

# Domain-Based Proxy for Efficient Location Tracking of Mobile Agents

Sanghoon Song and Taekyoung Kwon

School of Computer Engineering, Sejong University, Seoul 143-747, Korea  
E-mail: {song,tkwon}@sejong.ac.kr

**Abstract.** The provision of location tracking for mobile agents is designed to deliver a message to a moving object in a network. Most tracking methods exploit relay stations that hold location information to forward messages to a target mobile agent. In this paper, we propose an efficient location tracking method for mobile agents using the domain-based proxy as a relay station. The proxy in each domain is dynamically determined when a mobile agent enters a new domain. The proposed method exploits the domain-based moving patterns of mobile agents and minimizes registration and message transfer costs in mobile agent systems.

## 1 Introduction

Mobile agents are software objects that can migrate across the network representing users in various tasks. The most attractive applications are e-commerce, network management, and real-time control in many distributed system areas. The code mobility provides many advantages. When the data volume in a remote host is very big, mobile agent systems can save the network bandwidth tremendously. Instead of requesting whole data through network connection, a mobile agent migrates to the target host, filters through the data locally, and brings back only the result. For real-time control of remote devices, the traditional client/server design is not a good candidate due to the irregular network delay. However, a mobile agent that has migrated to a remote system can directly control the target system in real time. Mobile agents are also useful for applications in wireless environments, such as laptops or PDAs, that can be disconnected at short notice [1,2,3].

Apart from these advantages, there are many problems to be solved. Most of the research focuses on providing system support for the security of mobile agents, reliable communication with fast moving mobile agents, and efficient location management [4,5]. The typical application of mobile agent is to bypass the communication link and to exploit local access to resources on a remote server. Thus one may argue that the communication issue is not important. However, we have several situations that require efficient communication with mobile agents. For example, a user may launch a mobile agent with some parameters directing the behavior of the agent and may want to change the parameters

later due to changes in the context that determined their creation [4]. Mobile agent systems should have location tracking functions to transfer messages only to target agents. Whenever a mobile agent migrates to a new node, the new location information should be registered somewhere in the system.

In this paper, we propose an efficient location tracking method called Domain-Based Proxy. A domain consists of a group of hosts that are close to each other, measured by the number of hops in a network. Mobile agents can reduce the length of their migration paths by visiting the hosts in the same domain first, rather than selecting hosts randomly. The proposed method exploits the domain-based moving patterns of mobile agents and minimizes the registration and message delivery cost. We do not consider the chasing problem that occurs when mobile agents migrate so frequently that relay stations keep forwarding messages to hosts where the target agent no longer stays [4].

Section 2 explains background information in the field and motivation for this work, and section 3 explains the idea of domain-based proxy and its effectiveness in reducing the registration and message delivery costs. In section 4, we discuss simulation results with various parameters. Finally, we present our conclusions in section 5.

## 2 Background and Motivation

In recent years, there have been several protocols on location tracking of mobile agents. The common ground of these protocols is to have relay nodes that hold the current location of an agent and forward messages to it [5]. There are three different forms of relay nodes: a relay node that is fixed, a relay node that is movable, and a chain of relay nodes that are linked with a pointer. The relay nodes provide location transparent service to senders so that senders do not care about the current locations of agents and their movements. We assume that senders know the homes of mobile agents and the home nodes also act as relay nodes.

### 2.1 Home

The home node of a mobile agent carries the current location information of the agent and forwards messages from senders to the destination agent. Whenever a mobile agent migrates to a new host, it registers its current location with the home node. The protocol is simple, but the registration cost is high when the agent is far away from the home. Since there is no other relay node between the home and the destination node, the message delivery cost is low.

### 2.2 Pointer Chain

Each node on the migration path of a mobile agent keeps the pointer to the next node on the path. The home node becomes the first node in the pointer chain. When a mobile agent migrates between the nodes within a domain that

is far away from the home, the registration cost is low compared to the Home method. However, the message delivery cost becomes very high, since messages are forwarded through all the nodes on the chain.

