A Hierarchical Dynamic Monitoring Mechanism for Mobile Agent Location

Ying-Hong Wang, Huan-Chao Keh, Tsang-Ching Hu, and Cheng-Horng Liao
Department of Computer Science & Information Engineering
Tamkang University, Tamshui, Taipei County, Taiwan
inhon@mail.tku.edu.tw, keh@cs.tku.edu.tw, 791190191@s91.tku.edu.tw, jingo@cs.tku.edu.tw

Abstract

Mobile agent location monitoring is a necessary mechanism for mobile agent system. Most of mobile agent systems themselves do not certainly provide the mobile agent location monitoring mechanism, e.g. Mole, [1] D ' Agents [2], Concordia [3] and Grasshdoper [4].

Even though IBM Aglets [5] system provides the location monitoring mechanism, it is adopted the centralized monitoring mechanism. The centralized monitoring machine is not suitable for the huge mobile agents because the location information processing does exist bottleneck issue, as well as the system lacks the scalability.

In this paper, the method proposed to use distributive mechanism for delivering MMA (Mobile Monitor Agent) to the network node (Agent Server) of mobile agent system. The MMA can be treated as a regional monitoring platform. It is responsible to collect the mobile agent's information instantly in its monitoring area. Furthermore, the users can also inquire their own mobile agent's current location and working status.

The above mentioned mechanism is to distribute MMA dynamically to solve the hierarchical monitoring mechanism scalability issue, and still keeps the advantages of hierarchical monitoring mechanism to decrease information processing bottleneck issue which the centralized monitoring mechanism encounters.

Keywords: *Agent, Mobile Agent, Hierarchical monitoring, Location monitoring, Mobile Agent location monitoring.*

1. Introduction

The mobile agent extremely provides an effective method to use in the Internet and supports work with synchronously and asynchronously. The ability of mobile agent location monitoring is usually classified for the ability of mobile agent communication, such as finding out mobile agent current location and the execution environment. Although mobile agent location monitoring is a necessary mechanism for mobile agent system, it also provides many advantages in the ability of mobile agent communication. Even though IBM Aglets system provides the location monitoring mechanism, it is adopted the centralized monitoring mechanism. The centralized monitoring machine is not suitable for the huge mobile agents because the location information processing does exist bottleneck issue, as well as the system lacks the scalability. Due to limitation and disadvantages for the centralized monitoring mechanism, we need to discuss a fast and effective mechanism for mobile agent location monitoring.

In this paper, we proposed a distributive mechanism for delivering MMA to the network node (Agent Server) of mobile agent system. The MMA can be treated as a regional monitoring platform. It is responsible for collecting the mobile agent's information instantly in its monitoring area. Furthermore, the users can also inquire their own mobile agent's current location and working status. The above mentioned mechanism is to distribute MMA dynamically to solve the hierarchical monitoring mechanism scalability issue, and still keeps the advantages of hierarchical monitoring mechanism to decrease information processing bottleneck issue which the centralized monitoring mechanism encounters.

This paper is organized as follows: Section 2 describes the related works. Section 3 presents the architecture of mobile agent system. The delivery steps of the hierarchical dynamic monitoring mechanism and its corresponding mechanism when it changes are in section 4. The last part of this paper is our conclusion and future works.

2. Related works

According to the researches for location monitoring, it is called location management for mobile agents. Currently, it focuses on mobile communication

Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05)
1550-445X/05 $20.00 © 2005 **IEEE**

research, and it manages and monitors the location of mobile hosts. Nowadays, the mobile agent location monitoring has the following several methods [6]:

- **Footprints**

Footprint means that mobile agent left the record after it was there. When mobile agent migrates to the next host, it will leave a record on the host. Its disadvantage is when it has more than mobile agent on network, the Footprints mechanism had to consider when it can delete Footprints record on agent system. If it deletes a record earlier, the user can't track the mobile agent location. On the other hand, if it keeps all agent records on agent system, it needs to take more memory to save these information, and it will spend a long time for tracking route when mobile agent visited more agent servers.

- **Broadcast**

When mobile agent migrates to next server and uses Broadcast to inform all mobile agents, all agents can base on this information to find out its current location quickly. Its disadvantage is when many mobile agents roam on agent system or visit many agent servers; it will cause to remain more location information on network. Neither user wants to track their mobile agent current location, nor does user want to communicate with other agents. It will reduce the network bandwidth and take more memory to store every agent current location data.

