Carleton University

School of Computer Science

Honours Project (COMP 4905)

Relational Concept Knowledge in Referral Networks

Shaun McQuaker
Dr. Tony White,
School of Computer Science
April 8th, 2005

Abstract

This paper deals with the topic of peer-to-peer referral systems and the policies that allow for the emergence of efficient retrieval of requested information. In an agent based peer-to-peer network, member agents are capable of giving and following referrals to each other. This results in the emergence of communities where agents neighbour with other agents that supply the required service or will refer the right source. This paper is based on the research presented in Munindar P. Singh's paper entitled "Emergent Properties of Referral Systems" [Singh and Yolum, 2003a]. A simulation program for a referral network is discussed and the additional topic of propagating relational concept knowledge in the network to enhance search queries is investigated. Agents with relational concept knowledge can supply suggestions to a querying agent on how to adapt their interests, and thus allowing agents to search in a new manner so as to return better results.

Acknowledgements

I would like to thank Dr. Tony White for the idea for this project and his continued help and suggestions throughout its development. The support of family and friends was also extremely helpful.

Table of Contents

List of Figures and Equations

1. Social Networks

In day-to-day life, social networks within humans are created and changed every moment. They consist of a group of people who know each other and have formed some form of a model that enables them to interact with each other in unique ways. As an example, a person certainly interacts with a spouse in a different fashion than a friend or stranger. Within the virtual world these sorts of communities can emerge as well. On the Internet communities or groups can aide us in identifying important topics or sites. So-called experts within these groups can give us useful information on how to complete a task we may be faced with.

In the real world, when we face a problem we cannot solve ourselves we usually turn to a trusted source or expert for advice on how to proceed with solving the problem at hand. If we don’t know of such a person, we usually ask our friends if they know such a person, and could perhaps provide us with some initial contact. This simple idea can be computationally modeled in what is called a referral network.

1.1 Referral Networks

A referral network is a “de-centralized agent-based approach for service location, where agents provide and consume services, and also co-operate with each other by giving referrals to other agents” [Singh and Yolum, 2003b]. This models how humans interact in a social setting, when asking or answering a question. If they know the answer, they provide one, if they don’t they usually mention someone who may be knowledgeable on the subject matter or they simply do not answer at all.

1.1.1 Framework

In a referral network an agent’s purpose is to find a provider or expert (the words are used interchangeably within this paper) for a specified service. This can also be thought of as the agent attempting to find an answer to a question. An agent attempts to find a provider or an answer to a question by querying its neighbouring agents. The neighbouring agents may answer directly if they have enough expertise for the question at hand. However they may not be experts in the domain that is being requested of them so they will not answer, but instead, in accordance with their *referral policy* decide to give a referral to another agent who may be more knowledgeable in the given domain. A referral policy is a directive on how to refer an agent to another agent, when an answer cannot be given directly [Singh and Yolum, 2003a]. Each agent maintains a model of its environment, which details the expertise (the quality of service they provide) and the sociability (the quality of referrals they give) of agents who are neighbours (directly connected). After a certain amount of time querying, an agent may decide, based on its *neighbour selection policy*, to change its neighbours. For instance I may want to have someone as my neighbour who is an expert in the area that I have been questioning or I may want to have a sociable neighbour who gives very good referrals to experts or providers in the network. Intuitively a neighbour selection policy is a directive on who to keep as neighbours (or who to change to) and is weighted based upon whether you want sociable agents or expert agents as neighbours.

1.1.2 Applicable Domains

Agent based referral networks best simulate the interaction of consumers and providers. One important domain of referral networks is knowledge management. Here experts (or providers) are knowledgeable in a certain domain, i.e. the theory of relativity. A consumer is someone who has low expertise in a certain domain but may have interest in that domain and by querying experts over time may gain enough expertise in that domain to in turn become an expert. The higher a consumer's interest in a domain the more likely they are to achieve an answer in that domain from an expert, and in turn the more they learn.

1.1.2 Modeling Interest and Expertise

Expertise and Interest within a referral network is represented using the Vector Space Model. Traditionally the Vector Space Model (VSM) is used to index the content words of a document in an m-dimensional vector, where m is the number of content words and the length of m^{th} component of the vector is associated with the term (word) frequency. This vector can then be compared against other vectors, thus enabling search functionality (Google employs this method for keyword searching). For knowledge representation in a domain, each term or dimension in an interest or expertise vector represents the amount of expertise or interest an agent has in that domain, normalized between 0.0 and 1.0, where 1.0 represents a high interest or expertise in a domain and 0.0 represents a low interest or expertise in a domain.

1.1.3 Evaluation Architecture

In order for a referral network to function the comparison of interest and expertise vectors is of great importance. There are three main evaluation metrics used.

Similarity

When an agent is about to make a query it must evaluate how similar the query is to the modeled expertise of its neighbours so as to not ask a completely irrelevant question to one its neighbours. This is similar to how human social networks function, for instance we don't ask a chef how to solve differential equations. Similarity is computed as follows [Singh and Yolum, 2003a]:

$$I \oplus J = \frac{e^{-\|I-J\|^2} - e^{-n}}{1 - e^{-n}}$$

EQUATION 1: Similarity

Measured this way similarity is commutative, measures the Euclidean distance between two vectors and is normalized it to obtain a result between 0.0 and 1.0. Both vectors J and I are of the same length (n in this case).

Capability

A measurement of how good the expertise of an agent is for a given query is needed. This allows an agent to reply with a certain degree of surety or else give a referral. Capability resembles cosine similarity but also takes into account the magnitude of the vector [Singh and Yolum, 2003a]. This means that agents with more expertise will have a greater capability of answering the query. Capability is computed as follows [Singh and Yolum, 2003a]:

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

EQUATION 2: Capability

In Equation 2, Q is a query vector of length n [q_1, q_2, ... , q_n] and E is the expertise vector [e_1, e_2, ... , e_n].

PageRank

A ranking of agents within the referral network is used to measure the authoritativeness of agents and is computed using the PageRank metric. This metric is used by Google to rank web pages returned from a query. In its classical setting of ranking web pages, PageRank is a measurement of how important a webpage is based on the links to it from other web pages on the Internet. In the setting of a Referral Network it is a measure of how authoritative an agent is and uses the number of other agents that have chosen the given agent as a neighbour. The PageRank of agents is computed as follows [Singh and Yolum, 2003a]:

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

EQUATION 3: PageRank

In Equation 3, *P(i)* denotes the PageRank of agent *i*, I_i is the set of agents that have *i* as a neighbour, and N_j is the set of agents that are neighbours of *j*. The PageRanks are normalized using the constant d, which is taken to be 0.85, so that the normalized sum of all the agent's PageRanks will be one. [Rogers, 2002]

Now that methods to quantitatively measure a referral network are in place, the next section deals with how a simulation of a referral network is put into place and executed.

2. Simulating a Referral Network

Simulating a referral network involves the four main execution steps:

1. Setup Referral Network Policies.
2. Setup Referral Network Variables.
3. Initialize a population of agents and randomly select neighbours.
4. Run a simulation.
    - Send Queries
    - Receive Referrals
    - Receive Answers

These four main execution steps are subsequently explained.

2.1 Referral Network Policies

The first step in setting up a referral network is defining policies, which are utilized during several key points in the execution of a referral network. These policies give direction for the following actions: asking, answering, referring, learning, changing neighbours, and querying.

2.1.1 Asking Policy

The Asking Policy in a referral network defines an interface between an agent and its policy for making decisions regarding which questions it will ask which agents. This policy gives the directive as to what neighbours should be asked a given query, thus uses the similarity metric when comparing a query with the modeled expertise of its neighbours.

2.1.2 Answering Policy

A policy for making decisions regarding the answering of queries is called the Answering Policy. An agent uses this policy when generating answers for a given query, thus utilizes the capability metric. If the capability metric, when comparing the generated answer to the given query, is above a certain threshold (explained later) then the answer is sent to the querying agent.

2.1.1 Referring Policy

The Referring Policy defines how and to which of its neighbours an agent will refer another agent, after being posed a query it cannot answer. This policy is similar to the asking policy and uses the similarity metric. There are three main referral policies used in referral networks.

2.1.1.1 Refer All Matching Neighbours

An agent uses the similarity metric to calculate how similar a neighbour's modeled expertise is to the query. Only neighbours scoring above a certain threshold will be referred [Singh and Yolum, 2003a].

2.1.1.2 Refer All Neighbours

When an agent cannot answer a query they refer all their neighbours. This resembles the Gnutella search process, which refers all it's known neighbours if it does not have the requested file [Singh and Yolum, 2003a].

2.1.1.3 Refer the Best Neighbour

The neighbour with the highest score of similarity with the current query is referred. This is similar to how the Freenet protocol refers clients to the neighbour that most likely has the requested information [Singh and Yolum, 2003a].

2.1.2 Learning Policy

The Learning Policy defines how an agent learns things about the other agents that it interacts with. Two main functions of this policy is to determine how to learn from questions, which is used to update the modeled interest of the querying agent, and learn from responses, which is used to update the modeled expertise and sociability of an agent who answered or gave a referral to a query.

2.1.3 Changing Neighbours Policy

Based upon the quality of answers and referrals that an agent receives from its neighbours it may decide to drop existing neighbours in favour of some other agent. The policy that achieves this is called the Changing Neighbours Policy. There are three implementations of this policy. The Providers policy sorts acquaintances by how their expertise matches the agent's interests, thus an agent using this policy would tend to have providers (or experts) as neighbours. The Sociables policy sorts acquaintances in terms of their sociability. An agent using this policy would have agents who give referrals as neighbours. And a weighted average policy sorts acquaintances in terms of a weighted average of sociability and how a neighbour's expertise matches the agent's interests.

2.1.4 Querying Policy

The Querying Policy defines how the agent will generate queries. Usually queries are a slightly perturbed version of the agent's interest vector.

2.2 Referral Network Variables

There are certain variables that are an integral part of the policies used during the simulation of a referral network. These variables are initialized during the setup phase of the simulation.

2.2.1 Similarity Threshold For Asking Neighbor

This is the threshold that determines whether a neighbor will be contacted for a query. After the similarity metric is applied to the query and the modeled expertise of the neighbor, it is compared to this threshold. If the value exceeds the threshold, then the neighbor is contacted. The default value is 0.01. Having a small value ensures that at least most of the neighbors will be contacted.

2.2.2 Similarity Threshold For Answering Questioner

This is the threshold that determines whether an agent should answer a query it receives. The agent applies the capability metric to the incoming query and its own expertise vector. If the resulting value is above the threshold, then the agent answers the query. The default value for this threshold is 0.3. This value ensures that only agents with high expertise (and thus good answers) attempt to answer the query.

2.2.3 Similarity Threshold For Referring

Again the similarity metric is used to compare a query with the modeled expertise of an agent's neighbours. If the similarity is above this threshold then the agent's neighbour is referred to the querying agent. This is the threshold used by the Refer all and Refer all matching policies. When the threshold is taken to be 0.1, the Refer all policy is in effect. Other thresholds apply to the Refer all matching policy. The default referral policy is Refer all matching with the Similarity Threshold For Referring set to 0.2.

2.2.4 Weighted Sociability

This denotes the weight of sociability in choosing neighbors. The neighbor selection policies use this weight. When W is set to 0, only the expertise of the neighbors is considered. When W is set to 1.0 only sociability of the neighbors is considered. The default neighbor selection policy is weighted average, where W is set to 0.5.

2.2.5 Similarity Threshold For Evaluating Answer

This is a threshold used for evaluating an answer. When an agent receives an answer to its query, it applies the capability metric to the query and the answer. If this value is above this threshold, then the answer is considered useful and is used to by the agent to update its expertise and the model of the agent from which it got the answer. The default value for this variable is 0.2.

2.2.6 Number Of Neighbor Set Changes

This denotes how many times each agent can change its neighbors. By default experiments are run until the agents stop changing their neighbors [Yolum, 2003], i.e., until each agent is satisfied with its neighbors.

2.2.7 Number Of Questions Per Neighbor Set

This denotes how many queries are allowed before a neighbor change takes place. The default value is 3.

2.2.8 Maximum Perturbation Percentage For Generating Query

This defines the maximum amount the interest vector will be perturbed to generate a query and is used by the Querying policy. The default value is 0.1.

2.2.9 Default Interest and Expertise Model

This defines the default number of dimensions and values for expertise and interest vectors in each agent.