### 2.3 Mailbox

Each mobile agent has a mailbox that relays messages to it. The agent registers its current location with its mailbox whenever it moves to a new node. The mailbox decoupled from the agent can reside in different hosts and moves independently [3]. Since it is movable, the home node should have the updated location information for the mailbox. After getting the current mailbox location from the home node, the sender delivers messages to its mailbox that can relay the messages to the target agent. The method of mailbox movement has not been published yet. Without an efficient mailbox movement, the performance will be nearly the same as the Home method.

## 3 Domain-Based Proxy

We define the term domain as a group of hosts that are close to each other in network structure. Each host belongs to only one domain. A proxy is determined at the time of entry to a new domain. The first host that a mobile agent visits in a new domain serves as a proxy in the domain. Whenever the mobile agent moves to a host in the same domain, it registers its location with the proxy in the domain. If the mobile agent migrates to a host in another domain, the host becomes a proxy in the new domain and registers with the proxy in the previous domain. Since another mobile agent can enter the same domain by visiting a different host, there may exist several proxies in one domain. A proxy has a data structure that points to the proxy of the next domain to which the mobile agent has already migrated, or points to the host which the mobile agent is currently visiting. We can lower the registration cost if hosts in a domain are close to each other. Messages are forwarded through the proxy chain and the last proxy in the chain forwards them to the host in which the mobile agent stays.

Figure 1 shows the proxy chain after the mobile agent migrates to  $h_{10}$ . It followed the path  $P_1, h_2, h_3, h_4, P_2, h_6, h_7, P_3, h_9, \text{ and } h_{10}$ . Proxy  $P_1, P_2, \text{ and } P_3$  represent the first host the mobile agent visited in each domain respectively. The solid line indicates the message-forwarding path. The message-forwarding path is relatively shorter than that in the Pointer Chain method. Assuming that the inter-domain distances are relatively far, the registration costs within a domain are lower than those between domains. Thus we can reduce the registration cost for migrations within a domain. If the domain of the current host is equal to that of the previous host, two hosts are in the same domain. Hence, the current host will register with the proxy of this domain. If the domains are different, the current host becomes the proxy of the new domain. Consequently, the proxy of the new domain will be linked to the proxy of the previous domain.

