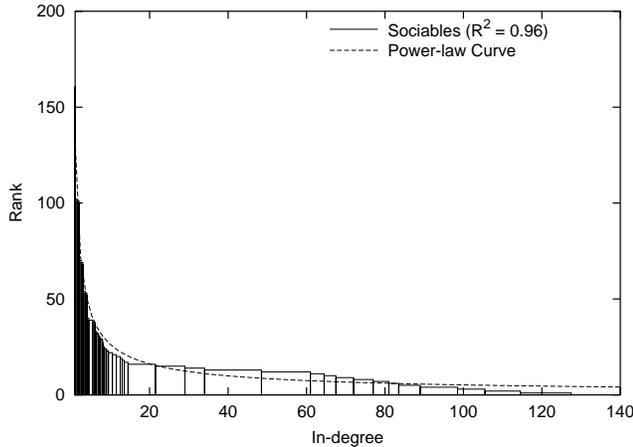


Figure 5: In-degree vs rank (Policy = All matching)

Conversely, when *Refer all matching* policy is used, the in-degrees of the agents are distributed as a power-law as shown in Figure 5. Whereas few agents have high in-degree, many agents have small in-degree. This differs from the previous in-degree distribution, where agents who had non-zero in-degree all had high in-degree. As a result of this, the agents who have smaller in-degree can also gain PageRanks. This explains why with *Refer all* few agents have high PageRanks, while with *Refer matching* more agents share the PageRanks.

3.3 Adaptability

At certain intervals during the simulation, each agent has a chance to choose new neighbors from among its acquaintances. Since agents learn about other agents through the answers they receive, changing neighbors allow them to point at agents that are expected to be more useful to them. This allows us to study the evolution of PageRanks, since PageRanks of the agents change as a result of the neighbor changes.

Figure 6 plots the distribution of PageRanks after several neighbor changes. The plots correspond to the number of agents that get a PageRank higher than the value given on the X axis. To reduce clutter, we have omitted the PageRanks smaller than 1. After the first neighbor change, most agents have a PageRank between 1 and 3. As the agents change their neighbors and point to the ones that are more useful to them, some authorities emerge. This is reflected in the graph by the increased number of agents that get a higher PageRank. After the final neighbor change, there are agents with PageRanks greater than 25. In other words, the agents do learn about the authorities and show a preference for linking to them.

3.4 Neighbor Selection Policies

Recall that each agent chooses its neighbors based on local information only, without knowing which neighbors other agents are choosing. A neighbor selection policy governs how neighbors are

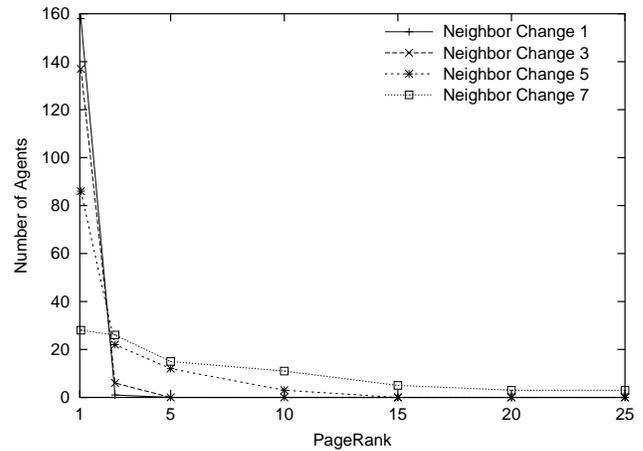


Figure 6: Change in PageRank distribution as agents adapt

added and dropped. Such policies can strongly influence the formation of authorities.

To evaluate how the neighbor selection policies affect the PageRank distribution, we compare three policies using which an agent selects the best m of its acquaintances to become its neighbors.

- *Providers*. Sort acquaintances by how their expertise matches the agent's interests.
- *Sociables*. Sort acquaintances in terms of sociability.
- *Weighted average*. Sort acquaintances in terms of a weighted average of sociability and how their expertise matches the agent's interests.