2.3 Initialize a Population of Agents

During this stage of execution agents are generated. This involves setting up a specific number of agents with the policies and network referral variables described above. Certain agents (experts) are given a domain of expertise, and thus given a high value in the dimension corresponding to that domain of their expertise vector. The rest of the agents (consumers) are given a range of interests, which is randomly chosen thus distributing their interests across all possible domains (although it can be tuned to place interests in a certain domain). Once all the agents are initialized each is setup with a predefined initial number of randomly chosen neighbours. The agents are now ready to run the simulation.

2.4 Referral Network Simulation

A Referral Network Simulation runs until the number of neighbour set changes variable is met. Each time, every agent will ask a certain number of questions (number of questions per set variable) of its neighbours. The neighbours are given a chance to answer or provide referrals. Once all answers have been received the agent then updates its model of all its neighbours. An agent generates their queries in accordance to the following algorithm [Singh and Yolum, 2003a].

Algorithm 1 Ask-Query()

```
Generate Query (using the Query Policy)
Send query to matching neighbours (using the Asking Policy)
while (!timeout) do
        Receive Message
        if(message.type == referral) then
                Send query to referred agent
                Add referral to referral graph
        else
                Add answer to answerset
        end if
end while
for i = 1 to |answerset| do
        Evaluate answer(i) (must be above Similarity Threshold For
```

```
            Evaluating Answer variable)
14:         Update agent models (using the Learning Policy)
15:   End for
```

An agent uses the following algorithm to answer a query [Singh and Yolum, 2003a].

Algorithm 2 Answer-Query()

```
if hasEnoughExpertise then (using the Answering Policy)
        Generate Answer
else
        Refer Neighbours (using the Referral Policy)
end if
```

Once an agent has completed asking and answering a set of questions, they are allowed to choose new neighbours based on their updated agent models (line 14, algorithm 1). The Changing Neighbours Policy is used to do this step.

Once a referral network simulation completes, performance metrics are run and results obtained as to the effectiveness, efficiency and authority of the referral network and its associated setup. The results of Yolum and Singh are presented and explained in the subsequent section.

3. Measuring Performance of Referral Networks

Once a simulation has ended several metrics can be used to evaluate the policies and setup of a referral network. There are three main evaluation points for referral networks. They are:

1. Effectiveness
2. Efficiency
3. Authoritativeness

3.1 Effectiveness

The effectiveness of a system measures how easily agents find useful providers [Singh and Yolum, 2003a]. There are two metrics used to measure effectiveness. The first is *direct quality*, which is measured as the usefulness of the direct neighbours of an agent, in terms of their expertise and the agent's interests. The second is *nth* best quality. This metric takes into consideration "how well the agent's interest matches the expertise of all other agents in the system, scaled down with the number of agents it has to pass to get to the agent" [Singh and Yolum, 2003a]. So further away agents will contribute less to the quality of a given agent. The contribution of agent *j* to agent *i's* quality is given by:

$$\frac{I_i \otimes E_j}{path(i,j)}$$

EQUATION 4: Quality Contribution

The nth best quality metric is computed by calculating all qualities (using Equation 4) from an agent to all other agents in the network, and then taking the *nth* best measure from a sorted list. The value of *n* is taken to be twice the number of neighbours for a given agent.

3.1.1 Effectiveness Results

It has been shown in [Singh and Yolum, 2003a] that when increasing the threshold to refer a neighbour, i.e. moving from the “refer all” policy to the “refer all matching” policy that the quality of network graph increases. Intuitively this is summarized that giving better referrals implies a referral network with a better overall quality. As agents are more selective in their referrals the likelihood of a better answer being found by a querying agent increases, thus they will find an expert and will be able to receive a high-quality answer to their query. Yolum and Singh make the following observation, “among these referral policies Refer All Matching results in graphs with higher quality, where the best threshold increases with the percentage of experts in the society.” Yolum describes this in [Yolum, 2003] as follows, “Exchanging more referrals does not guarantee that the quality of the network will be high. The structure of the network can prevent consumers from locating some of the service providers". For instance all of an agent's neighbours may not be able to give good referrals and thus a high number of referrals do nothing in the way of aiding the querying agent in finding a good answer.

The findings of Singh and Yolum in [Singh and Yolum, 2003a] are presented in the graphic below. Notice that the Best Matching Neighbour policy is independent of the referral threshold as the best neighbour for a query is always returned when this policy is in effect.

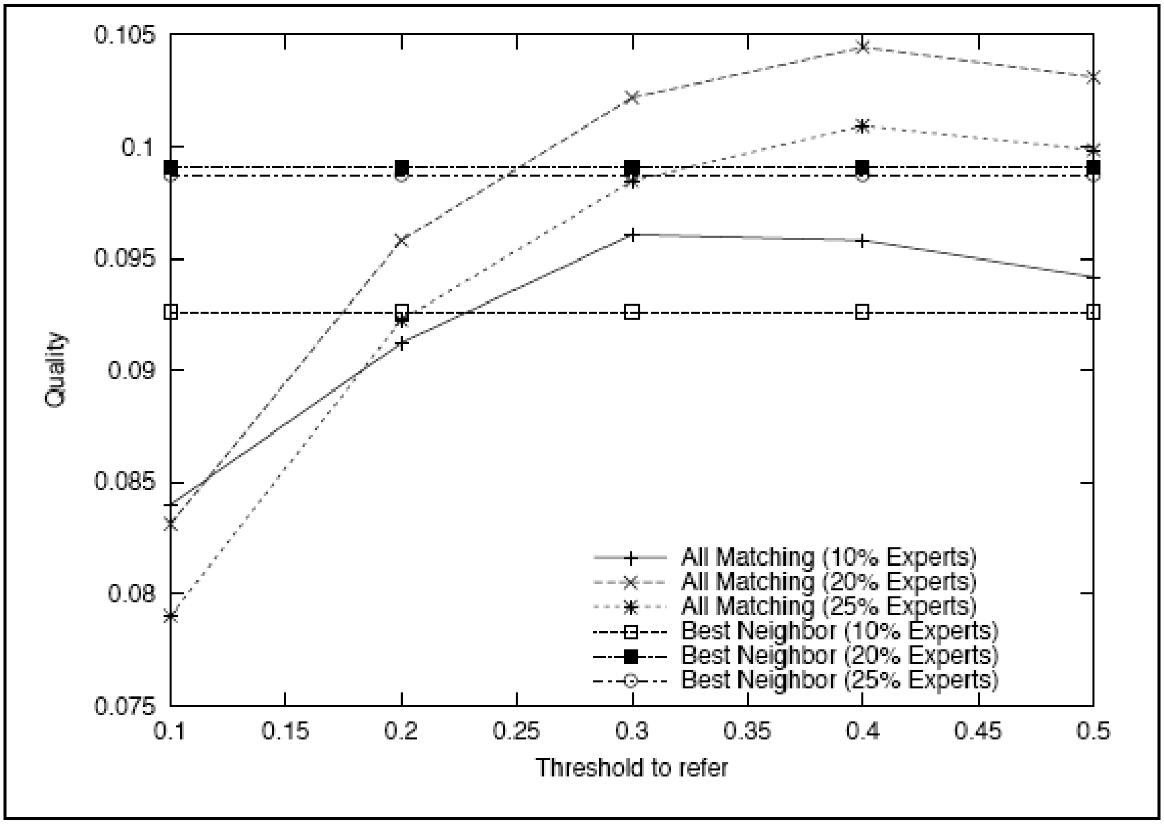

FIGURE 3.1 Performance of Referral Policies

3.2 Efficiency

The quality measure of a network graph is optimistic [Singh and Yolum, 2003a] since it is possible that a provider may not respond and other agents may not make useful referrals. Thus a high quality graph does not mean that an agent will reach the service it is trying to find. A new metric is introduced to measure the efficiency of finding answers. Efficiency is defined as the ratio of correct answers received to the number of agents contacted [Singh and Yolum, 2003a]. Efficiency is a measure of how well experts are responding to questions within the network.

3.2.1 Efficiency Results

Similar to the effectiveness metric the efficiency metric is dependent on the setting of threshold to refer. When this threshold is set low, the referrals become less selective and more agents are contact, thus lowering the efficiency. If it is set to high, then referrals are too restricted and not enough agents are contacted to find useful answers. The following graphic summarizes the findings from [Singh and Yolum, 2003a] in this area.

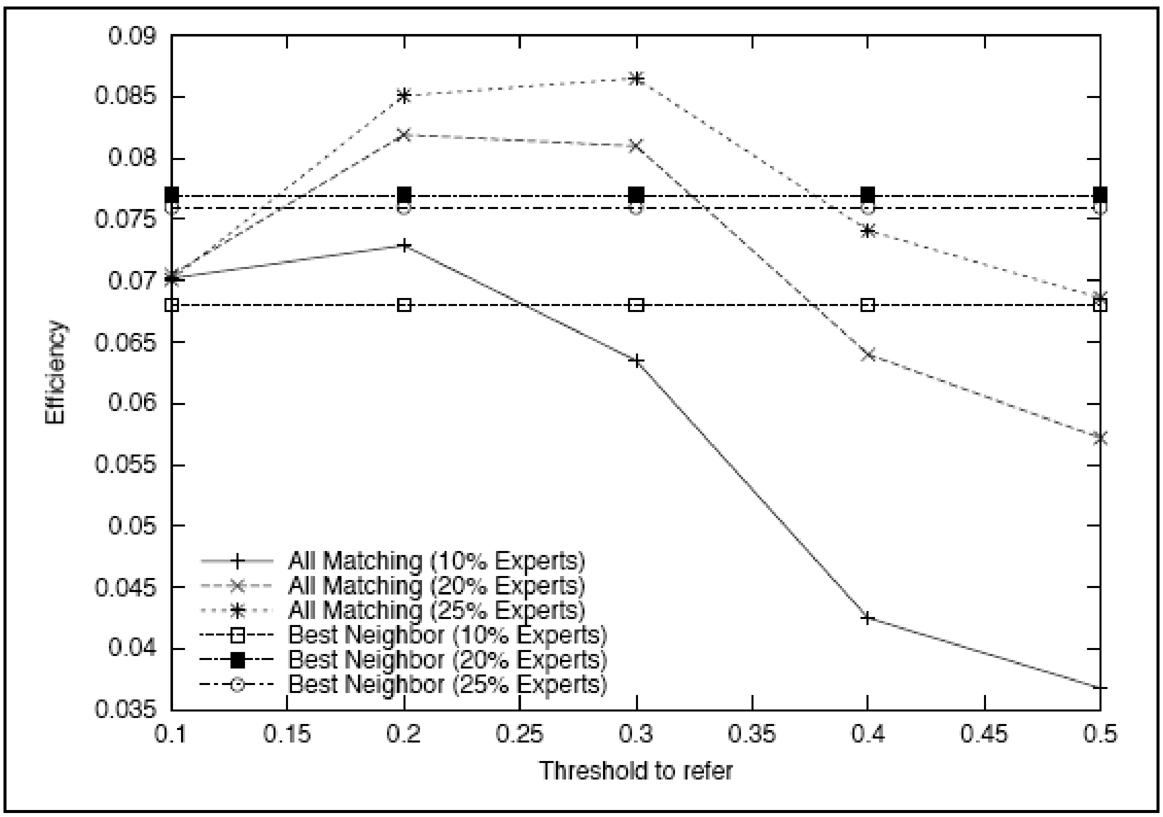

FIGURE 3.2 Efficiency of Referral Network

3.3 Authoritativeness

Some agents will be chosen as a neighbour by a greater number of other agents, and are thus identified as more authoritative. They may be more sociable or have a high expertise, and thus are chosen as a neighbour more often than other agents who are less sociable or have a smaller expertise. The authoritativeness is measured using the PageRank metric to study the emergence of authorities in referral networks. Yolum and Singh have found that more authorities emerge through the Refer All Matching Policy, but that the Refer All Policy causes the emergence of authorities whose level of authoritativeness is higher [Singh and Yolum, 2003a]. This is shown in the graphic below, from [Singh and Yolum, 2003a].