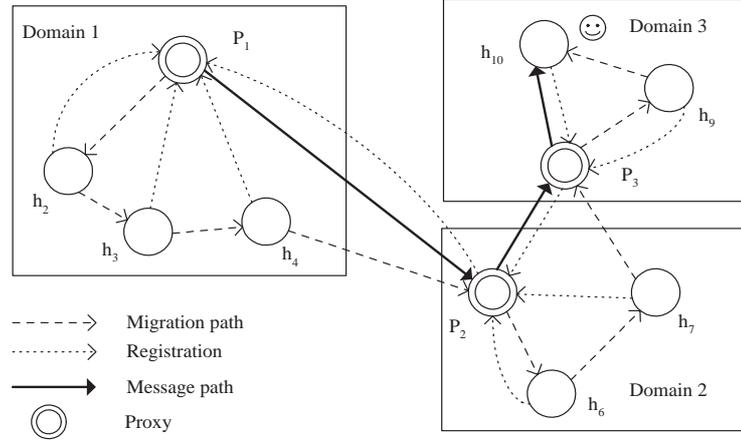


Fig. 1. Migration in the Domain-Based Proxy scheme

### 3.1 Compacting Proxy Chain

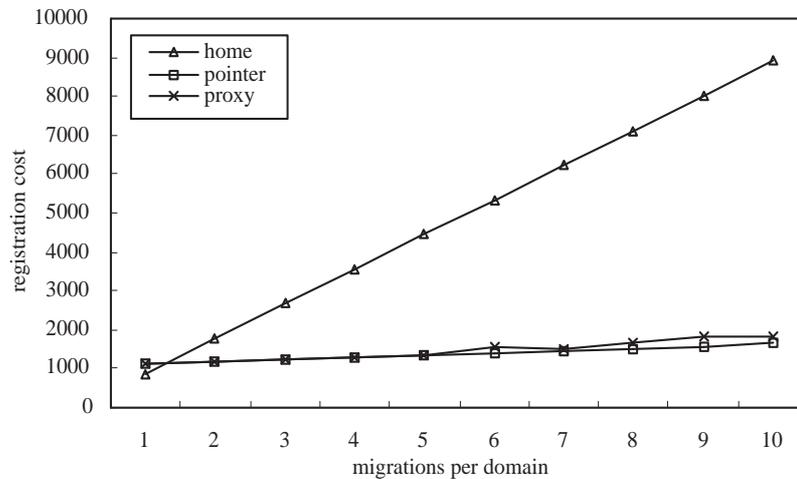
Messages are forwarded through the proxy chain and delivered to the host where the target agent is. When the number of messages is high, the long proxy chain may monopolize the overall cost by overshadowing the low registration cost. Given the estimation of the number of messages, we can compact the proxy chain to reduce the message delivery cost.

We define following parameters to describe compacting the proxy chain.

- $N$  : the expected number of messages to receive
- $D_0$  : the distance from the home to the current node
- $D_p$  : the distance from the home to the current node through the proxy chain
- $R_0$  : the registration cost at the home node
- $R_p$  : the registration cost at the proxy node

After executing the mobile agent many times, we may predict the expected number of messages to receive. We assume that each node knows the distance from all other nodes including the home node. To evaluate the distance  $D_p$  from the home to the current proxy through the proxy chain, each agent should carry the  $D_{p-1}$ , which denotes the distance from the home to the previous proxy through the chain. Distance  $D_0$  and  $D_p$  can determine the registration cost  $R_0$  and  $R_p$  respectively. With these parameters, we can estimate the registration and message delivery costs for the cases of proxy chain compacted and proxy chain without compacted. Since the migration within a domain does not change the proxy, only the case of migrating to a different domain requires the following decision on whether to compact or not.

$$N * D_p + R_p > N * D_0 + R_0$$



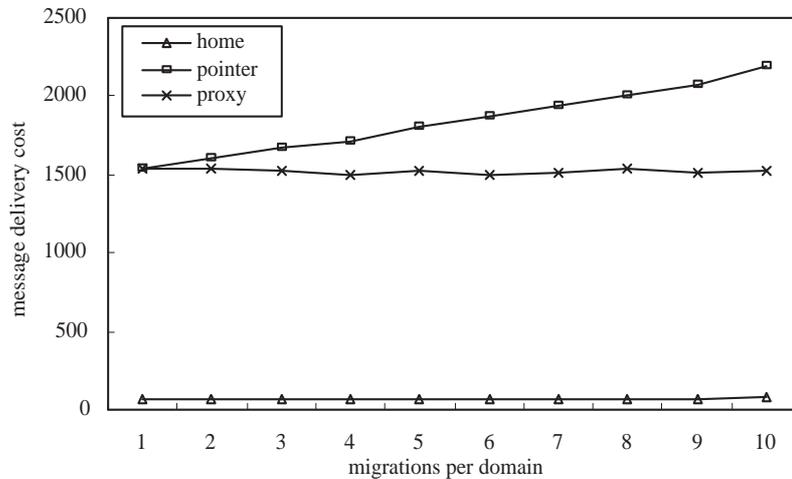
**Fig. 2.** Registration cost

As an extreme case, an agent may migrate through a path such that  $D_p$  equals to  $D_0$ . Since  $R_p$  is definitely smaller than  $R_0$  for this case, keeping the proxy chain can reduce the overall costs. In most cases, the distance through the proxy chain,  $D_p$ , is larger than the direct distance,  $D_0$ , and the registration cost  $R_p$  is smaller than  $R_0$ . When no message is expected to arrive, the compaction is not necessary. However, as the expected number of messages increases, we can minimize the overall cost by compacting the proxy chain.