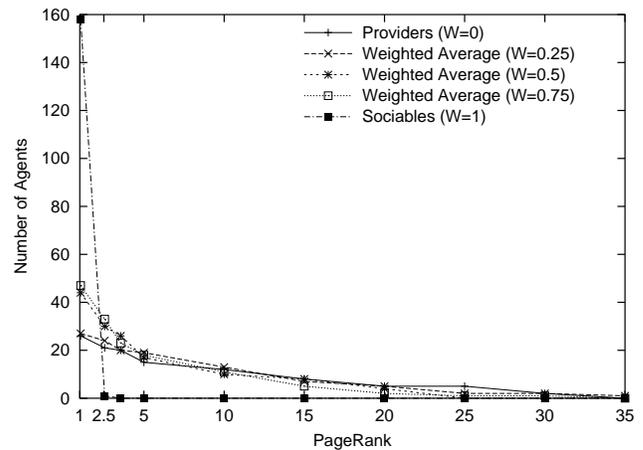


Figure 7: PageRank distributions for neighbor selection policies

Figure 7 plots the distribution of PageRanks with respect to some neighbor selection policies. Again, the X axis shows PageRanks and the Y axis denotes the number of agents that get a PageRank greater than the PageRank shown on the X axis. The five plots correspond to *Providers*, *Sociables*, and three *Weighted average* neighbor selection policies with different weights. W denotes the weight of the sociability in choosing a neighbor. When W is set to 0, the *Providers* policy, and when W is set to 1, the *Sociables* policy is

in effect. Other values of W measure weighted averages of the sociability and expertise.

All curves, except the one for *Sociables* policy, are similar to each other. In all four cases, only a few number of authorities emerge. But, the level of their authoritativeness is high. For example, for the *Providers* policy, while only 26 agents get a PageRank above 1, five of them get a PageRank above 20. Increasing the effect of the sociability slightly increases the number of agents with medium authority while slightly decreasing the number of agents with high authority. For example, with *Weighted Average* policy, when the sociability and the expertise are weighted equally, the number of agents that get a PageRank above 1 is 44, while four of them get a PageRank above 20.

The *Sociables* policy does not follow this distribution. Initially, when not too many experts have been discovered, choosing neighbors only based on sociability does not help agents find service providers. Hence, when agents follow the *Sociables* policy in the beginning, most agents get average PageRanks (e.g., 158 agents get a PageRank around 1). However, once the PageRanks distribution settles, following *Sociables* policy instead of *Providers* policy determines how the PageRank distribution evolves.

In other words, for strong authorities to emerge, it is important that the agents put a high value on the ability to produce high quality of service. If at the outset, the agents prefer sociables, there is little grounding in quality, it is difficult to find good providers and strong authorities do not emerge. However, once the network has stabilized, sociability helps as there is a basis for good referrals to be given and there is value in those who can give good referrals.

4. EVOLUTION OF PAGERANK DISTRIBUTIONS UNDER PERTURBATION

The factors mentioned in the previous section influence the PageRank distributions. As expected, the agents that get the most PageRanks are agents with high expertise or high sociability. Now, we manually modify the models of a few of the agents in the system to measure the consequences of the perturbation (vis à vis the existing PageRanks, which characterize the entrenched players) on the resulting population.

For example, what happens when an agent with a high expertise start providing poor quality of service? Conversely, if an agent gains expertise over time, can it acquire high PageRanks and get a high ranking?

First, we run the simulation until the neighborhood graph stabilizes, i.e., there are no more neighbor changes. Then, we pick four agents that have ranked in the top 25 and switch their expertise values with four agents with low expertise. This will ensure that the four previously expert agents will start giving low quality service. Rather than choosing the top four agents, we choose the four agents uniformly from the top 25 (1st, 4th, 16th, and 24th). This will allow us to see the impact of the initial position in the rankings.