- **Centralized**

Setting a Centralized Location Register station on agent system, it is responsible to manage all of system current mobile agent location. This station controls all of location information as well as processes the location information. Its disadvantage is that all of mobile agent location data handle by a station and will cause data process Bottleneck issue.

- **Hierarchical**

Using the Hierarchical to manage all of agent location data, it may use more area monitoring platform to share responsibility for centralized monitoring to cause data process Bottleneck issue, but the area monitoring platform is predestined to set. Therefore it will not adapt network change. The traditional hierarchical mechanism is limited about the system the scalability.

3. System components construction

The mobile agent system has the following several components [Figure 1]:

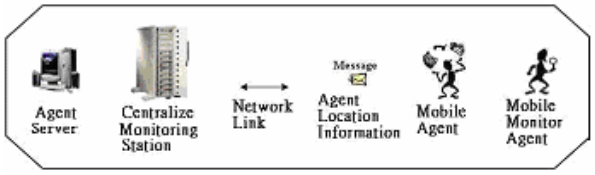

Figure 1 System components construction

- **Centralized Monitoring Station**

The function of centralized monitoring station manages the mobile agent's location information. It holds a location table, which is a key role in this system. The table shows the memory putting mobile agent's identity number and current location. This information can retrieve from each region MMA.

- **Agent Server**

Agent server refers to FIPA's[7] Agent Management Reference Model to revise. Agent server provides the Mobile Agent's working platform and the resources, which include Directory Facilitator (DF), Agent Management System (AMS) and Message Transport Service (MTS) [Figure 2].

- **Directory Facilitator (DF)**

DF is an optional component of the agent server. If it is present, it must be implemented as a DF service provides yellow page services for other agents. Agents can register their own services to the DF, or ask DF to find out other agents, which provide some services. An agent server can provide services with several DFs or only one DF.

- **Agent Management System (AMS)**

The AMS is a mandatory component of the agent server. The AMS is a system manager to be in charge of controlling whole access from the agent server. Only one AMS is in a single agent server. The AMS maintains a directory of agent identities, which contains transportation addresses (among other things) for agents registered with the agent server. The AMS offers white page services to other agents. Each agent must register with an AMS in order to get valid agent identities. Agent Management System (AMS) contains the register, deregistration and the searched.

- **Register**

When a mobile agent to enter a mobile agent server, it can have an agent identities for the agent management system, and add it to centralized monitoring station for this mobile agent in its location table.

- **Deregistration**

When a mobile agents to leave the mobile agent server, it can have deregistration to be able to delete agent

identities for the agent management system, and to delete this mobile agent in its location table

◆ **Searched**

The users can inquiry agent management system to know their mobile agent present location and current status.

■ **Message Transport Service (MTS)**

MTS is the default communication method between agents on different agent servers

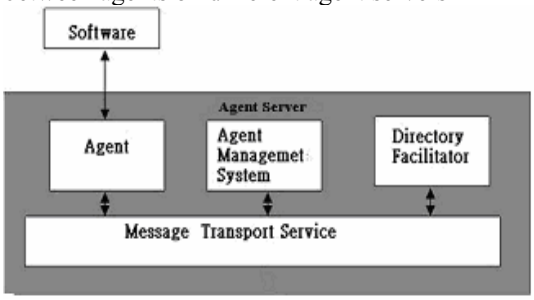

Figure 2 Agent Server components construction

■ **Mobile Agent**

When a user sends out the mobile agent, who migrates to each mobile agent server, it carries out its tasks and returns the results to the original user.

■ **Mobile Monitor Agent (MMA)**

MMA [Figure 3] may treat as a region mobile agent location monitoring station. MMA is responsible for collecting the mobile agent location in its mobile agent server.

Figure 3 MMA constructions

■ **Mobile Agent Location Table**

Mobile Agent Location Table stores the mobile agent locations under the whole area. We can treat as a location information database.

■ **Monitor Objects**

The main work of Monitor Objects is the connection which takes with each mobile agent server in agent management system. The penetration standard Message Transport Service (MTS) brings back its Location Table in material and renews own Mobile Agent Location Table. And it also penetrates MTS the relation which takes with the centralized monitoring station and all of mobile agent location material to pass on its region in renewal processing.

■ **Agent Operations**

Because MMA itself is mobile agent, it can the adjustment which makes along with the network condition to change corresponds achieve the integrated system best condition. The main function of Agent Operations manages its MMA the migration operation.

