Computing and Informatics, Vol. 25, 2006, 273–290

AN ALGORITHMIC EVALUATION OF INFORMATION SEARCH IN A MOBILE AGENT-BASED DEMAND-ORIENTED INFORMATION SERVICE SYSTEM

Iftikhar Ahmed, Muhammad Jafar Sadiq

Department of Electrical & Computer Engineering
International Islamic University, Kuala Lumpur, Malaysia 53100
e-mail: imadg13@yahoo.com

Manuscript received 26 April 2005; revised 10 February 2006
Communicated by Tomáš Vojnar

Abstract. The rapid advance in information and communication technology (ICT) has given new impetus to shift the information services paradigm from platform centric to network centric computing. Commercial activity is blooming on the internet. The user or customer has become dynamic and has changing tastes and pattern in the demand of service, and desires the service at any time or place. The service provider therefore has to tailor the service provision format to suit the dynamic nature of users of information services. A business cannot afford to ignore the rapid and evolving nature of its customers.

However, the current state of the wide area network services is finding it difficult to respond to constantly changing and heterogeneous demands of modern business, being centralized in nature, with the service provided through a single URL. It is imperative to update the pattern of information service provision and utilization. Faded information field architecture (FIF), reported recently, holds the potential to address these issues, being a demand-oriented architecture. Although research into various aspects of FIF has been reported, we suggest algorithms to characterize the behavior of mobile agents to seek the required information at a given node in the FIF architecture. Simulations were carried out to show the effect of various parameters on the performance of the FIF system.

Keywords: Mobile agents, information systems, wide area network, information fields

1 INTRODUCTION

Networking is the hallmark of emerging technologies in the domain of information and communication technology (ICT) in recent times. In today's business environment information, has become the prime ingredient of a successful business enterprise. The rapid advances in ICT have opened new avenues to experiment with the e-commerce and information service provision and utilization on the wide area networks. The traditional paradigms of business, learning and other essential services are undergoing an evolution. The internet is a huge data repository that is expanding without a central authority. The number of worldwide internet users is predicted to exceed one billion by the end of 2005 [1]. Moreover, 300 terabytes of information is published online every year [2]. As the performance demands of the internet and its usage increase exponentially, the emphasis is shifting from platform centric computing to network centric computing. Information systems have to be designed that are distributed, dynamic and have high assurance and fault tolerance to meet the heterogeneous demands of users. As ICT advances, the dynamics of e-commerce tends to be more data intensive and complex. The traditional business to customer paradigm is encompassing the business to business transactions as well. Companies have to comprehend the trends and demands of the users quickly in order to survive in the competitive environment of today. The expectations of users and customers have soared high and they seek a flawless any time any where service syndrome. In addition these systems are expected to provide high assurance coupled with fault tolerance and timeliness [3,4]. The traditional information services based on the client-server model therefore cannot cope with the rising demand placed by the complexities of data intensive heterogeneous computing. It is therefore imperative to employ new techniques to address these problems in order to reduce the network traffic and improve user's access time to information services. Information fading based on a demand-oriented service architecture is one such technique reported recently [5]. The architecture balances the cost of information provision and utilization on the network by employing push and pull mobile agents (MAs) to service user requests and conduct the business of information provision and utilization on the network. The push-MAs are sent by the service provider (SP) to disseminate the information contents on the FIF nodes. Pull-MAs are sent by the user to search for the required information in the FIF.

In the faded information field (FIF) the service provider distributes the most demanded and popular information contents closer to its vicinity on various nodes in the network. The volume of distributed information on a node is inversely proportional to the distance of information storing node from the service provider. Thus the FIF permits the fairness of information distribution to unspecified users with equal access time. Various attributes of a FIF have been investigated recently [6–9]. We evaluated the pull MA search performance by simulating its behavior under the programmed criterion of a number of algorithms. The results show that the information search efficiency can be preset by using a particular algorithm depending on the nature of the particular service being provided on the network. Simulation results show the viability of proposed algorithms. This paper is organized as follows: Information service system performance requirements are discussed in Section 2, followed by a survey of mobile agents in Section 3. The algorithmic evaluation of pull MAs will be presented in Section 4 and simulation results will be discussed in Section 5. The paper will be concluded in Section 6.