## 4 Simulation and Discussion

We assume that the network structure is in the form of a two dimensional grid and the location of each host is expressed by the coordinate  $(x, y)$  in the grid. The distance between two hosts can be calculated with the geometrical distance of two coordinates in the grid. The grid is partitioned to form domains. Each domain is also a square grid with smaller size. The costs for location registration and message delivery depend on the distance between two hosts and the data size. We assume the data size for registration is about a quarter of the data size in message delivery [5].

In the simulation, we calculate the registration and message delivery costs that occur in the host on the migration path. Starting from a randomly selected host in a domain, we continue to move to randomly selected hosts in the same domain until the number of migrations per domain is met. After completing a given number of migrations in a domain, we move to a new domain and visit hosts in the domain. Since the migration pattern can be different in various applications, the domains are selected randomly for simulation. We assume that



**Fig. 3.** Message delivery cost

all the hosts involved are lightly loaded and there is no additional delay in processing messages. We repeat the above simulation by varying the number of migrations per domain and the number of messages.

Figure 2 represents the registration costs against the number of migrations per domain. We observe the Proxy and the Pointer Chain method outperform the Home method as the number of migrations per domain becomes larger. The reason is that the inter-domain distances are greater than the distances between hosts in a domain. When the number of migrations per domain equals one, we do not expect any performance advantage either in the Proxy or the Pointer Chain method. Since the next domain to visit is selected randomly, the distance from the visiting domain to the home node will be comparable to the average inter-domain distance.

Figure 3 shows the message delivery costs against the number of migrations per domain. Since the Pointer Chain method delivers messages through all the relay stations that are on the migration path, the message delivery cost increases linearly as the number of migrations per domain increases. However, the message delivery cost of the Proxy method remains constant because the proxy chain length does not increase even if the number of migrations per domain increases. The Home method delivers messages directly to the target agent without any relay station and the message delivery cost remains minimal.

Figure 4 shows combined costs against the number of messages with the number of migrations per domain fixed at 13. As the number of messages increases, the message delivery costs in the Proxy and the Pointer Chain method begin to dominate the registration cost and monopolize the combined cost respectively. Since the pointer chain length is longer than the proxy chain, the increase rate

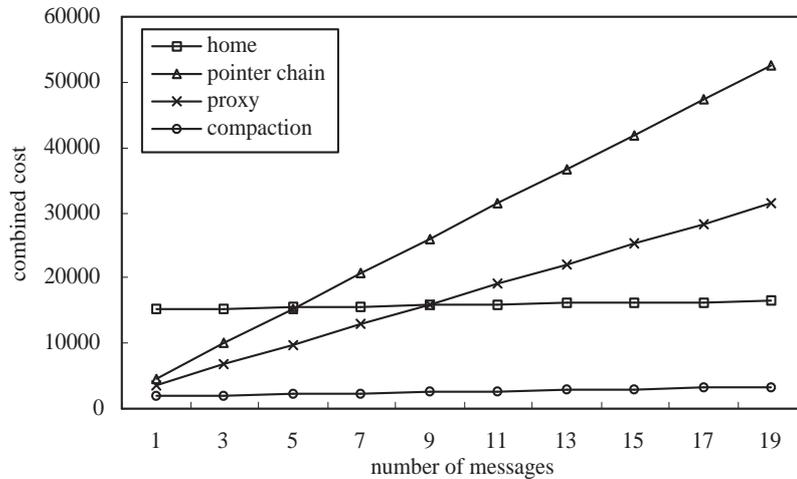


Fig. 4. Combined cost against the number of message delivery

of the Pointer Chain method is steeper than that of the Proxy method. For the Home method, however, the registration cost still dominates the message delivery cost and the rate of increase is unnoticeable. The Compaction method demonstrates low registration cost by exploiting the domain-based proxy and keeps message delivery cost low by maintaining a short proxy chain. Random selection of domains works favorably for the Compacting method. In real-life situations, domains may be selected in a sorted order to reduce the migration path. If that is the case, we may need an elaborated compacting method to calculate the minimum path.