We are interested in how the PageRank distribution and the ranking of these eight agents will change. Intuitively, we would expect the PageRanks to drop for the previously expert agents and increase for the agents who have now become experts. To give a more complete analysis, below we look at two orthogonal setups. In the first setup, agents use a neighbor selection policy that favors sociability, whereas in the second setup the agents use a neighbor selection policy that favors expertise.

4.1 Agents Prefer Sociables

We first study the four agents that have started giving low qual-

ity of service. Figure 8 plots the evolution of PageRanks for two previous experts, Agent 1 and Agent 24. Interestingly, when agents prefer to be neighbors with sociable agents, the PageRanks of the two agents (4th and 24th) get a lot more affected than the other two agents. More precisely, the PageRanks as well as the rankings of these two agents drop.

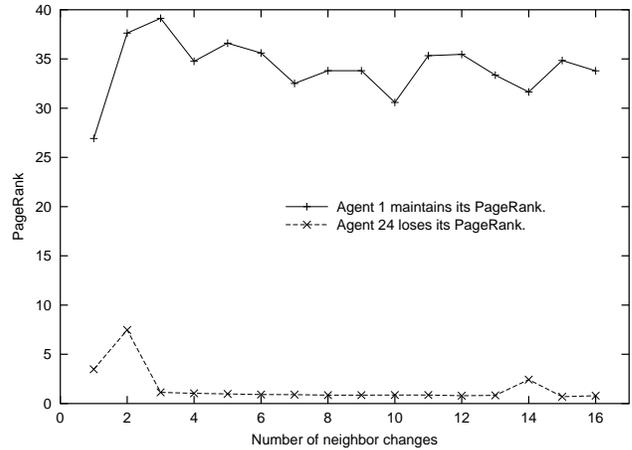


Figure 8: Evolution of PageRanks for two previous experts

The agent that has ranked 4th is now ranked at 18 and the agent that was ranked 24th is now ranked at 48. However, the other two agents (1 and 16) do not get affected by the fact that they have started offering lower quality of service. Both their PageRank and ranking stay almost the same with minor shifts between neighbor changes. This is unexpected. Looking closely at the agents that model these, we see that even though agents 1 and 16 have low expertise, they are still being kept as neighbors for their sociability. In other words, now that agents 1 and 16 cannot provide adequate answers, they start giving good referrals. This is enough for other agents pointing at these to keep them as neighbors. In other words, agent 1 and 16 keep their PageRanks because of their neighbors. On the other hand, since agents 4 and 24 do not have as useful neighbors, there is no use for other agents to point at them. What we see here is that when agents gain enough PageRank, they can maintain their ranking even after they lose their expertise as long as they maintain useful neighbors.

Meanwhile, the four agents who have now started offering good services can increase their PageRanks only slightly; the greatest increase was of PageRank 1. So, even though these agents have high expertise now, since the other agents care for sociability more, these four agents are not being picked up as neighbors as much as expected.

4.2 Agents Prefer Experts

The previous section explained how the PageRank distribution can evolve when the agents prefer sociable neighbors. In this section, we look at the converse case: how does the distribution evolve if the agents choose experts over sociables?

Again, we look at the four agents who have stopped providing high quality of service. This time, all of their PageRanks, and hence their ranking, drop slowly. The interesting cases occur for the other four agents who have now started providing high quality of service. Figure 9 plots the evolution of PageRanks for two of the agents.