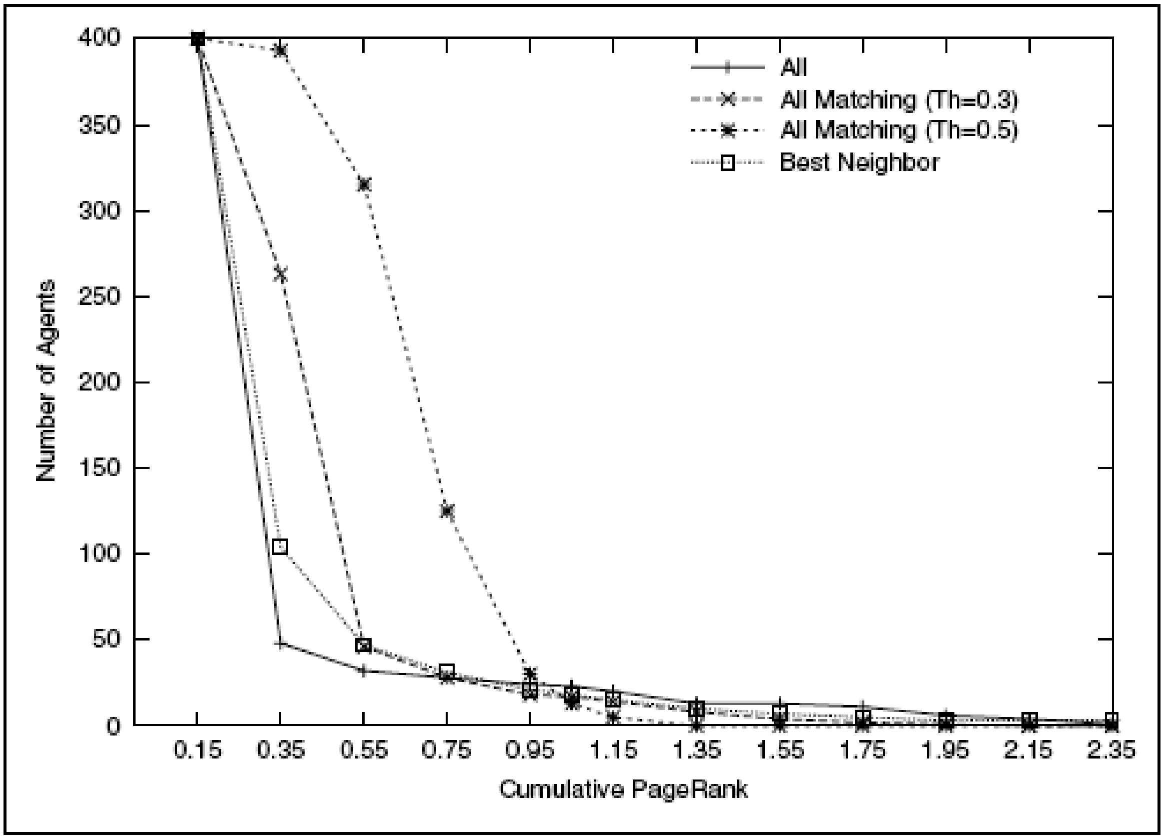

FIGURE 3.3: PageRank Distribution

With the basics of a referral network covered and how it is evaluated, the next chapter introduces the most important portion of this paper, the integration of relational concept knowledge into a referral network.

4. Concept Relations in Referral Networks

The notion of referral networks thus far and their application to knowledge management, lacks two fundamental aspects; firstly the relation of concepts within a domain, and secondly the ability of an agent to dynamically change their interests. In the real-world humans innately relate concepts, for example dog is to animal what apple is to fruit. It is our ability the relate concepts and form hierarchies of relations that allows us to effectively process information in our dynamic world.

Within a real-life social networks concept relation knowledge is of great importance, it allows us to change our interests to reflect newfound knowledge, rephrase questions in a manner that can return better results, and give suggestions to people asking us questions, in the form “did you mean this instead of that.” For instance within a real-life social network, you may ask a mathematician about Einstein’s Theory of Relativity, the mathematician may not be able to answer you with same amount of expertise as a physicist because he has little knowledge of the theory, but he may have enough relational concept knowledge to give you a suggestion on what you should ask for instead. He may tell you, "instead of asking me about Einstein’s Theory of Relativity, why don’t you ask me about Einstein’s field equations"; both are topics in the same domain and tightly related to each other. The mathematician is most likely very knowledgeable about the intricacies of the math involved in Einstein’s Field Equations and would give good answers to questions in that domain. Provided you could understand the math involved in Einstein's Field Equations learning about them would no doubt help augment your knowledge of the Theory of Relativity. Thus learning that a topic is related to another and then learning about this new topic, in the long run educates you on the original topic.

Within the Vector Space Model (which is used to model expertise and interest in a referral network) it is quite possible that the expertise vector of an expert and the query vector of a consumer are very different upon analysis using the similarity metric described in Equation 1. However it is very possible that the interests of the consumer lie in the same domain as the expertise vector of the expert and may even be tightly related. Thus a small amount of concept relational knowledge would be of great benefit, if the consumer were aware of it.

As can be seen visually below being aware of relational concept knowledge allows an agent to dynamically change its interests to better learn about concepts in the domain of its interests. Figure 4.1 depicts the expertise vector of an expert in Einstein's Field Equations, while Figure 4.2 depicts a consumer interested in the Theory of Relativity. If a consumer were to ask the expert in Figure 4.1 about Einstein's Theory of Relativity it would not get an answer (or at least not a very good one). However, after learning that Einstein’s Field equations and the Theory of Relativity are related, if the consumer were to modify its interests placing more of an onus on Einstein's Field Equations and ask a question based on these newfound interests, the expert would then be able to answer with a much better degree of accuracy. This would lead to a satisfactory response, and thus the querying agent would learn about Einstein's Field Equations and in the long run the Theory of Relativity.

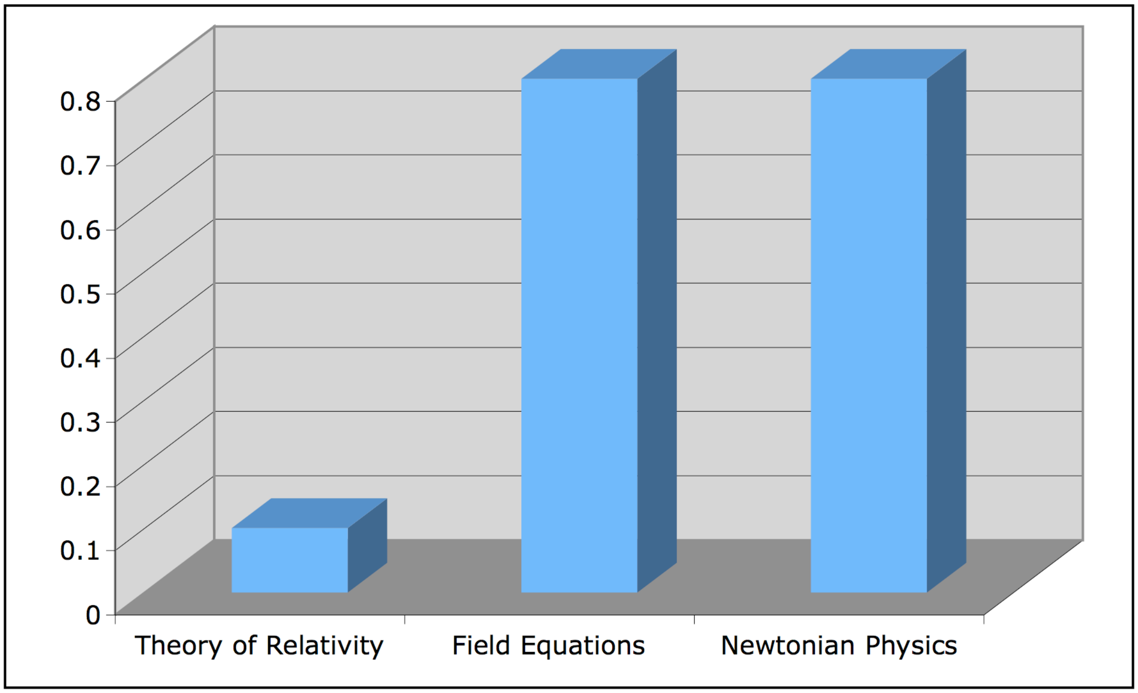

FIGURE 4.1: Example Expertise Vector

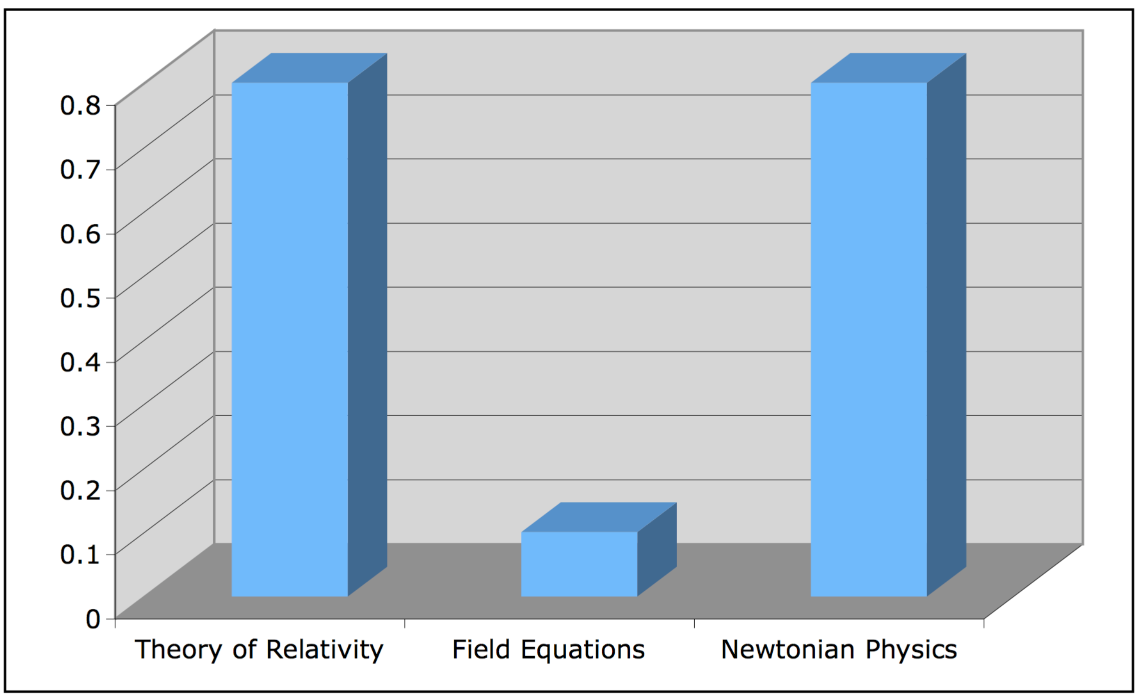

FIGURE 4.2: Example Consumer Interest Vector Before

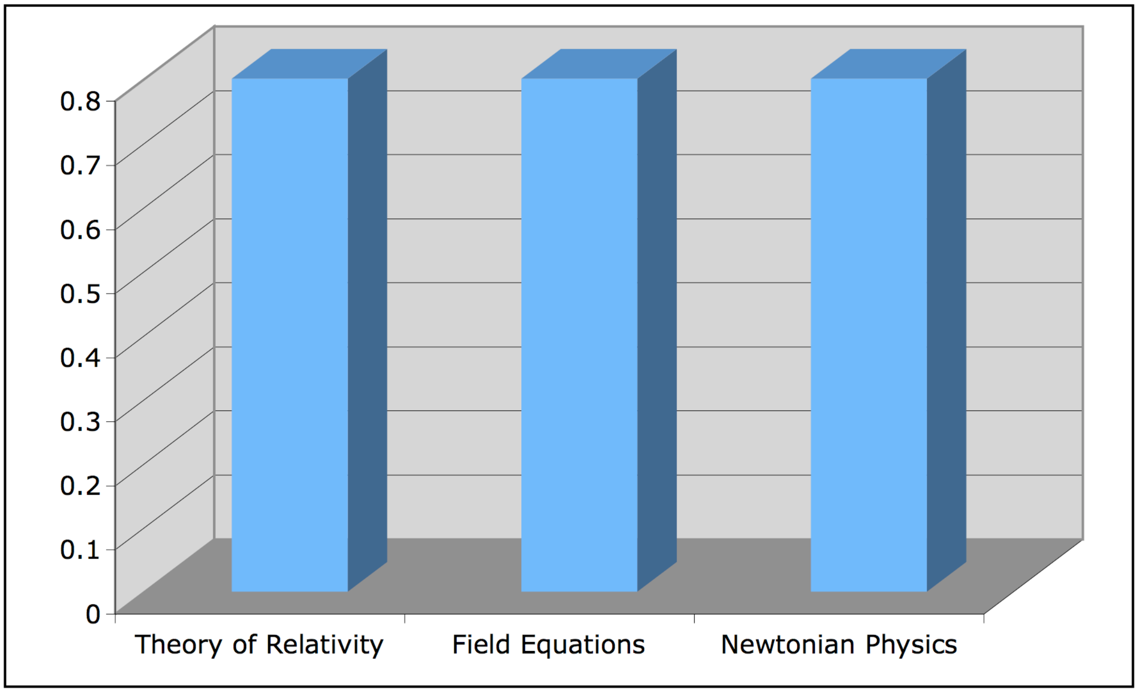

FIGURE 4.3 Example Consumer Interest Vector After

Notice that interest vector in Figure 4.3 now matches the expertise vector in Figure 4.1 in 2 of the 3 areas instead of 1 of the 3 areas as in Figure 4.2. This will result in the query (based on the interests of the querying agent) being more similar to the expertise vector of the expert and thus is more likely to overcome the capability threshold and be answered by the expert.