2 CRITICAL PERFORMANCE REQUIREMENTS OF AN ADAPTIVE INFORMATION SYSTEM

The wide area network services are gradually creeping into our daily lives with the number of users constantly on the rise. The information systems now involve a constantly changing environment where stringent performance demands are placed on the network. The businesses are taking advantage of the wide area network facility to offer bargains and put various items on sales and special promotions resulting in the users having much more flexibility and choice available to them making the process of information provision and utilization a complex matter to deal with on the network.

Current information systems tend to handle very high information throughputs. In order for such a system to be efficient and effective, it must be dynamic in nature to handle the stringent service provision and utilization requirements effectively. The two major components of the information service domain, i.e., the information environment and the search systems, have traditionally been kept separate in the past, linked only through a conventional communication interface [10]. However, the current high volume of information traffic handled by the information service systems warrants the appropriate integration of the information and search environments effectively to facilitate the much needed dynamic interaction between the users and the service providers.

It is thus imperative for an information system to respond to both user needs and service provider (SP) requirements to respond in time to these needs in a rapidly changing environment. The main desirable attributes of such a system therefore should be flexibility, reliability and quick reaction time. Only a system with these properties can successfully satisfy both users and SPs in a dynamic network environment.

The traditional data retrieval methodologies focused on the optimization of digging in a huge data base to satisfy a search request. On the contrary, the FIF employs the concept of autonomous provision and utilization of information based on mobile agents. The amount of information to be faded away from SP is a function of network conditions like congestion, popularity of information contents and any other criteria programmed into mobile agents [5].

2.1 The FIF Architecture

The information in a FIF is distributed on various nodes in the system. The information pattern in the field is analogous to the electromagnetic radiation radiated

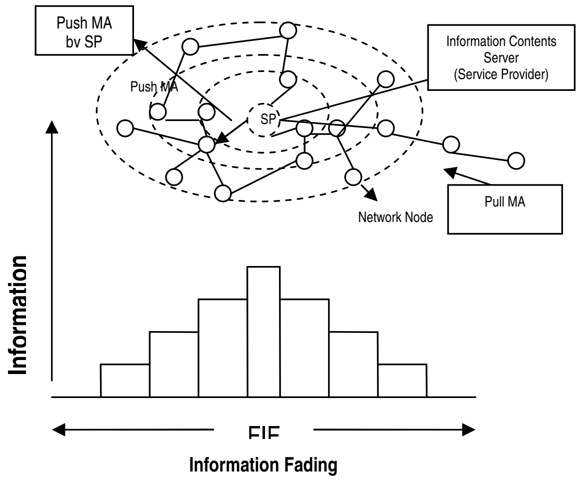

Fig. 1. The layout of a FIF architecture

from an antenna, high in intensity near the antenna and low in intensity away from the antenna. The SP fades away the information contents to the nodes in the field with the richness in information contents inversely proportional to the distance of the node from the SP through push mobile agents (MAs). The user on the other hand sends out Pull MAs seeking the desired information [5, 11].

Information fading has the advantage of increase the reliability of the information. If there is any change in the information, the SP will send push-MAs to update the nodes. The further the nodes, the less the likelihood of the information being updated. Thus, further nodes may have outdated information, and therefore, it is better that those nodes store less information than the nearby nodes. Hence, fading of information improves the likelihood that any information retrieved from the FIF is up-to-date. General scheme of FIF architecture is depicted in Figure 1.

2.2 FIF System Components

The system essentially consists of logically connected nodes through which users and service providers correspond. Mobile agents are used by both parties to acquire and provide information, under evolving/changing situation.

2.2.1 Push Mobile Agents

The mobile agents (MA) generated by service providers are termed push mobile agents (push MAs). The faded information field is created using push mobile agents. Push MAs carry out the function of autonomous coordination and negotiation with other nodes for information fading according to the network traffic status and the level of importance attached to the particular information contents. The level of importance of particular information content is based on its popularity, determined from a high hit rate of query by the information seeker on the network. Push-MAs are sent periodically to update the field. Thus, new information will not need to be indexed, but can be readily made available in this recursive pruning process.

2.2.2 Pull Mobile Agents

Once an information field has been created, users need a means to retrieve this information. Since there is no central index for the field, the users send their own agents to traverse the field and gather information. The pull mobile agents (pull-MAs) are generated by users and they autonomously navigate in search of the required information on the network nodes in a step by step fashion. Since a pull-MA is autonomous, it can navigate the network until it meets its user's requirements, even if the user goes offline. The pull-MA can simply deliver the message through other means, or when the user comes back online.