Agent 237 cannot improve their PageRank at all. The reason for this is that only a few other agents point at it. Therefore, it is not referred to other agents. Over all, even though this agents has

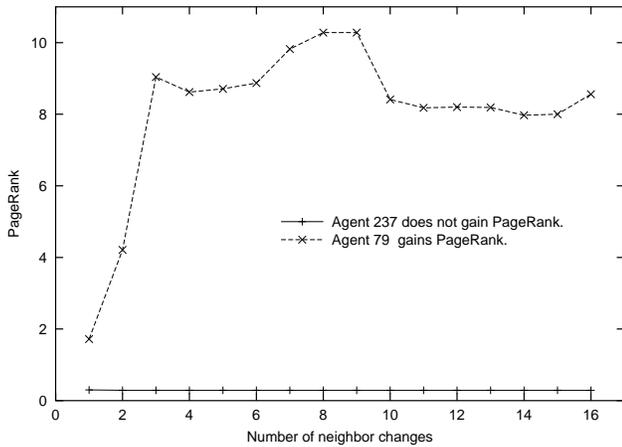


Figure 9: Evolution of PageRanks for two new experts

higher expertise than most of the agents that rank above it, it cannot get enough PageRank to improve its ranking.

This is not the case for Agent 79. It gets discovered by other agents and hence can substantially increase its PageRank. However, there is a limit to this increase. That is, even though it accumulates a high PageRank, it cannot make it to the top ten. This is caused by the stability of the other agents' neighbors. Most other agents have already found good service providers for their needs, so they rarely change neighbors.

4.3 Does the Winner Take All?

As noted above, the distributions of in-degrees, out-degrees, and PageRanks all follow power law distributions. These distributions suggest some sort of *preferential attachment* wherein Web links are likelier to be made to those pages that already have a high in-degree or PageRank. Hence, the mottos “the rich get richer” and the “winner takes all” used to describe this phenomenon. However, Pennock *et al.* recently discovered that the reality is closer to a combination of preferential and uniform attachment, meaning that the winner doesn't always take all [2002].

Please note that the traditional Web approaches seek to understand and model the development of the Web without regard to an understanding of why links emerge. Also, these are limited to static pages where the links are generated by hand and any semantics is hidden in the surrounding text. By contrast, we are interested in and able to model the *process* through which the links among agents emerge. When the agents contact others or generate referrals, they do so with respect to specific service requests. Further, we can explicitly model the various policies through which the referrals are generated and through which the agents update their sets of neighbors. This enables us to capture various shades of trust formally.

Specifically, from the above experiments, we conclude that authorities with high PageRanks emerge when (1) there are few experts in the system, (2) agents exchange more referrals, and (3) agents change neighbors based on other agents' expertise and sociability. The PageRank distribution then evolves based on how these authoritative agents keep up with their quality of service. This evolution is highly influenced by whether agents prefer sociables or experts. When agents prefer sociables, agents who have gained high PageRanks can sometimes keep their PageRank even without providing high quality of service. On the other hand, when experts are chosen over sociables, the agents with high PageRanks have to continue to offer high quality of service.

Similarly, when sociables are preferred, there is little chance for newcomer experts to get high PageRanks since they are not pointed to by any sociable agents. Preference for experts relaxes this case but still does not guarantee high PageRanks for newcomers. These findings lead us to conjecture more broadly that in environments where sociability dominates (with respect to expertise), the winner will take all, whereas in settings where expertise dominates, the winner may not take all. If new services emerge and draw some of the demand, then there will be greater churn in the rankings of the top most authorities.

5. DISCUSSION

Self-organization has been influential in understanding the macro behavior of many complex systems [Axelrod and Cohen, 2000]. Perhaps the most studied case is self-organizing ant colonies, where ants communicate through signs to discover the shortest routes to food supplies [Dorigo *et al.*, 1996; Parunak and Brueckner, 2001]. Johnson analyzes the similarities between self-organization of ant colonies and human societies and—to some extent—World Wide Web [Johnson, 2002, p. 226]. Our approach enables us to study the emergence of link structure of the Web as agents give one another referrals. Because the agents are adaptive, they learn about others through prior interactions and autonomously exchange referrals and choose who they want to interact with. The link structure evolves based on agents' preferences in neighbor choices (i.e., expertise or sociability). The dynamic nature of the link structure helps us study the emergence of authorities, rather than merely identify the authorities on a static graph. Below, we discuss some related approaches.