4.1 Suggestion and Interest Adaptation Policies

For a referral network to function as described above, two new policies need to be introduced. One is a suggestion policy and deals with what concept knowledge an agent will suggest to another agent upon receiving a query. The second is an interest adaptation policy and functions in the capacity of updating the interests of an agent who has concept knowledge about related topics.

4.1.1 Suggestion Policy

Within the referral network simulator the suggestion policy is used when an agent is asked a question. Algorithm 2 introduced in section 2.4 is modified in the following manner.

Algorithm 2 [Revised] Answer-Query()

```
1:  if hasEnoughExpertise then (using the Answering Policy)
2:          Generate Answer
3:  else if hasAGoodSuggestion then (using the Suggestion Policy)
4:          Give a suggestion in form of concept relation
5:  else
6:          Refer Neighbours (using the Referral Policy)
7:  end if
```

The standard policy for suggestions as implemented in this project follows the following directive. For all the concepts that the queried agent knows, find the pair of concepts that have the largest differential and return that information. Thus this policy is a “the best bang for the buck” implementation, and works on the assumption that where concepts are most different, learning a concept relation between them will make the most difference in the quality and effectiveness of subsequent queries.

4.1.2 Interest Adaptation Policy

The interest adaptation policy is used when an agent has finished asking a set of questions and is learning from the responses it has received. Algorithm 1 from section 2.4 is modified in the following manner to incorporate the functionality of the Interest Adaptation Policy.

Algorithm 1 [Revised] Ask-Query()

```
Generate Query (using the Query Policy)
Send query to matching neighbours (using the Asking Policy)
while (!timeout) do
        Receive Message
        if(message.type == referral) then
                Send query to referred agent
                Add referral to referral graph
        else
                Add answer to answerset
        end if
end while
for i = 1 to |answerset| do
        if answer(i) is a suggestion then
                Update concept knowledge store
        end if
        Evaluate answer(i) (must be above Similarity Threshold For
        Evaluating Answer variable)
        Update agent models (using the Learning Policy)
End for
Update interests based on knowledge (using Interest Adaptation Policy)
```

The standard method for updating interests is to apply concept knowledge to a pair of related concepts that have the largest difference in the agent's current interest vector. This is applied until the two related concepts have the same strength, and then the policy desists from modifying these two concepts. This allows for good integration with the standard suggestion policy, as it will not suggest a concept relation for two concepts that are very similar or exactly the same.

4.2 Propagation of Knowledge in Referral Network

The propagation of knowledge within a referral network depends on the structure of the network. If a graph is bi-partite then concept knowledge will be confined to the one section of the network in which one (or many) agent(s) contains concept knowledge. As shown below the set of agents on the left part of the graph will never become aware of concept knowledge that agents on the right may know about, as they are not connected.

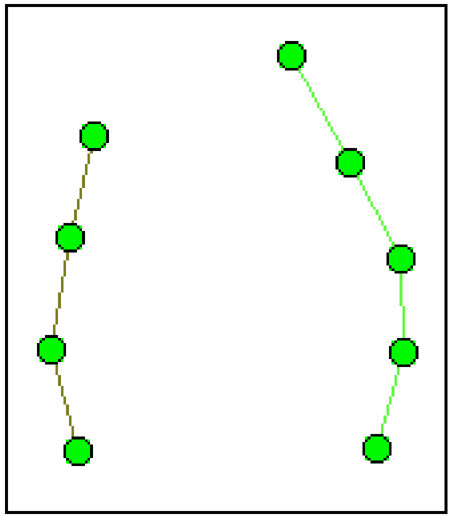

FIGURE 4.4 – Bi-partite Graph

If however the network graph is complete (fully connected) then concept knowledge will be quickly proliferated throughout the network. Through several simulation runs it was shown that all agents in the network are made aware of a concept relation within a relatively small number of queries if the network graph is complete, concept knowledge is always shared (regardless of whether or not an answer can be given), and messaging between agents is done in an asynchronous manner.

4.3 Strength of Concept Relations

A concept relation was modeled quite simply in the simulation by linking one dimension in a simulation vector (either interest or expertise) to another and attaching a strength to this relation. The strength of a relation is normalized between 0.0 and 1.0, where 0.0 denotes a small degree of relation and 1.0 indicates a large degree of relation. The bracketed numbers in the illustration below denote the strength of the concept relation.

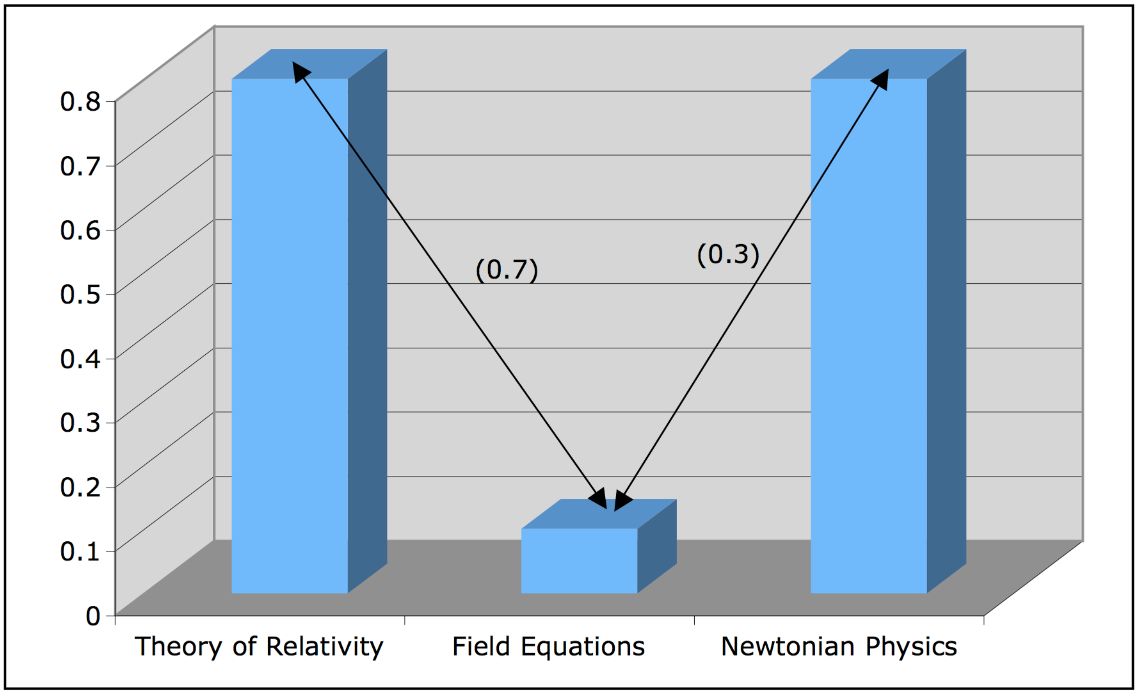

FIGURE 4.5 - Strength of Concept Relations

With the ability for concept knowledge to be integrated into a referral network and the suggestion and interest adaptation policies in place, an extensive set of simulations were run to test the effect on agents within the network. The results from these simulations are presented and explained in the following chapter.

5. Simulation Results

The main goal of this paper is to show that knowledge of concept relations within a referral network will outperform a traditional referral network in the three main areas of effectiveness, efficiency and authoritativeness.

To prove the true power of concept knowledge in a referral network, all simulations were tuned for a worse case scenario; agents would ask questions about a concept in a domain for which there were no experts in the system. However there would be experts for a related concept in the system. This setup would allow agents with concept knowledge to adapt their interests and begin learning a related topic. Concordantly expected properties of a referral network should emerge.

5.1 Effectiveness

To demonstrate the effect of concept knowledge on the effectiveness of a referral network, a referral network was tuned so that only experts of one kind existed in the network. All agents were generated with a high interest in the same domain of the experts but in differing concepts. For example in simulations below, the expertise vectors (of the experts in the system) were as follows [0.95, 0.05, 0.05, 0.05], while the interest vectors of the consumers were as follows [0.05, 0.95, 0.05, 0.05]. In Simulation 2 one agent had relational knowledge that the concept in the first index was related to the second with a strength of 0.7 and was allowed to share this knowledge with questioning agents.

Simulation 1 - with no concept knowledge	
Number of Agents	100
Percentage of Experts	10%, 20%, 25%
Number of Agents with Concept Knowledge	0
Threshold to Refer	0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5
Rate of Learning	0.0, 0.1
Cycles	10
Questions per cycle	3

Simulation 2 - with concept knowledge	
Number of Agents	100
Percentage of Experts	10%, 20%, 25%
Number of Agents with Concept Knowledge	1
Threshold to Refer	0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5
Rate of Learning	0.0, 0.1
Cycles	10
Questions per cycle	3

Observation 1

It is shown through these two simulations that even though consumers may initially be querying topics for which there are no experts, once they have learned that their interests are related to another concept (which experts in the network can answer), performance is similar to a referral network where consumers have interests in the domains and concepts that are known by experts in the network.

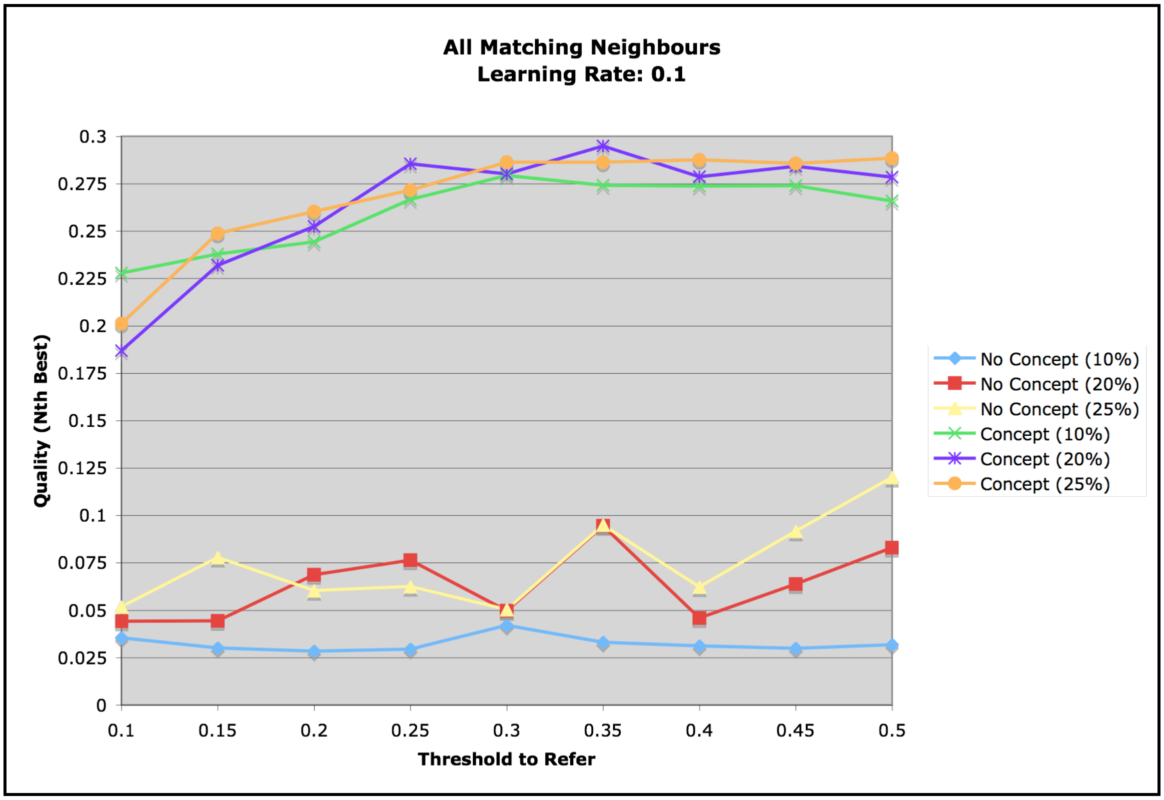

FIGURE 5.1 - Concept Knowledge and Effectiveness

As can be seen by the above graph a referral network without concept relation knowledge performs randomly and much worse than one with concept relation knowledge. The quality either fluctuates based on the threshold to refer (as in populations with 20% and 25% experts) or remains fairly constant (as in the population with 10% experts).