Once the required information is located, these agents report back to the information seeking source. The push- and pull-MAs have no direct correspondence with each other. A user will issue a pull-MA with his/her search criteria. The degree of complexity of pull-MAs may vary to cater for both advanced and less advanced users searching for the information. This owes to the inherent flexibility of mobile agents being software programs. Thus, pull-MAs can potentially search exactly what the user wants and filter out the unwanted sites resulting in an efficient search.

2.2.3 Nodes

An information provider sends out push-MAs, which distribute the information provider's information to a group of computers in the network termed "nodes". It is a platform for both storage of information and program execution. The users seek the desired information by sending out pull-MAs that visit the network nodes. It monitors the local information-based system conditions and autonomously makes decisions for allocation requests by the SP. The information contents are handled at the node by a resident software program providing the necessary execution and navigation environment for mobile agents [12–13]. The nodes are responsible for negotiating with the push-MA to decide what and how much information is required to be stored on the node. The nodes that store a server's information constitute the faded information field of the server. These nodes help reduce congestion as many user requests will be satisfied at the nodes, reducing the number of requests made to the service provider and thus reducing the load on the service provider. Each subsystem is autonomous in terms of control to execute its operations and coordination with other nodes under evolving network conditions.

It may be pointed out that a node may be shared among many information providers, who may all use the node as part of their own faded information field. In such a case, faded information fields may overlap, resulting in an aggregated global faded information field.

2.3 Communication Format in the FIF

The conventional communication techniques incorporate the destination address to send and receive the required data on the network and therefore cannot cope with the evolving conditions in a heterogeneous network environment where the state of nodes, the status of the SPs and the stability of connections are highly unpredictable as the user demand to access the information is dynamic in nature. FIF therefore employs a different technique referred to as content code communication technique [5].

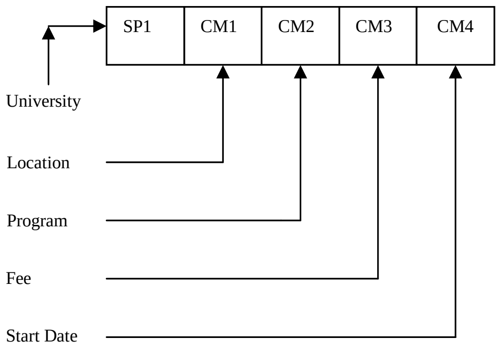

Fig. 2. The message format in a FIF architecture

An example of content code communication is depicted in Figure 2. The information about a university is structured according to the degree of importance in this figure. SP1 specifies the university name followed by CM1 indicating the location of university with respect to a particular country and city. CM2 depicts the programs offered by the university. The message format components thus lead to the breakdown of the principal source of information available on the SP into its uniquely defined characteristic codes CMs. The information content is broken down to specify its characteristics code CHs which correspond to the information properties of CM [14]. Push MAs are sent out in the FIF by SPs specifying Content Codes (CMs) of information to the nodes using the message format as shown in Figure 3. The pull MA carries out a negotiation process with the resident program at the particular node immediately after its arrival at that node. As a pre-requisite of the process it registers its information search requirements in the form of CMs and CHs. The pull MA is then appraised of the local conditions at the node in terms of existing network conditions and the state of the information update at the node at that time. The agent then decides whether to continue its quest for the required information in the FIF or to traverse the same path back to the sender in case the requisite search request has been satisfied at that node. The selection of information storage/allocation is autonomously carried out by the nodes based on CMs. Similarly the pull MAs sent out by the users search for the required information based on CMs.

3 A SURVEY OF MOBILE AGENTS

A distributed environment is most suitable for the employment of mobile agents. The details of such applications in a distributed environment can be found elsewhere [15–17]. Only a brief introduction of mobile agents and their role in a network based information system will be discussed in this paper. The term "mobile agent" is often context-dependent and has two separate and distinct concepts: mobility and agency. The term "agency" implies having the same characteristics as that of an agent. These are self-contained and identifiable computer programs that can move within the network from node to node and act on behalf of a user or other entity. These can halt execution from a host without human interruption [18]. A mobile agent does not need to maintain communication with its source. Therefore, it is an autonomous entity. It may have some or all of the following properties [19]:

Autonomous – the program exercises control over its actions.