Referrals have been used in multiagent systems before, but the emergent nature of referral network has not received enough attention. For example, MINDS, which helped users find documents, was an early agent-based referral system [Huhns *et al.*, 1987]. MINDS system allows adaptivity but the self-organization of the nodes have not been studied. Kautz *et al.* study some properties of static graphs that result from the referrals exchanged among agents, but do not have adaptivity and hence have no self-organization [1997].

Yu and Singh develop a method for searching referral networks effectively [2003]. Yu and Singh's method is based on building a weighted referral graph from the referrals, such that the nodes denote agents and the edges denote referrals. Both nodes and edges are coupled with weights. They develop a method to minimize this graph so that only the promising referrals are followed.

Kumar *et al.* develop an approach to infer web communities from the link structure of the Web [1999]. Kumar *et al.* propose that any community structure should contain a bipartite core where the *fans* and *centers* make up the independent sets. Fans and centers are defined recursively, such that fans are pages that point at good centers and centers are pages that are pointed to by good fans. Kumar *et al.*'s approach assumes that if many fans point to the same set of centers, then they are likely to be on the same topic, and hence form a community.

Wang develops an approach for service location based on forming clusters of similar agents [2002]. Each agent reports to one middle agent. Middle agents identify similar agents based on agents' queries and exchange agents so that one middle agent is responsible for agents with similar interests. Assuming that similar agents pose similar queries, the middle agents can serve the agents more efficiently.

Pujol *et al.* calculate the reputation of an agent based on its position in its social network [2002]. The social networks are built based on the link structure induced by the web pages of the users.

An agent gets a high reputation only if the agents that point to it also have high reputation, similar to the notion of authority exploited in search engines such as Google. Pujol *et al.* test their approach to find the reputations of authors where the reputation of an author is defined as the number of citations received. Even though each agent can calculate its own reputation based only on local information (i.e., the agents that point at it), a central server is needed to access others' reputations.

Adamic *et al.* study different local search heuristics that exploit high out-degrees in power-law networks [2001]. For example, one of their heuristics is based on sending the message to the neighbor with the most out-degree, assuming each node is aware of the number of outgoing edges of their neighbors. This is similar to our concept of sociability. In their approach, a peer with high out-degree is chosen because it will allow the message to get to more peers. In our case, rather than maximizing the nodes that will receive the query, we try to send the query only to those who are likely to answer. For this reason, finding a sociable agent is important since the referrals it gives will help locate an expert.

Ng *et al.* study the stability of PageRank and HITS algorithms and propose variants of these algorithms that offer more stability with respect to the addition or deletion of vertices [2001]. They take a graph-theoretic stance and simply recalculate the given metrics. Their results are of course valuable in judging the stability of the different algorithms. By contrast, however, we consider the processes explicitly. Thus we can address the question of how the links will evolve.

Several recent approaches try to improve the current PageRank algorithm. Haveliwala develops an approach where the PageRanks of the pages are influenced by the current query [2002]. Instead of generating one PageRank vector, several PageRank vectors are generated based on a predefined set of topics. Then, the query is mapped to one of the topics and the related PageRank vector is used to compute the rankings. In some respects, our approach exhibits a similar intuition in that neighbors are contacted and referrals are given highly topic-sensitive manner. However, we consider the PageRank computations based on the neighborhood relation—an agent's neighbors are the acquaintances that it values the most all things considered.

Overall, we believe that a process-centric view doesn't detract from the previous studies, but adds additional depth and understanding to how networks evolve in cooperative settings, but under local control. It also suggests how to apply various policies to maximize the quality of a given network of agents. There is an immediate value for the smaller networks, for instance, for knowledge management by locating experts, e.g., [Yu and Singh, 2003], and a longer-term value in understanding the evolving Web at large.

6. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation under grant ITR-0081742.