Note that the learning rate is how agents learn from responses and is not related to how agents learn about related concepts. Although learning a concept combined with learning from answers is a powerful combination. Not only does an agent learn about a subject, but also it learns how to question on a subject in a different manner, thus enables a greater degree of learning.

The next simulation test shows the effects of the Best Neighbour Referral Policy, with the learning rate set to 0.1, thus agents learn from the answers they receive.

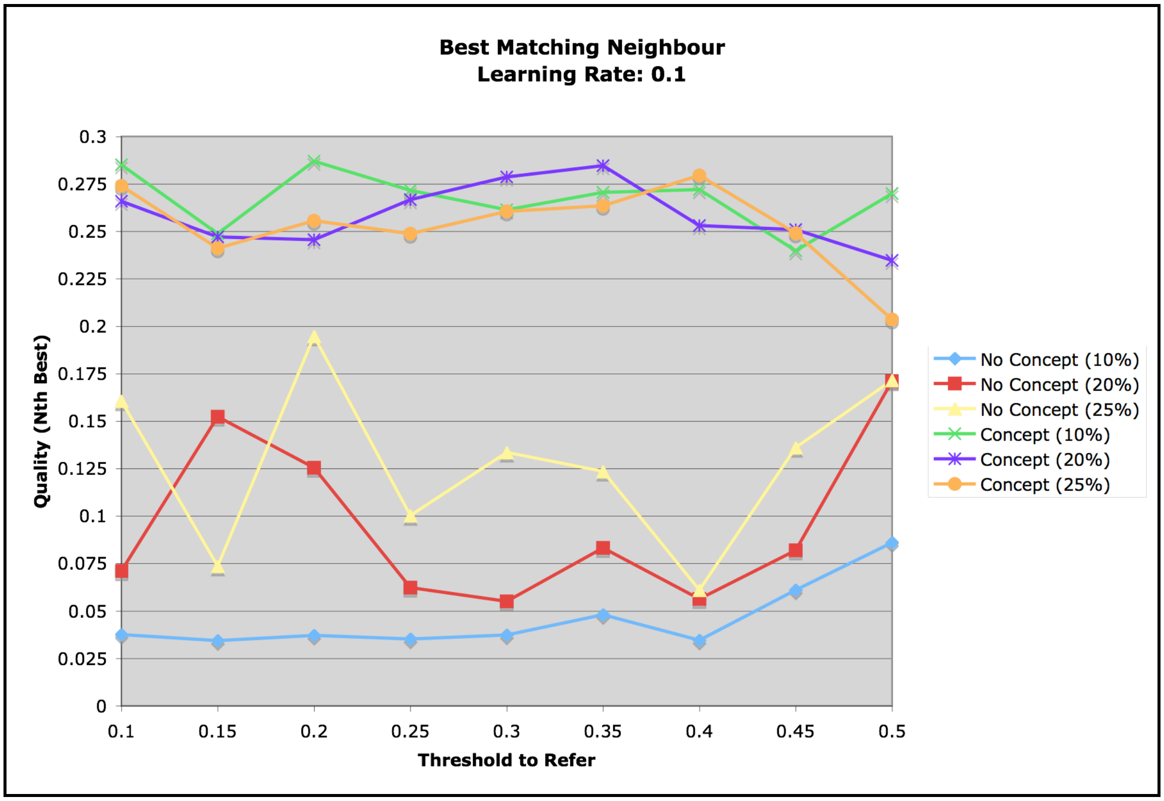

FIGURE 5.2 - Concept Knowledge and Effectiveness

Once again with the "refer best matching neighbour policy", the quality of the referral networks with concept knowledge is much better, while a traditional referral network performs randomly. Notice the quality of the graph for the referral network with concept knowledge slightly decreased with the learning rate set to 0.1. This is due to the fact that more experts are created at lower thresholds, as any agent is more likely to refer its best neighbour, thus creating more experts and increasing the overall quality of the graph. A referral network with a learning rate of 0.0 is much more stable as illustrated below.

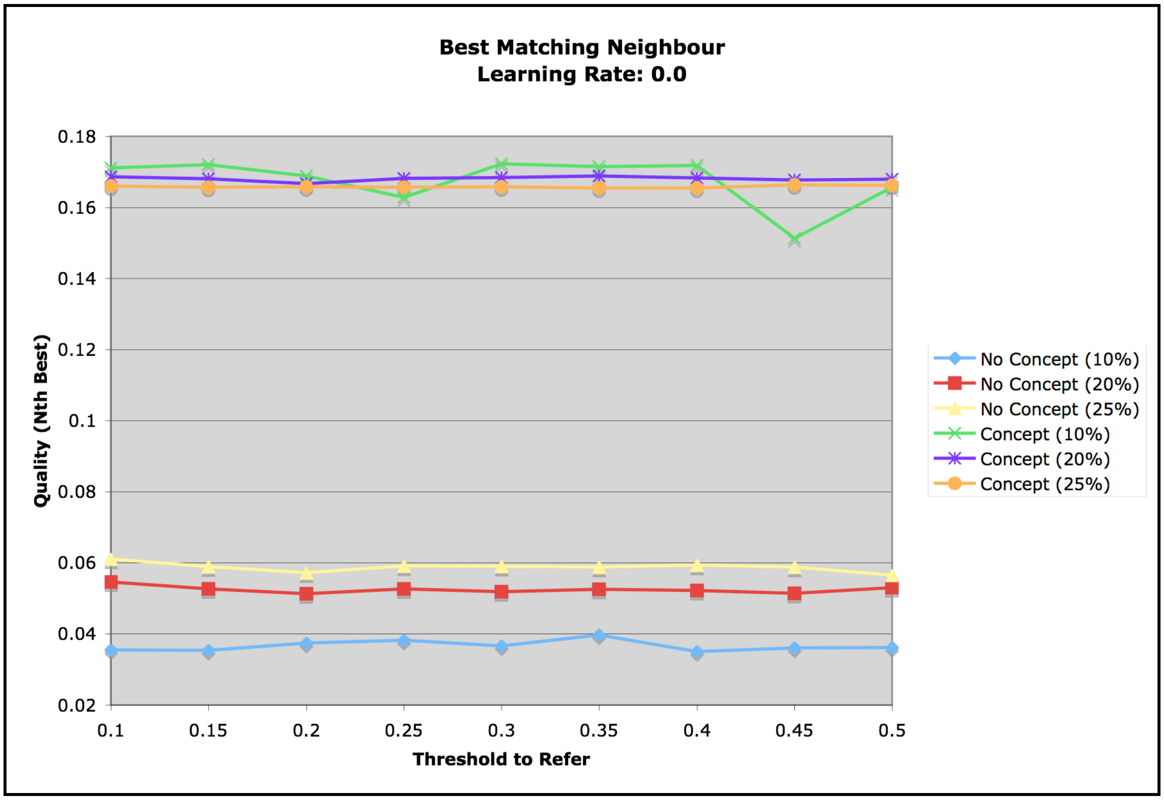

FIGURE 5.3 - Concept Knowledge and Effectiveness

Observation 2

The quality of a referral network with concept knowledge will never be worse than one without, and in the case where consumers are querying in topics related to the expertise (of specific experts) within the system, it will outperform a traditional referral network.

5.2 Efficiency

Simulation runs for obtaining efficiency results were setup exactly the same as for effectiveness except the learning rate was set to 0.0, as an increase in learning propagates experts and thus stabilizes the accuracy measure. The following results were obtained for efficiency in referral networks with and without concept knowledge.

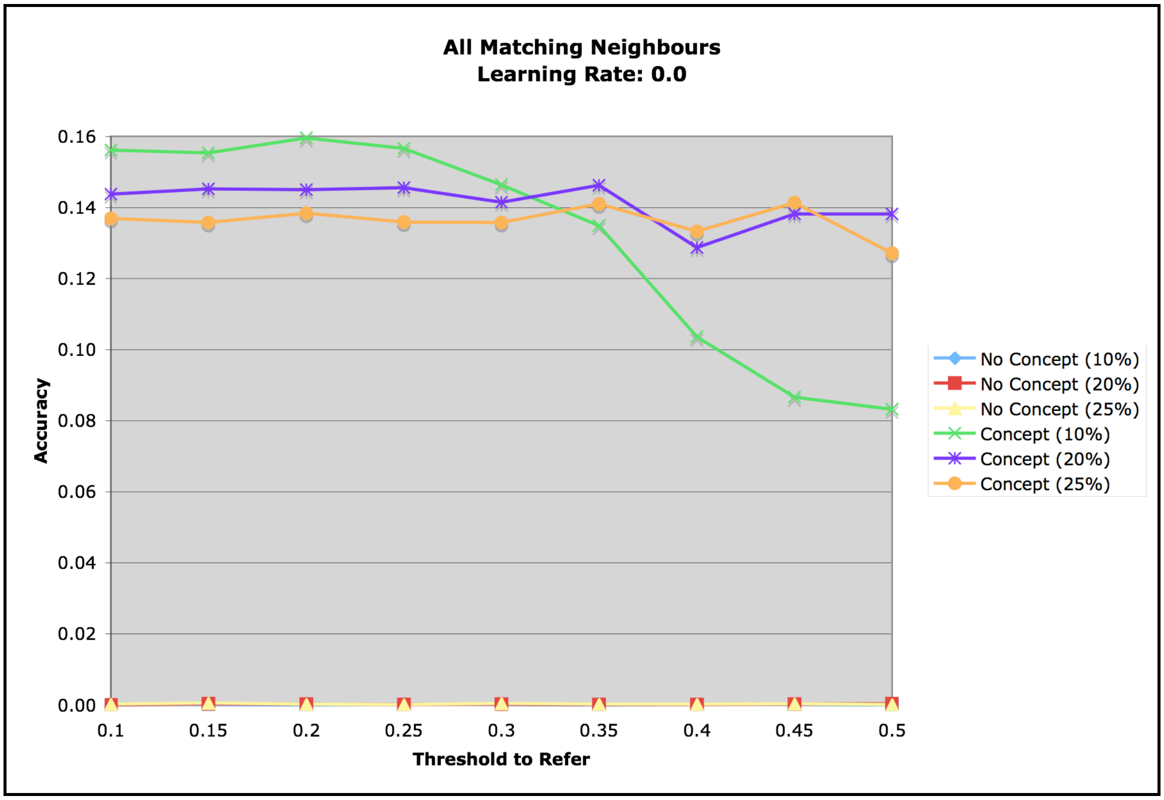

FIGURE 5.4 - Concept Knowledge and Efficiency

Observation 3

Accuracy in a referral network with concept knowledge is drastically improved due to the fact that agents eventually ask questions of the experts in their "domain of expertise", and thus receive good answers. In normal referral networks agents continue to ask questions of an expert even if it is in the wrong domain (as there is no chance for concept learning to take place), and thus efficiency suffers. The referral networks with concept knowledge experiences a decrease in accuracy as the threshold to refer increases due to the fact that agents give fewer answers at higher thresholds. This is an expected result and is similar to that found by Singh and Yolum [Singh and Yolum, 2003a].

Similarly with the Best Matching Neighbour policy the accuracy or efficiency of answering queries is much better with concept knowledge. Expert agents are found within the network and queried based on their specific expertise.