Reactive – it responds in a timely fashion to changes in environment.

Goal-oriented – it does not simply react to the environment.

Communicative – it communicates with other agents or with devices.

Adaptive – it changes its behavior based on past experience.

Mobile – it is able to transfer itself from one host to another.

Flexible – it is able to cope with unexpected situations.

The current distributed network environment is based on the traditional client-server paradigm. In the case of mobile agents employed in a network, the service provision and utilization can be distributed in nature and is dynamically configured according to the changing network performance metrics like congestion. Mobile agents are typically suited to applications requiring structuring and coordinating wide area network and distributed services involving a large number of remote real time interactions [20].

4 INFORMATION SYSTEM SIMULATION

The FIF structure was simulated using the following parameters:

- A total of 50 web servers acting as service providers (SPs) in the FIF.
- Each web server stores between 7 and 10 categories of information out of the total of 50 categories of information at random.
- Routers were generated at random using polar coordinates with maximum radius specified as 50 units. The total of 200 routers were created in the simulated information system.

4.1 Algorithms for Pull-MA Negotiation for Information Search

Since a fast response time is required for a viable information system search query, pull-MAs must visit as few nodes as possible so as to minimize the service time in their efforts to locate the desired information on the network. Since each SP creates its FIF in a shared network environment, the various FIFs of each service provider overlap, implying that if a pull-MA visits a single node, it can become aware of several other service providers on the network (Figure 3).

An algorithm is therefore needed to enable the pull-MA to select a set of nodes that allows it to find the maximum possible amount of information in its operating field of information. The following algorithms were suggested and subsequently tested in this regard.

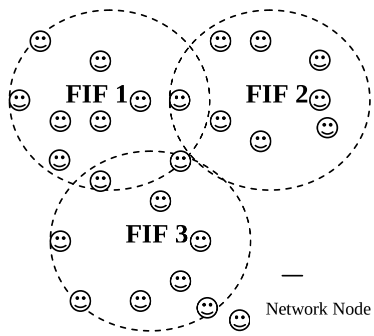

Fig. 3. The possible overlap of various FIFs in the information system

Random – The pull-MA chooses its next destination node at random from a set of all available unvisited nodes. This algorithm is analogous to the random selection of pages in a CPU's cache being as good as other algorithms in improving the cache's performance. This is the least computationally intensive strategy.

Weighted – The pull-MA chooses a node at random from all available unvisited nodes in the FIF, but gives more weight to nodes at a medium distance from the current node. It is anticipated that if the pull-MA hops in this manner, it will cover a decent subset of nodes in the FIF.

In order to generate this distribution, there are many possible mathematical functions. It was chosen to use an inverted x^2 graph, shifted towards the center of our range of numbers, for the lower random numbers. For the higher random numbers, a similar shifted x^2 graph was used. The following equation was used to generate such a distribution:

$$S = 0.5 \times N \times (1 - (r/(L/2) - 1)^2) r < L/2$$

$$S = 0.5 \times (N - 1) \times (1 + (r/(L/2) - 1)^2) r > L/2$$

where
S is SP chosen from nodes sorted by distance
N is the number of nodes in the shared FIF network
r is random number
L is maximum value of the random number.

This arrangement results in the distribution shown in Figure 4 termed a quadratic strategy. The figure shows that a larger proportion of random numbers depicts a node that is at a medium distance from the SP, since that corresponds to the flat portion of the graph, while the steep part of the curve shows that the nodes near and far are less likely to be selected, since a low value of random numbers leads to the selection of those nodes in the FIF.

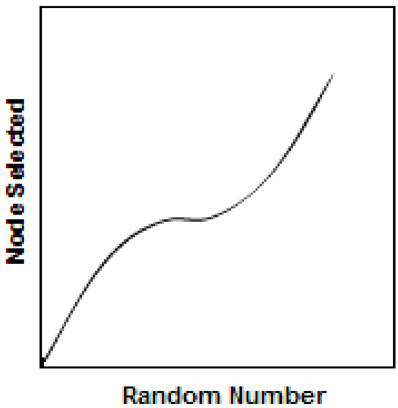

Fig. 4. Quadratic strategy

Limited – In this case the pull-MA chooses a node at random, the distance being weighted towards nodes that lie at medium distance from the SP. However, all nodes too close or too far from the current node being visited are excluded from the list. The nearby nodes are excluded since they are likely to be a part of the same FIF as the current node, and therefore they are not likely to contain updated information. The furthest nodes are rejected in order to prevent a situation where the pull-MA hops between two extremes. This is the most computationally intensive strategy, but it is also expected to produce optimum results for the information search. The curve is more flat compared to Figure 3 for quadratic strategy. Moreover, the nodes are relocated initially with no value of random number. This is so since the nodes on the extreme end of the FIF are excluded.