References

- Lada A. Adamic, Rajan M. Lukose, Amit R. Puniyani, and Bernardo A. Huberman. Search in power-law networks. *Physics Review E*, 64(46135), 2001.
- Robert Axelrod and Michael D. Cohen. *Harnessing Complexity: Organizational Implications of a Scientific Frontier*. Basic Books, New York, NY, 2000.
- Albert-László Barabási, Réka Albert, and Hawoong Jeong. Scale-

free characteristics of random networks: The topology of the World Wide Web. *Physica A*, 281:69–77, 2000.

- Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.
- Marco Dorigo, Vittorio Maniezzo, and Alberto Colomi. The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, 26(1):29–41, 1996.
- Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power law relationships of the Internet topology. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 251–262. ACM, 1999.
- Taher H. Haveliwala. Topic-sensitive PageRank. In *Proceedings of the 11th International World Wide Web Conference (WWW)*, pages 517–526. ACM Press, May 2002.
- Monika R. Henzinger. Hyperlink analysis for the Web. *IEEE Internet Computing*, 5(1):45–50, January 2001.
- Bernardo A. Huberman. *The Laws of the Web: Patterns in the Ecology of Information*. MIT Press, Cambridge, MA, 2001.
- Michael N. Huhns, Uttam Mukhopadhyay, Larry M. Stephens, and Ronald D. Bonnell. DAI for document retrieval: The MINDS project. In Michael N. Huhns, editor, *Distributed Artificial Intelligence*, pages 249–283. Pitman/Morgan Kaufmann, London, 1987.
- Steven Johnson. *Emergence: The Connected Lives of Ants, Brains, Cities, and Software*. Touchstone, New York, NY, 2002.
- Gene Kan. Gnutella. In *Oram [2001]*, chapter 8, pages 94–122. 2001.
- Henry Kautz, Bart Selman, and Mehul Shah. ReferralWeb: Combining social networks and collaborative filtering. *Communications of the ACM*, 40(3):63–65, March 1997.
- Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. Extracting large-scale knowledge bases from the Web. In *Proceedings of the 25th Very Large Databases Conference*, 1999.
- Adam Langley. Freenet. In *Oram [2001]*, chapter 9, pages 123–132. 2001.
- Andrew Y. Ng, Alice X. Zheng, and Michael I. Jordan. Link analysis, eigenvectors and stability. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 903–910, 2001.
- Andy Oram, editor. *Peer-to-Peer: Harnessing the Benefits of a Disruptive Technology*. O'Reilly & Associates, Sebastopol, CA, 2001.
- Gopal Pandurangan, Prabhakar Raghavan, and Eli Upfal. Using PageRank to characterize Web structure. In *8th Annual International Computing and Combinatorics Conference (COCOON)*, 2002.
- H. Van Dyke Parunak and Sven Brueckner. Entropy and self-organization in multi-agent systems. In *Proceedings of the Fifth International Conference on Autonomous Agents (AGENTS)*, pages 124–130. ACM Press, 2001.

- David Pennock, Gary Flake, Steve Lawrence, Eric Glover, and C. Lee Giles. Winners don't take all: Characterizing the competition for links on the Web. *Proceedings of the National Academy of Sciences*, 99(8):5207–5211, April 2002.
- Josep M. Pujol, Ramon Sangüesa, and Jordi Delgado. Extracting reputation in multi agent systems by means of social network topology. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 467–474. ACM Press, July 2002.
- Gerard Salton and Michael J. McGill. *An Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.
- Munindar P. Singh, Bin Yu, and Mahadevan Venkatraman. Community-based service location. *Communications of the ACM*, 44(4):49–54, April 2001.
- Fang Wang. Self-organising communities formed by middle agents. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1333–1339. ACM Press, July 2002.
- Pınar Yolum and Munindar P. Singh. Emergent properties of referral systems. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. ACM Press, July 2003. To appear.
- Bin Yu and Munindar P. Singh. Searching social networks. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. ACM Press, July 2003. To appear.