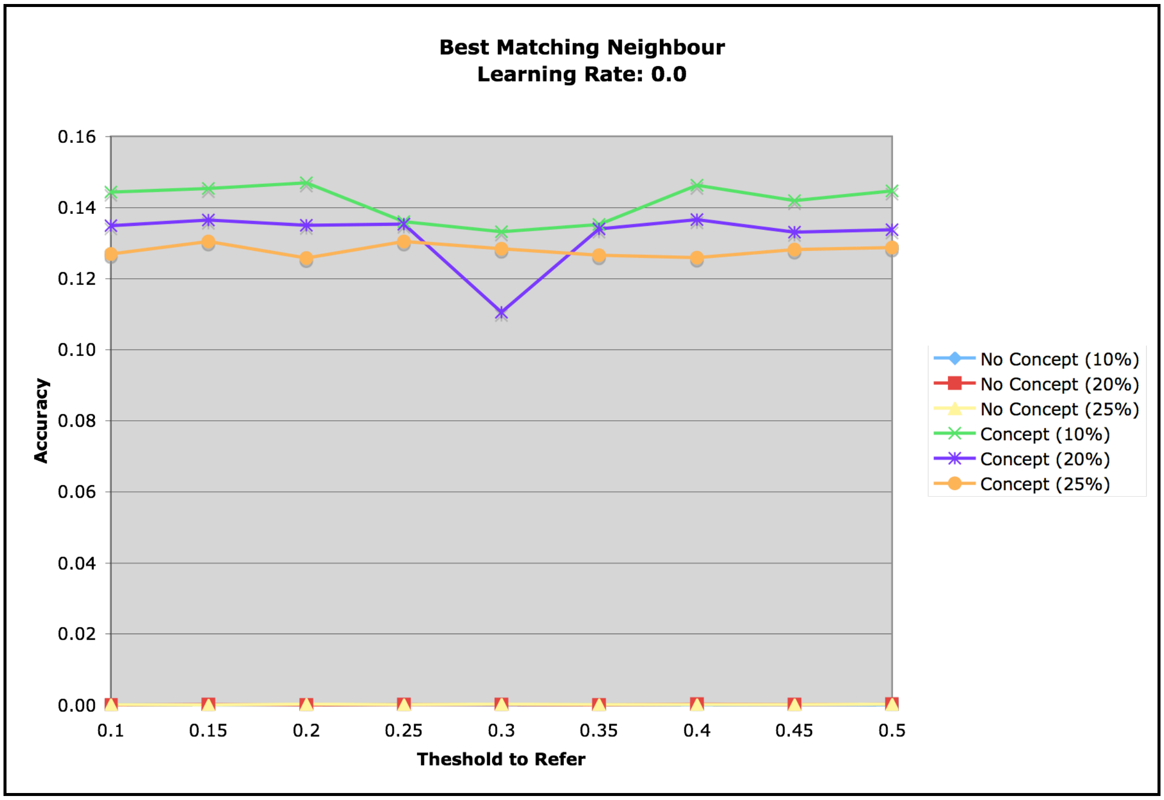

FIGURE 5.5 - Concept Knowledge and Efficiency

5.3 Authoritativeness

The emergence of authorities was measured in two distinct manners, one the emergence of the most powerful authority and two the emergence of the most number of authorities. The following results were obtained from referral network with the learning rate set to 0.0, and the same conditions as stated in section 5.1. Presented below are the results obtained for the highest-ranking authority in the referral networks.

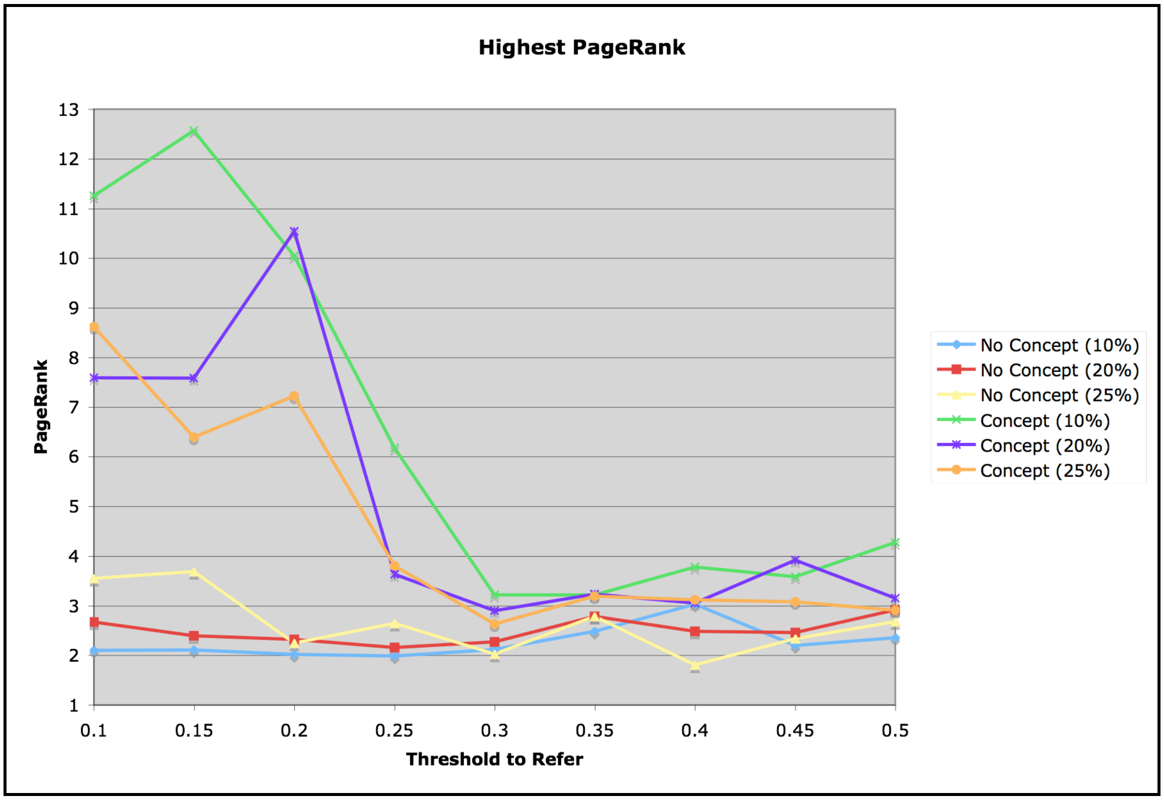

FIGURE 5.6 - Concept Knowledge and Highest PageRank

Observation 4

A referral network with concept knowledge promotes the emergence of stronger authorities as more consumer agents choose experts as neighbours. This is a direct result of the agents asking experts the correct questions (ones they can respond to), getting good answers and thus keeping them as neighbours. This causes the emergence of agents with a much higher authority.

Notice that in populations with a fewer number of experts that a higher PageRank is observed as there are less experts in the system and thus more agents choose fewer experts as neighbours; thus leading to some experts with a much larger set of neighbouring agents and a higher PageRank. This shows the intuitive fact that expertise is more concentrated in networks with a fewer number of experts.

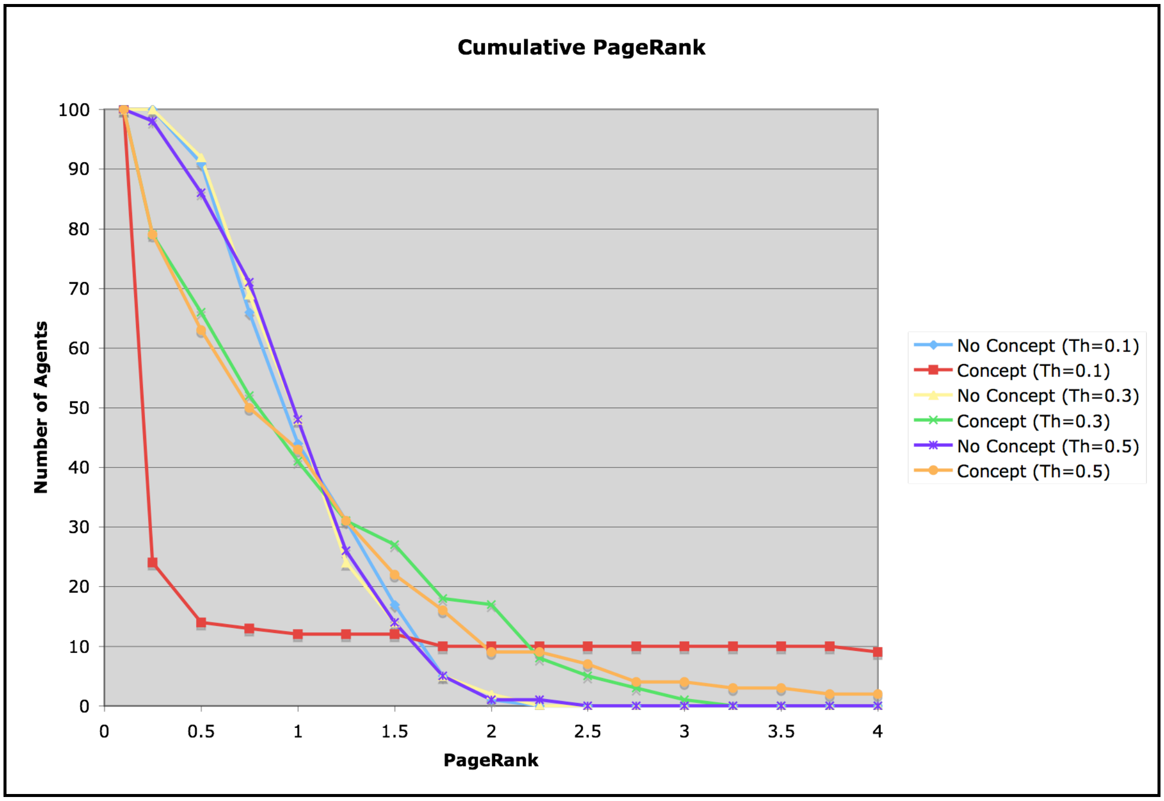

FIGURE 5.7 - Cumulative PageRank

Observation 5

Referral networks with no concept knowledge cause the emergence of more agents with less authority, while referral networks with concept knowledge cause the emergence of a greater number of more authoritative agents. Again this is due to the fact that agents are neighbouring with experts in the system thus raising their authoritative rankings. These results comply with the findings of Singh and Yolum [Singh and Yolum, 2003a].

It has been shown through these simulation runs that a referral network with concept knowledge will outperform one without, in the areas of effectiveness, efficiency and authoritativeness, as it allows agents to better find and ask answers of their neighbours.

6. Conclusions and Future Work

Social networks are a natural way for people to interact within an environment and obtain knowledge. Referral systems are an effective model of social networks because they capture two essential aspects of social networks, how they are used and how they change. The integration of relational concept knowledge allows for referral networks to even better model the dynamic nature of how humans interact and represent the world in which they live. It not only provides a mechanism for discovering answers to questions more effectively and efficiently and allows the expertise of a network to be better represented, but allows agents to discover new questions to ask in order to learn more effectively.

The referral networks described in this paper are much less complex than real-life social networks, but already could spawn some interesting new developments. Firstly, the effect of relational concept knowledge on the sociability of agents who provide suggestions could be tested and measured to ascertain the effect of keeping neighbours who have given suggestions in the past. Perhaps more importantly a more complex model of an agent's interests, expertise, and relational knowledge could be implemented. Currently these are modeled at a fixed size and cannot grow, an agent in a future referral system should be able to learn about new domains of knowledge and expand their interests, knowledge and expertise accordingly, in much the same way that humans learn not only about the things they already know, but also things that are new to them.

Appendix A - References

1 IPR Computing Ltd. (2002), The Google PageRank Algorithm and How It Works by Ian Rogers, http://www.iprcom.com/papers/pagerank/

2 Singh, Munindar P, Yolum, Pinar (2003) Emergent Properties of Referral Systems, Department of Computer Science, North Caroline State University.

3 Singh, Munindar P, Yolum, Pinar (2003) Dynamic Communities in Referral Networks, Department of Computer Science, North Caroline State University.

4 Yolum, Pinar (2003) Properties of Referral Networks: Emergence of Authority and Trust, Department of Computer Science, North Caroline State University.

Appendix B - Test Cases/Suites

The following are the test suites/cases used to obtain the results demonstrated in the preceding sections of the paper.

Identifier	**TC-01**
Title	Basic concept knowledge versus no concept knowledge.
Description	Test referral network with agents that have an interest in a domain which is related to the domain that a expert agent has expertise in.
Steps	1. Setup a population such that an expert agent has expertise in one domain. 2. Make all other consumer agents interested in a related domain, but not the domain the expert agent has expertise in. 3. Create one agent that has concept knowledge of the relation of two domains (the one the expert agent has expertise in and the one the consumer agents are interested in). 4. Run two simulations. 5. Compare effectiveness (quality), efficiency, and authoritativeness.
Expected Result	The population of agents with concept knowledge should have a higher quality graph, a better efficiency ratio (correct answers to total answers) and the expert should have a higher PageRank.
Actual Result	Network with Concept Knowledge ➢ Quality: 0.23671812785724256 ➢ Effectiveness: 35:155 = 0.2258 ➢ PageRank: 3.43244511 Network without Concept Knowledge ➢ Quality: 0.09157160853273438 ➢ Effectiveness Ratio: 0:150 = 0.0 ➢ PageRank: 1.919361473
Pass/Fail	Pass
Discussion	The quality of the network with concept knowledge shows not only much better quality but a much better change in quality. The initial quality was 0.08807202684830155 showing an increase of 0.14864 while the network with no concept knowledge had an initial quality of 0.08113833401520265 showing an increase of 0.0104333. Notice that the number of total answers given in the network with concept knowledge is 155, which is greater than the total given in the network with no concept knowledge. This is due to the fact that referrals were given

	to the expert node once the concept relation had been learned and thus the expert had to answer more questions. The PageRank results are straightforward; the expert has more neighbours and thus more authority. This would lead to more clustering around the expert and a greater direct quality for the network.