To generate this distribution, it was chosen to use an x^3 graph shifted to center within the range of random numbers. The following equations were used to generate such a distribution:

$$LL = N \times (100 - P)/200$$
$$S = 0.5 \times P/100 \times N \times ((r/(L/2) - 1)^3 + 1) + LL$$

where

LL is lower limit of acceptable nodes

P is percentage of nodes from which next hop is selected.

This gives a distribution as shown in Figure 5.

4.2 Algorithms for Information Fading on Nodes

The degree of information fading depends on the distance of a node from the SP. This dependency can be represented by various functions. However, the following three functions were tested in the simulations. The nature of the information content to be stored is decided randomly by the node after negotiating with the pull MA.

Straight Line – The information is faded proportionally with the distance, where a node at zero distance would have 100 % of the information and a node at the FIF edge would have 0 % of the information. However, since the furthest node of the FIF is still a member of the FIF, it should still retain a portion of the information.

The following function is used, which is basically a downward sloping straight line:

$$P = (L - 100)/W \times D + 100$$

where

P is percentage of information of the push-MA that is stored

L is minimum percentage of information that a node in the FIF is required to store

W is width of the FIF

D is distance of the current node from the source.

An example calculation of percentage of information stored by push-MA on a node is shown in the table below. Keeping the other variables constant, the distance of the information storing node is changed resulting in a decline in the percentage of information stored as the distance of the information storing node increases from the Sp, in unison with the fading strategy.

Quadratic – In order to increase the amount of information available in the FIF, the fading may be done with a quadratic function. The reliability drawback is not severe in a FIF system designed for the application of information retrieval

L (minimum % of information that a node in the FIF is required to store)	W (width of FIF)	D (Distance of current node from the SP)	P (% of information stored by the push-MA)
10	50	10	98.2
10	50	15	73
10	50	30	46
10	50	50	10

Table 1

since, in such a system, the information is fairly static as updates are done infrequently. A typical example would be a shop that discontinues the sale of a certain item. In this strategy, each node in the FIF stores more information from the push-MA than in the previous strategy, thus increasing the likelihood of finding the information. The following function is used to achieve quadratic fading. It is an inverted x^2 graph shifted upwards so that it almost intersects the distance-axis (0 % information stored) at the furthest node.

$$P = (1 - (D/(M \times W))^2) \times 100$$

where

M is multiplier to increase field size so that the perimeter node stores at least some of the push-MA's information.

4.3 Release of Pull-MAs to Find Information

The performance of the FIF-based information retrieval system was tested from the user's point of view in the simulation runs. The sequence of events is given below:

1. **Create pull-MAs**

   A server is selected at random at regular intervals as the source of a pull-MA. The pull-MA is assigned a certain category of information to discover. For each such user query, only one pull-MA is created.

2. **Pull-MA chooses destination**

   The pull-MA chooses a destination by using an algorithm chosen by the system designer and travels to that node.

3. **Pull-MA seeks information**

   The pull-MA searches the node for the category of information it was created to find.

4. **Pull-MA continues in pruning process**

   The pull-MA continues through stages (b) and (c) until it has traversed a predetermined number of nodes, chosen as 10 nodes in our simulation.

5. **Pull-MA report**

   The pull-MA sends its report back to its source, and ultimately terminates.

5 INFORMATION SEARCH RESULTS

Initially, using the quadratic algorithm for pull-MAs to choose their next hop, the performance criteria chosen were the percentage of pull-MAs that managed to discover the required information and the percentage of information that was discovered, which is measured as the percentage of SPs that the pull-MA found out of the total number of SPs that provide the required information. The results are shown in Figures 6 and 7. The results are averages of 1000 pull-MAs.