4 Hierarchical dynamic monitoring mechanism

4.1 Dynamic deployment MMA mechanism

MMA is delivered by the centralized monitoring station. It delivers on the agent servers to take a region the location monitoring station, reduces the operation and the data process bottleneck which is the centralized monitoring station. The centralized monitoring station along with the network environment can decide the delivery how many MMAs to come up to its network. Each MMA can have the different serial number to take the discrimination. The centralized monitoring station may act according to its marking to take the discrimination. A flow-chart diagram of the proposed MMA deployment algorithm is depicted in Figure 4.

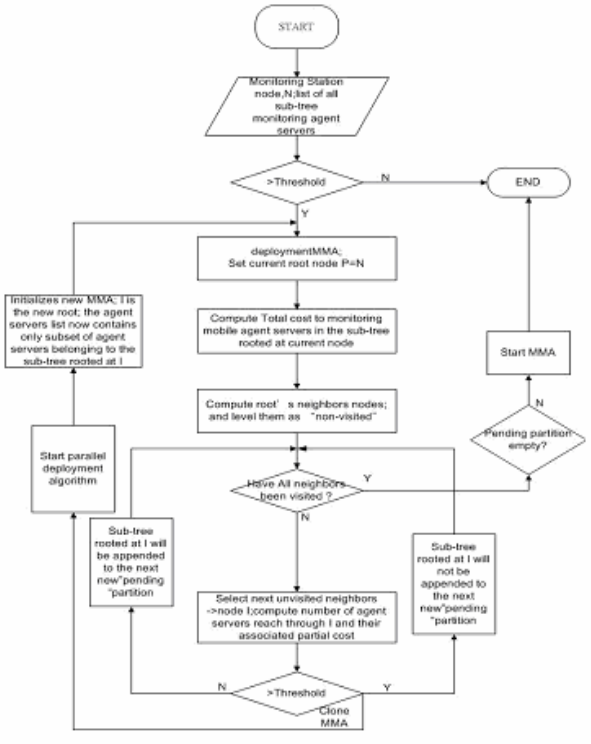

Figure 4 MMA deployment algorithms

The Procedure for deployment algorithm as listing:

- This deployment mechanism is from created MMA to start in the centralized monitoring station. The all of collection need to monitor the information of the agent servers. If there is no the threshold of the achieved deployment, this deployment step will stop.
- Starting to create MMAs to monitor all Agent Servers
- Carrying on the exploration computation regarding the under agent servers, it calculates by way of various agent servers reaction time, by took whether duplicates once more deployment MMA to the sub-tree agent servers basis. This is the judgment which hands over all of visited until the under agent servers.
- The computation core is with this MMA agent server connected by each agent servers, and takes the judgment to take this local routing table as the foundation:
  - At current MMA monitoring all of agent servers the complete cost.
  - The supposition penetrates new MMA (candidate agent server) to monitor the cost which the sub-tree agent servers needs
  - Calculates complete cost sum total, which all of neighbors agent servers extends.
- The choice neighbor agent server achieved when deployment threshold. MMA can duplicate new MMA, and penetrate this new MMA to monitoring all of agent server duty, which this agent server arrives.
- Then new MMA migrates to this neighbor agent server when MMA arrives, MMA will once start to hand over the exploration the process.
- When it does not have new MMA, it will need to duplicate, then this deployment step can end.

This development of the execution algorithm is by the parallel but is not by the order. It hands over the way to carry out this deployment movement. It starts from the root agent server, and may simultaneously decompose the network tree into several parts and deploy to carry out MMA to its part in. It achieves a highly parallel processing.

4.2 MMA Self-relocation

Because the Internet is a highly dynamic environment, this kind of mechanism must therefore have to be able adjustment correspondence to achieve the monitoring information according to the network condition to change the optimization condition. When loses the agent server (e.g. network bandwidth to obstruct), the system can base on the originally network condition to take the response. MMA can defer to the corresponding mechanism, which the agent server loses in originally the region to discover a best location to continue to carry out its monitoring task. In order to not let the overall system complication, therefore the network condition change corresponding measure only to defer to its originally region condition operation, but not produce any new MMA. Also it refers to originally the condition to pursue a best location.

A flow-chart diagram of the proposed MMA self-relocation algorithm is depicted in Figure 5.

.