Identifier	**TS-01**
Title	Effect of Learning Rate.
Description	Demonstrate the effect of learning rate on quality and accuracy of referral network.
Steps	1. Setup a population such that an expert agent has expertise in one domain. 2. Make all other consumer agents interested in a related domain, but not the domain the expert agent has expertise in. 3. In one population, create one agent that has concept knowledge of the relation of two domains (the one the expert agent has expertise in and the one the consumer agents are interested in) and in another exclude this agent. 4. Run one simulation with a learning rate of 0.1. 5. Run one simulation with a learning rate of 0.2. 6. Compare quality and accuracy.
Expected Result	Quality and accuracy in the referral network with higher learning rate will be greater than the one with a smaller learning rate.
Actual Result	As Expected.
Pass/Fail	Pass
Discussion	N/A

Identifier	**TS-02**
Title	Effectiveness in concept referral network.
Description	Test the effect on effectiveness of a referral network with concept knowledge versus one that does not.
Steps	1. Setup a population such that expert agents have expertise in one domain. 2. Make all other consumer agents interested in a related domain, but not the domain the expert agent has expertise in. 3. In one population, create one agent that has concept knowledge of the relation of two domains (the one the expert agent has expertise in and the one the consumer agents are interested in) and in another exclude such an agent.

	4. Run two simulations across the following thresholds to refer {0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5}, for two learning rates {0.0, 0.1} and with 10%, 20% and 25% experts. 5. Compare effectiveness.
Expected Result	The population of agents with concept knowledge should have a higher quality graph in all simulations. For the referral network with concept knowledge and the learning rate is set to 0.0 quality should drop off around the 0.3 threshold to refer mark. Also for a learning rate of 0.1, they should logarithmically increase. For a referral network with no concept knowledge, the quality should be linearly stable and low for a learning in rate of 0.0, and random for a learning rate of 0.1 as some agents will gain random expertise from some answers, thus increasing the quality of the graph.
Actual Result	As expected.
Pass/Fail	Pass
Discussion	See report section 5.1 for further details.

Identifier	**TS-03**
Title	Efficiency in concept referral network.
Description	Test the effect on efficiency of a referral network with concept knowledge versus one that does not.
Steps	1. Setup a population such that expert agents have expertise in one domain. 2. Make all other consumer agents interested in a related domain, but not the domain the expert agent has expertise in. 3. In one population, create one agent that has concept knowledge of the relation of two domains (the one the expert agent has expertise in and the one the consumer agents are interested in) and in another exclude such an agent. 4. Run two simulations across the following thresholds to refer {0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5}, for two learning rates {0.0, 0.1} and with 10%, 20% and 25% experts. 5. Compare efficiency.
Expected Result	The population of agents with concept knowledge should have a higher efficiency ratio in all simulations. For the referral network with concept knowledge and the learning rate is set to 0.0 efficiency should decrease from around the 0.3 threshold to refer mark. Also for a learning

	rate of 0.1, they should remain linearly stable. For a referral network with no concept knowledge, the efficiency ratio should be linearly stable and low for a learning rate of 0.0, and random for a learning rate of 0.1 as some agents will gain random expertise from some answers, thus will be able to answer some queries. This will randomly increase the efficiency ratio of the simulation.
Actual Result	As expected.
Pass/Fail	Pass
Discussion	See report section 5.2 for further details.

Identifier	**TS-04**
Title	Effectiveness in concept referral network.
Description	Test the result on effectiveness of a referral network with concept knowledge versus one that does not using the Best Neighbour Referral Policy
Steps	1. Repeat Test Suite TS-02 with the referral policy set to "Best Neighbour"
Expected Result	The population of agents with concept knowledge should have a higher quality graph in all simulations. For the referral network with concept knowledge the quality of the graph should remain linearly stable. For a referral network with no concept knowledge, the quality of the graph should fluctuate randomly for a learning rate of 0.1 and should be linearly stable for a learning rate of 0.0.
Actual Result	The quality of the graph for the referral network with concept knowledge slightly decreased with the learning rate set to 0.1. This is due to the fact that more experts are created at lower thresholds, as any agent is more likely to refer its best neighbour. All other results are as expected.
Pass/Fail	Pass
Discussion	See report section 5.1 for further details.

Identifier	**TS-05**
Title	Efficiency in concept referral network.
Description	Test the effect on efficiency of a referral network with concept knowledge versus one that does not using the Best Neighbour Referral Policy
Steps	1. Repeat Test Suite TS-03 with referral policy set to "Best Neighbour".

Expected Result	The population of agents with concept knowledge should have a higher efficiency ratio in all simulations. The efficiency ratios of both referral network simulations should remain linearly stable.
Actual Result	As expected.
Pass/Fail	Pass
Discussion	See report section 5.2 for further details.

Identifier	**TS-06**
Title	Authoritativeness in concept referral network.
Description	Test the result of concept knowledge on authoritativeness in a referral network.
Steps	1. Repeat Test Suite TS-02 and Test Suite TS-04. 2. Compare highest authorities across all thresholds for both referral networks with learning rate set to 0.1. 3. Compare cumulative PageRank for Best Neighbour policy, and thresholds 0.1, 0.3, 0.5 for both referral networks with learning rate set to 0.1.
Expected Result	The population of agents with concept knowledge should have an authority with a higher PageRank than the best authority in the referral network with no concept knowledge. Referral networks with no concept knowledge should produce more agents with smaller authority values, while referral networks with concept knowledge should produce more agents with higher authority values.
Actual Result	As Expected.
Pass/Fail	Pass
Discussion	See report section 5.3 for further details.

Appendix C - Source Code Actions

Referral Network Simulator		
File Name	**Action**	**LOC**
Agent.java	Added new instance variables and get/set methods.	20
Agent.java	Added code to handleQuestionMessage method for sending suggestions.	18
Agent.java	Added method handleSuggestionMessage.	20
Agent.java	Added code to learnFromResponses method for adapting interests.	8
Concept.java	Created class for representing concept relations.	138
ConceptVector.java	Created class for handling concept vectors.	119
InterestAdaptationPolicy	Created interface for adapting agent's interests.	17
InterestAdaptationPolicyStandard.java	Created implementation of InterestAdaptationPolicy interface.	104
SuggestionPolicy.java	Created interface for making suggestions.	17
SuggestionPolicyStandard.java	Created implementation of SuggestionPolicy interface.	58
Constants.java	Added constants for XML tags.	6
	Total:	525

Referral Network Viewer		
File Name	**Action**	**LOC**
Agent.java	Created thin agent for parsing STATE file.	82
Constants.java	Constants class for XML constants.	172
Edge.java	Edge class for drawing edge connections in Network Viewer.	70
Graph.java	Class for representing a graph within the network.	237
GraphEditor.java	Class for manipulating network graph.	163
Neighbors.java	Thin neighbor class for reading neighbor information from STATE file.	60
Node.java	Class for graphically representing an agent in a network.	159
ReferralNetworkViewer.java	Graphical User Interface for Referral Network Viewer.	679
StateReader.java	Thin state reader for STATE files.	88
	Total:	1710

Appendix D - UML Sequence Diagrams

The following are UML sequence diagrams for the Suggestion Policy and the Interest Adaptation Policy.

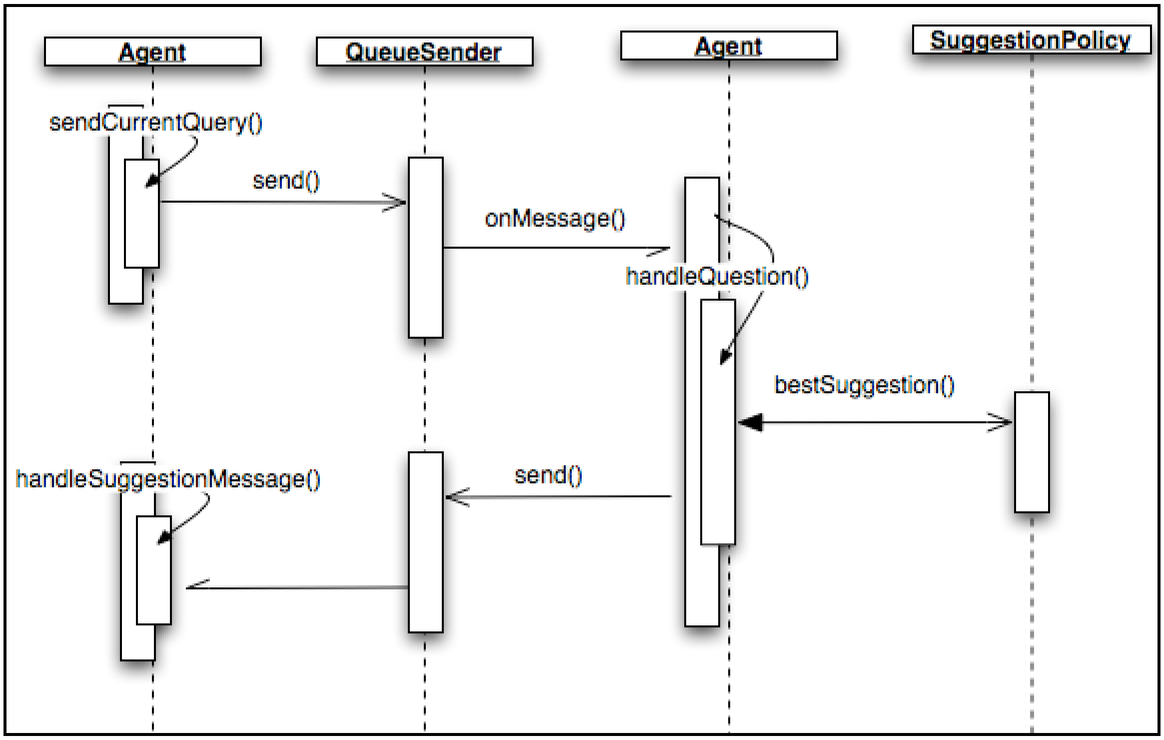

Figure D-1 - Suggestion Policy Interaction

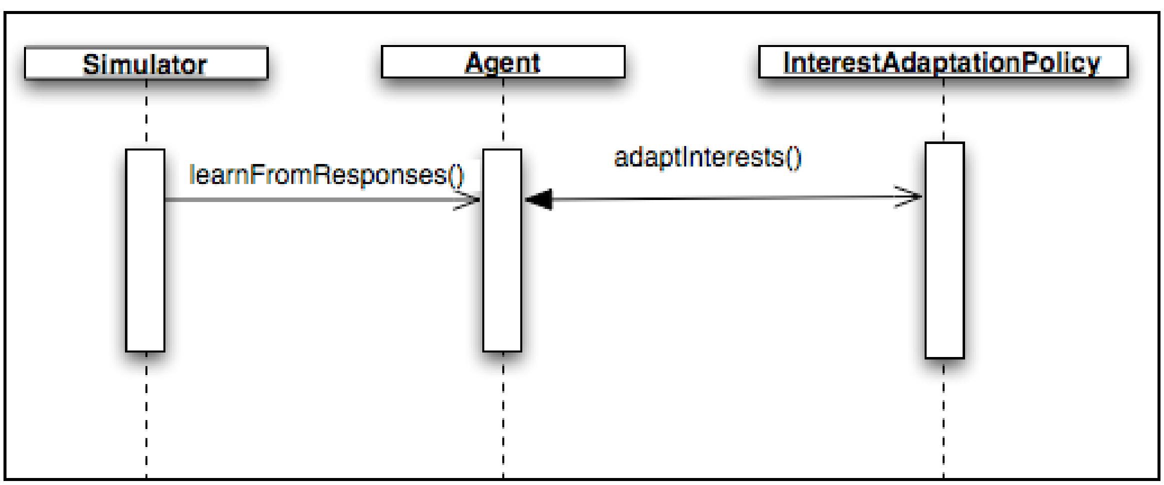

Figure D-2 - Interest Adaptation Policy Interaction

Appendix E - Class Diagrams for Network Simulator

The following are class diagrams for the packages that required the addition or deletion of code for integrating concept relational functionality in the referral network simulator.