The fading strategy did not produce the best results, but turned out to be not significantly worse than having no fading at all. This owes to the fact that the SPs continue to send push-MAs in the FIF with updated information. Eventually more and more of the information is stored on the nodes until the other strategies approximate a situation with no fading.

Another unexpected result is that only 21 % of the information requested is discovered, although a larger percentage of requests were satisfied. As will be demonstrated later, increasing the number of pull-MA hops can alleviate this problem.

For the rest of the simulation, the 'no fading' algorithm was used. This algorithm gave better results in addition to being less processor intensive since there is no computation required to determine which information contents have to be stored on a particular node. It is, however, memory intensive.

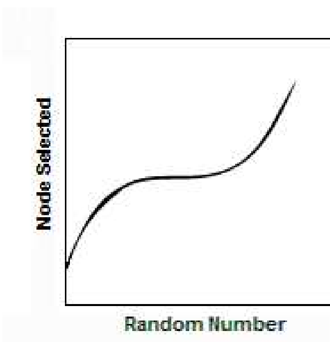

Fig. 5. Limited strategy

Next, the effect of the algorithms used by pull-MAs to choose hops was evaluated. Again, the same performance criteria were used as discussed above. The results are shown in Figures 8 and 9 respectively. The results show that using the quadratic function to choose nodal hops by the pull MA turned out to be the best strategy. The difference of information search performance between the random

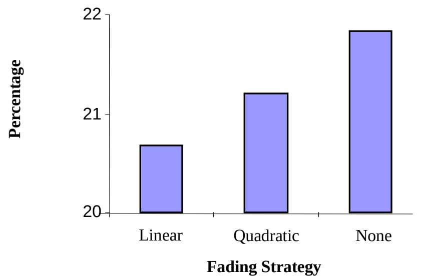

Fig. 6. Percentage of requests satisfied

(turned out to be the worst) and the quadratic is only about 1 %. The random strategy is, however, the least computationally intensive since there are no complex computations involved.

In order to improve the performance of the pull-MAs, the number of hops traveled by the pull-MA was increased in the simulation run. In this case, in addition to the effects on information retrieval, the time taken by the pull MAs to traverse the nodes should also increase.

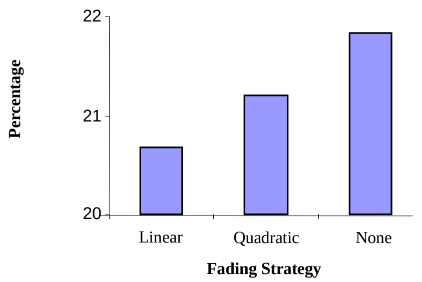

Fig. 7. Percentage of information discovered

The effect on the service time is shown in Figure 10. It was observed that the resultant time increase in the increase of number of hops by the pull MA is linear. The significant observation is that there is a 1.2 s increase in time over a 12 node increase in number of hops. This time increase may or may not go unnoticed by the average user. Figures 11 and 12 show that increases in the number of hops by pull MA improves the performance of the system. A 22 % increase in the number of requests for information is observed to be satisfied for a corresponding 1.2s increase in service time. Similarly, there is a corresponding 26 % increase in the amount of information discovered.

It is also observed that the rate of increase of requests satisfied decreases as the hops are increased. As more pull-MAs find the information they are looking for,

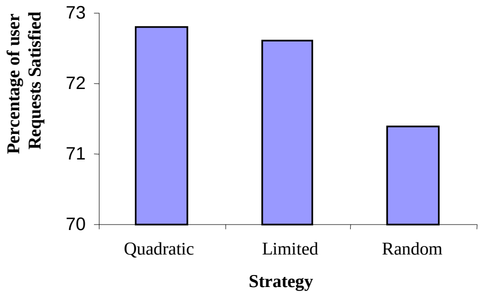

Fig. 8. Percentage of user service requests satisfied

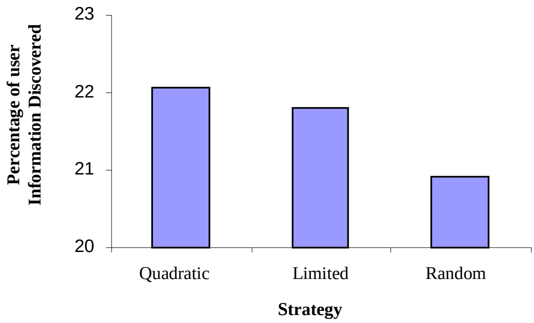

Fig. 9. Percentage of information discovered

there are less pull-MAs left that have not yet found their information, so the effect of increasing the number of hops decreases.