## 5 Conclusion

In this paper, we propose an efficient location tracking method for mobile agents using domain-based proxy. The domain-based proxy method can minimize registration and message delivery costs by exploiting migration patterns. The proxy is determined when a mobile agent migrates to a new domain. The first host that a mobile agent visits in a new domain serves as a proxy in the domain. In the simulation, we calculated the registration and message delivery costs by changing the parameters such as the number of migrations per domain and the number of messages. Assuming that the hosts in a domain are close to each other, we can minimize the registration cost by exploiting the proxy within the domain and minimize the message delivery cost by compacting the proxy chain. Since the domains were selected randomly in the simulation, the proposed simple compaction method is very effective in reducing the proxy chain length. In real-life situations, however, the domains may be selected in a sorted order to

reduce the migration path. For a specific pattern of migration, we may need an elaborated compacting method to minimize costs.

## Acknowledgement

We would like to thank anonymous referees for their invaluable comments on this work. Also we would like to express our deep appreciation to the committee members of NPC 2004.

## References

1. D.B.Lange, and M.Oshima, "Seven Good Reasons for Mobile Agents ", *Communication of the ACM* ,Vol.42,No.3, March 1999,pp.88-89.
2. D.B.Lange and M.Oshima, *Programming and deploying Java mobile agents with Aglets*, Addison-Wesley, 1998
3. J.Cao, X.Feng, and S.K.Das, "Mailbox-Based Scheme for Mobile Agent Communications", *IEEE Computer*, September. 2002, pp. 54-60.
4. A.Murphy and G.P.Picco,"Reliable Communication for Highly Mobile Agents ",*Agent Systems and Architectures/Mobile Agents (ASA/MA)'99* , October 1999, pp.141-150.
5. Jiannong Cao, Xinyu Feng, Jian Lu, Henry Chan, and Sajal K. Das, "Reliable Message Delivery for Mobile Agents: Push or Pull", *ICPADS 2002*, December 2002, Taiwan, ROC., pp. 314-320
6. J. Baumann and K. Rothermel, "The Shadow Approach: An Orphan Detection Protocol for Mobile Agents," , Springer-Verlag, Berlin, Germany, vol. 1477, 1998, pp. 2-13.
7. A.Pham, and A.Karmouch,"Mobile Software Agents:An Overview " , *IEEE Communications magazin* , Vol.36, No.7, July 1998,pp.26-37.
8. G..Kunito,Y.Okumura,K.Aizawa, and M.Hatori, "Tracking Agent:A New Way of Communication in a Multi-Agent Environment " ,*Proc. of IEEE 6 th Int'l Conf. on Universal Personal Comm.*, Vol.2, 1997, pp.903-907.
9. T.K.Shih,"Agent Communication Network -A Mobile Agent Computation Model for Internet Applications " , *Proc. 1999 IEEE Int'l Symp on Computers and Communications*, 1999, pp.425-431.
10. P.Francis, S.Jamin, V. Paxson, L. Zhang, D. F. Gryniewicz, and Y. Jin, "An architecture for a global Internet host distance estimation service," in *Proceedings of IEEE INFOCOM '99*, New York, NY, Mar. 1999.
11. N. Minar, K.H. Kramer and P. maes, "Cooperative Mobile Agents for Dynamic network Routing", in *Software Agents for Future Communication Systems*, Springer-Verlag, 1999. ISBN 3-540-65578-6.
12. A.Bar-Noy,I.Kessler, and M.Sidi, "Mobile Users:To Update or Not To Update?" *ACM/Baltzer J.Wireless Network* , Vol.1,No.2,July 1995,pp.175-195.
13. A. Fuggetta, G.P. Picco, and G. Vigna, "Understanding Code Mobility", *IEEE Transactions on Software Engineering*, 24(5), May 1998.
14. P.Bellavista, A.Corradi, C.Stefanelli, "Mobile Agent Middleware for Mobile Computing," *IEEE Computer* Vol.34, No.3. March 2001. pp. 73-81