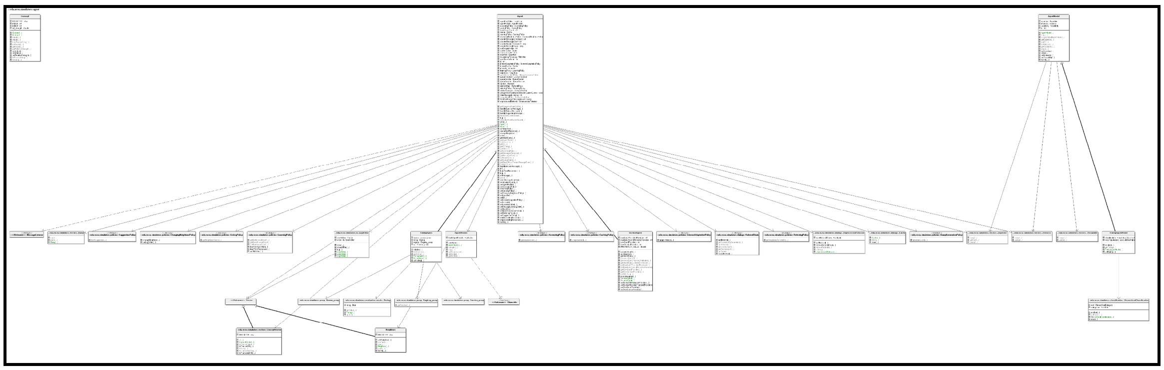

Figure E-1 - edu.ncsu.simulators.agent package

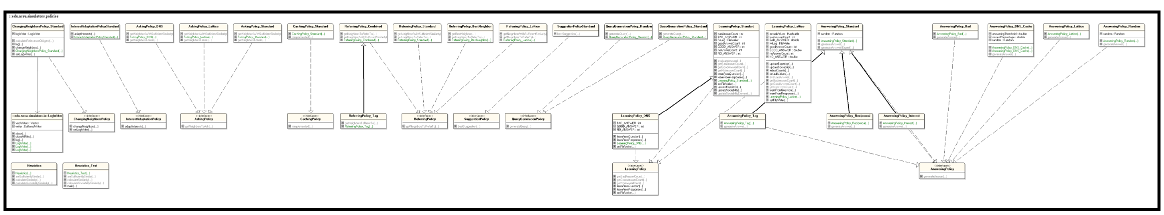

Figure E-2 - edu.ncsu.simulators.policies package

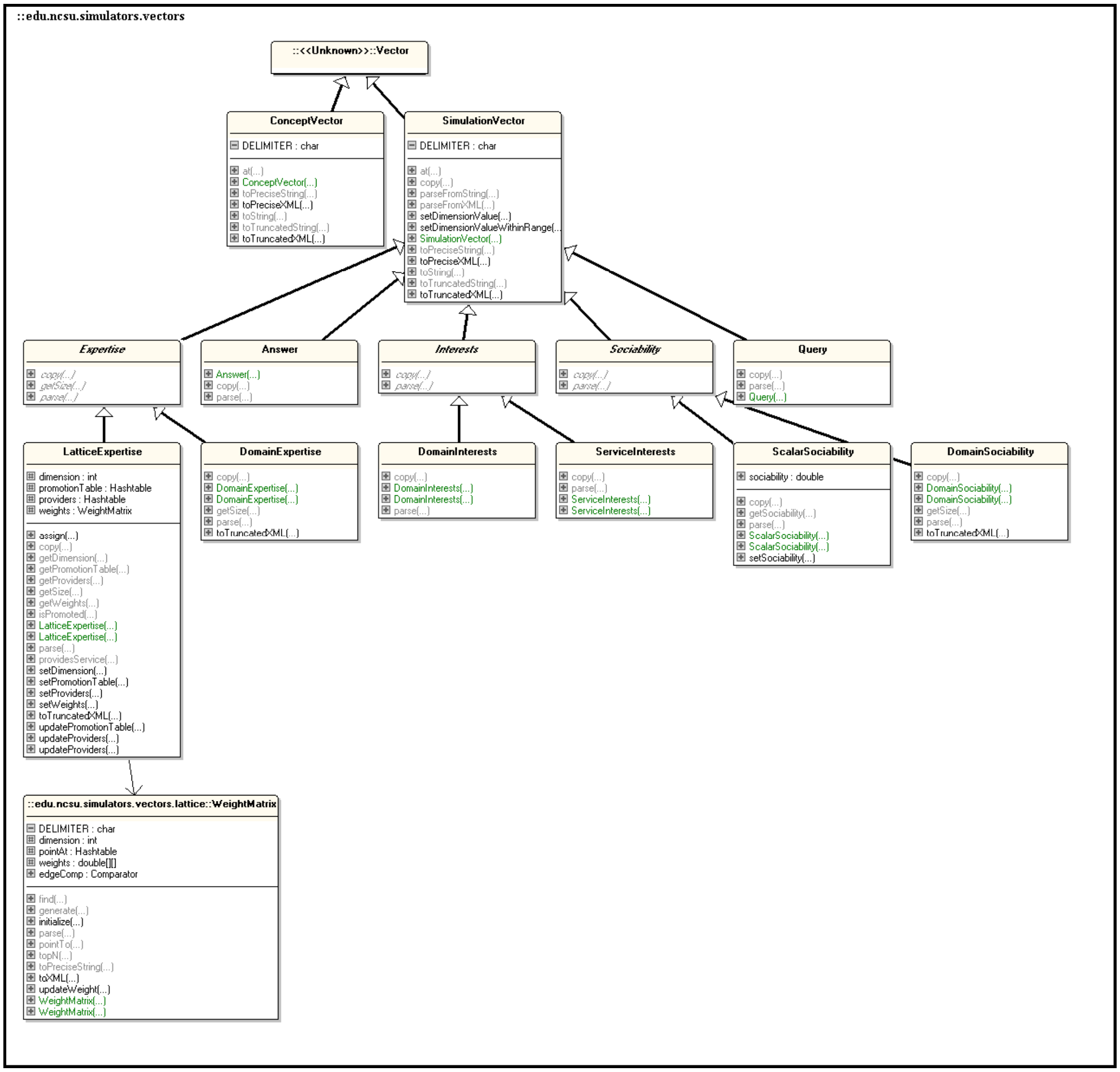

Figure E-3 - edu.ncsu.simulators.vectors package

Appendix F - Referral Network Viewer

The Referral Network Viewer is a graphical user interface add-on application to the Referral Network Simulator. It allows simulations to be run, and the connectivity of the referral network to be graphically shown based on the .STATE output files of the Referral Network Simulator. Each cycle of execution in a simulation outputs a state file which the Referral Network Viewer reads and graphically displays the agents and their neighbours. Information about the agents in the network can be viewed by selecting the desired agent.

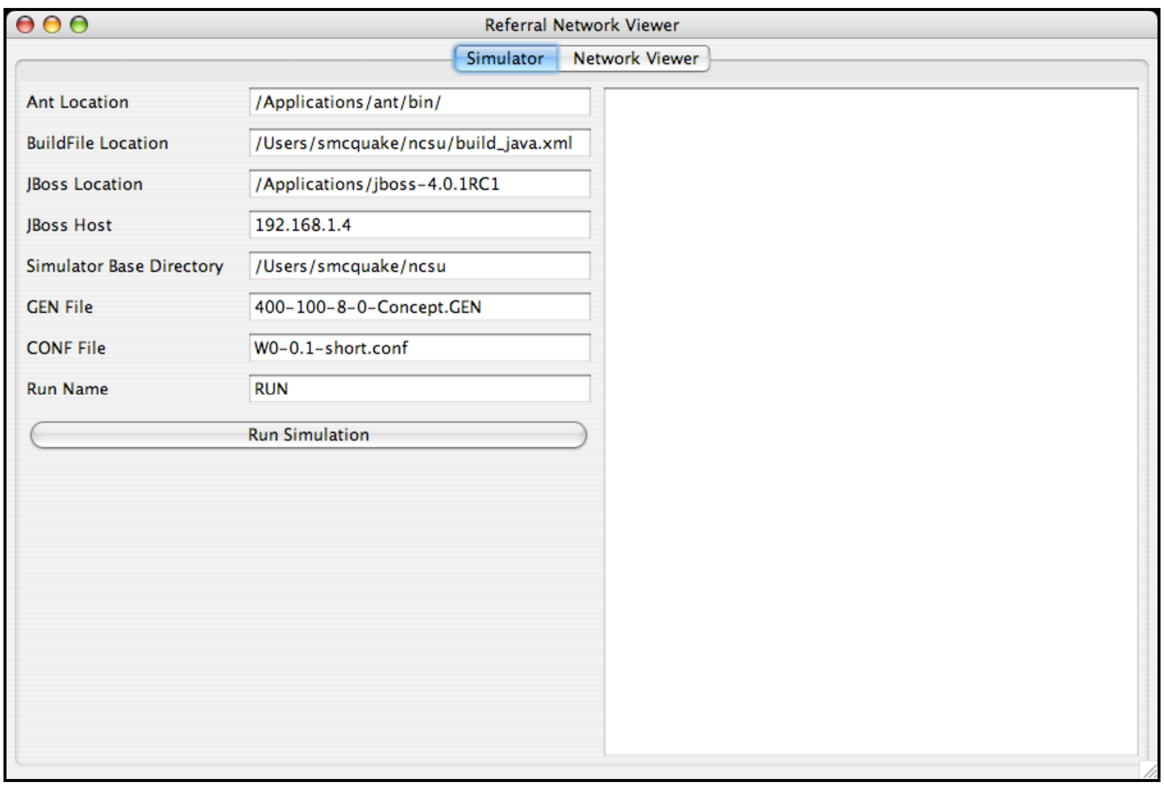

Figure F-1 - Referral Network Viewer Simulator

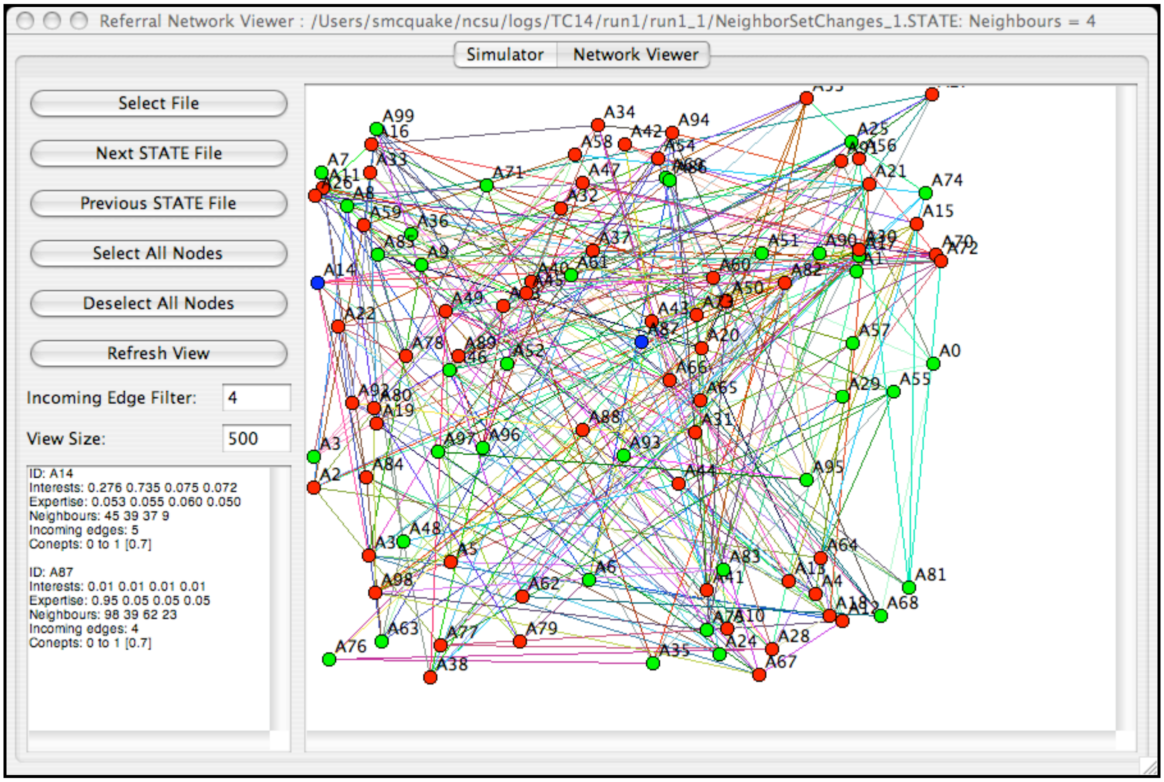

Figure F-2 - Referral Network Viewer