6 CONCLUSION

FIF architecture contributes to the assurance of an information service system in addition to the enhancement of system fault tolerance and timeliness. In the context of the three major components of a FIF, the pull MA information search behavior was characterized in this work.

A number of algorithms were devised to characterize the behavior of Pull MAs to seek information in a FIF. Simulations were performed to verify the theoretical predictions. It was demonstrated that the random hops selected in the course of limited strategy algorithm was the most computationally expensive. However, it produced the best search results compared to the quadratic and straight line strategy. The quadratic fading favored the essence of FIF with more information stored near the service provider. Each algorithm is therefore suited to a particular set up criteria for a FIF. The pull MA search criteria thus can be modified suiting the particular application for which the FIF is set up.

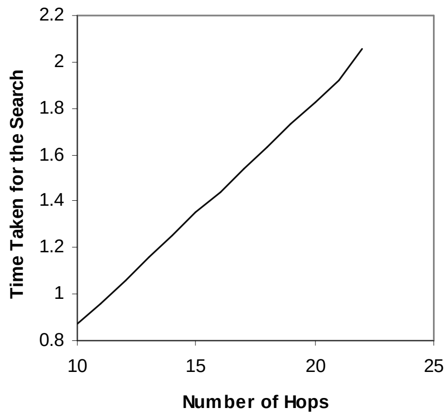

Fig. 10. Effect of number of pull MA hops on service time

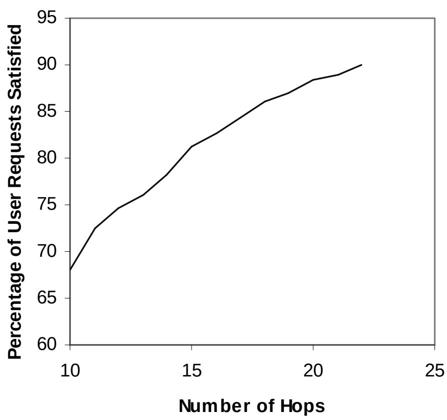

Fig. 11. Percentage of user requests satisfied

Some e-commerce applications may prefer to opt for quadratic or straight line fading for their information dissemination on the nodes because they have rapidly changing information. In other applications where the information does not expire often, it could be better to not fade the information to increase system performance.

The increase in the number of pull MA hops aids in enhancing user search requests resulting in the intangible benefit of network congestion reduction. These algorithms demonstrate the high potential coupled with flexibility of a FIF in addressing a number of challenging issues with respect to balancing the user demands and service provision in a wide area network information service system.

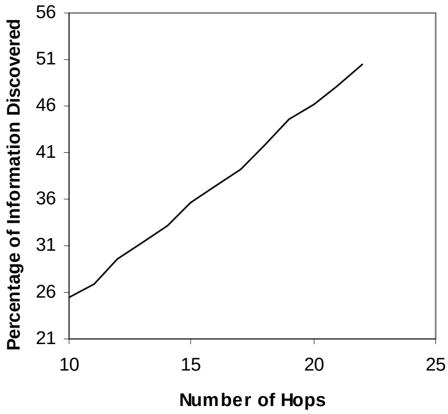

Fig. 12. Percentage of information discovered

REFERENCES

[1] Roberts, L. G.: Beyond Moore's Law: Internet Growth Trends. IEEE Computer, Vol. 33, 2000, No. 1, p. 117.

[2] Lyman, P.—Varian, H. R.: How much Information. Journal of Electronics Publishing, Vol. 6, 2000, No. 2.

[3] Ingham, D. B.—Shrivatava, S. K.: Constructing Dependable Web Services. IEEE Computing Magazine, pp. 25–33, 2000.

[4] Yen, I. L. et al.: Toward Integrated Methods for High Assurance Systems. IEEE Computer Magazine, pp. 32–34, 1998.

[5] Ahmad, H. F.—Arfaoui, H.—Mori, K.: Autonomous Information Fading by Mobile Agents for Improving User's Access Time and Fault Tolerance. Proceedings, 5$^{\text{th}}$ International Symposium on Autonomous Decentralized Systems, pp. 279–283, 2001.

[6] Ahmed, I.—Sadeq, J.: Characterization of Information Distribution Fairness in a Mobile Agent-Based Adaptive Information System. Sixth International Conference on Web Integration and Services. September 27–29, 2004, Jakarta, Indonesia.

[7] Ahmed, I.—Sadeq, J.: Nodal Congestion Analysis of an Autonomous Mobile Agent-Based Distributed Information Service System for Efficient Utilization of Information Contents. International Journal of Information Technology, Vol. 10, 2004, No. 1.

[8] Ahmed, I.—Sadiq, J: A Novel Autonomous Architecture for the Integration of Information and Control in an Automated Manufacturing System. Presented in the International Conference on Intelligent Agents, Web Technologies and Internet Commerce, July 12–14, 2004, Gold Coast, Australia.

[9] Ahmed, I.: Balancing the User Demand and Information Provision in a Web-Based Linguistic Learning System Employing Faded Information Field Architecture. In the

ASIA CALL International Conference on Information Technology and Linguistic Learning, Bangkok, Thailand, December 3–5, 2003.

[10] Bestavres, A.: WWW Traffic Reduction and Load Balancing through Server-Based Cashing. IEEE Concurrency Magazine, pp. 56–67, 1997.

[11] Arafoui, H.—Mori, K.: Autonomous Navigation Architecture for Load Balancing User demands in Distributed Information Systems. IEICE Transactions on Communications, E84-B, Vol. 10, pp. 1085–1093, 2001.

[12] Dalmeijer, M. et al.: A Reliable Mobile Agent Architecture. Proceedings OORT-DCS, pp. 64–72, IEEE, 1998.

[13] Silva, F. A. et al.: An Approach for Providing Mobile Agent Fault Tolerance. Springer, 1998.

[14] Tasaka, M. et al.: Autonomous Process or Go Navigation Technology for Congestion Avoidance in Distributed Information Service systems. Proceedings of the 23rd International Conference on Distributed Systems Workshops (ICDCSW03) 2003, IEEE.

[15] Karmouch, A.—Pham, V. A.: Mobile Software Agents: An Overview. IEEE Communications Magazine, Vol. 36, 1998, No. 7.

[16] Perdikeas, M. K. et al.: Mobile Agents Standards and Available Platforms. Comp. Net., Vol. 31, 1999, No. 19, pp. 1999–2016.

[17] Lange, D. B.—Oshima, M.: Seven Good Reasons for Mobile Agents. Communications of the ACM, Vol. 42, 1999, No. 3, pp. 88–89.

[18] Cao, J.—Chan, G. H.—Jia, W.—Dillon, T.: Check-Pointing and Rollback of Wide-Area Distributed Applications Using Mobile Agents. Proceedings, International Parallel and Distributed Processing Symposium, San Francisco: IEEE Computer Press.

[19] Covaci, S.: Autonomous Agent Technology. Proceedings of The Fourth International Symposium on Autonomous Decentralized Systems. pp. 255–257, 1999.

[20] Cao, J.—Chan, G. H.—Jia, W.—Dillon, T.: Check-Pointing and Rollback of Wide-Area Distributed Applications Using Mobile Agents. Proceedings, International Parallel and Distributed Processing Symposium, San Francisco; IEEE Computer Society Press, April 2001.

Iftikhar Ahmed graduated in avionics engineering from NED University of Engineering & Technology Pakistan in 1981. He then served as an engineer in Government of Pakistan till 1987 when he was awarded a National Talent Scholarship by The Ministry of Science & Technology Pakistan for higher studies abroad. He was awarded a M. Sc. in optical electronics and a Ph. D. in electrical engineering by Strathclyde University (UK) in 1989 and 1993, respectively. He served in various R & D Establishments in Pakistan and worked as an Associate Professor in the Computer Science Department of Bahria University (Pakistan) for two years before moving to Malaysia in 2000 where he is currently serving as an Associate Professor in the Computer Engineering Department of International Islamic University. His research interests include autonomous mobile agents in information service systems, fault tolerant computing and autonomous avionics systems.

Muhammad Jafar Sadeq graduated with a Bachelor's in computer and information engineering from International Islamic University Malaysia in 2000. He is currently undergoing a Master's of Science Program in computer and information engineering at the same university. His research interests include distributing computing and adaptive information systems.