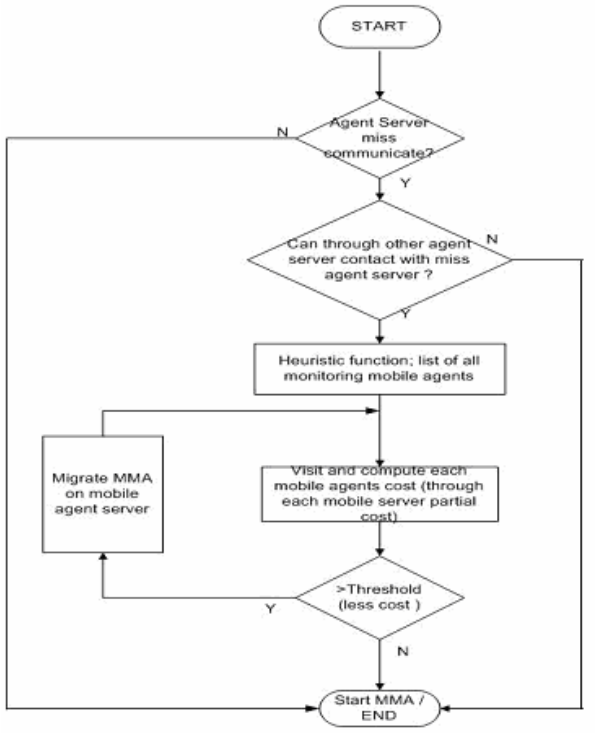

Figure 5 MMA self-relocation algorithms

The procedure for MMA self-relocation algorithm as listing:

- When there is an agent server lose, the MMA self-relocation mechanism starts.
- Examining whether penetrating in the region other agent server relations. If there is no good to cancel the mechanism, which the agent server loses corresponds. It is unable any relation information which takes.
- If it may penetrate other agent server to take the success and failure continually agent server; to withdraw the computation agent server cost, it presents monitoring cost.
- Handing over the computation by way of various agent servers cost, it obtains the best location

- MMA migrates to continue to this agent server to carry out the monitoring task.

4.3 Scenario

- MMA deployment algorithm

The algorithm is illustrated through a simple example .for network depicted in Figure 6. For the sake of simplicity, we assume that the list of monitoring mobile agent server consists of the entire network node. The basic phases the MMA location process, which depicted in Figure 6. The algorithm deploys the area monitoring station (MMA) during the network partitioning process through a "clone" and "send" process starting at centralized monitoring station. The number and location of MMAs are computed by subsequently comparing the monitoring task parameters with routing information extracted from network routers.

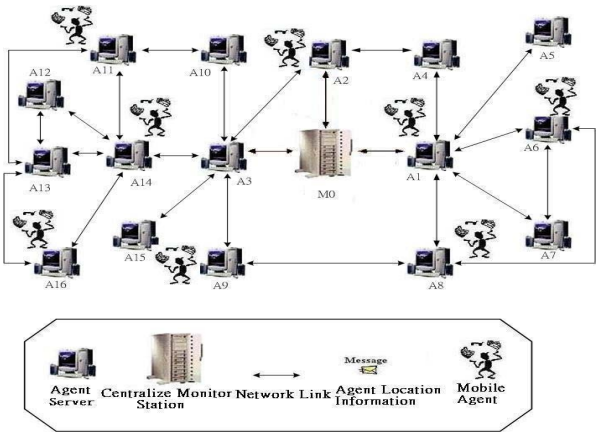

Figure 6 example network

In our example, the monitoring task includes the list of agent servers as well as the operations to be performed on them. It is delegated a first MMA (Figure 7a.). The centralized monitoring station estimates that probabilities agent serverA1 and agent serverA3 are associated to neighbors 1 and 2 respectively. They are sufficiently high to justify sending an agent to agent serverA1 and another one to agent serverA3 (Figure 7b). Agent serverA2 is relatively low (only one node is monitored through agent serverA2). Therefore, the network is partitioned into two portions. Two agents are needed; one agent is available already; one extra agent is cloned and the list of mobile agent server of each agent to be reset in order to reflect the new network partitioning. Finally, the MMAs are deployed to their new location (Figure 7c) and each of the two MMAs independently continues the 'clone and send' process.

The agent, sitting at agent server 1, is now ready to start monitoring its sub-partition since agent server4 to 8 are relatively small. In contrast, the MMA sits at location agent server3 to decide to produce a further sub-partition and to clone a new MMA – agent serverA14 is above threshold (Figure 7d). The decomposition /cloning /migrating process continue in a similar method (Figure 7e), leading to the final configuration of Figure 7f. The Figure 8 is final the MMA within the network.

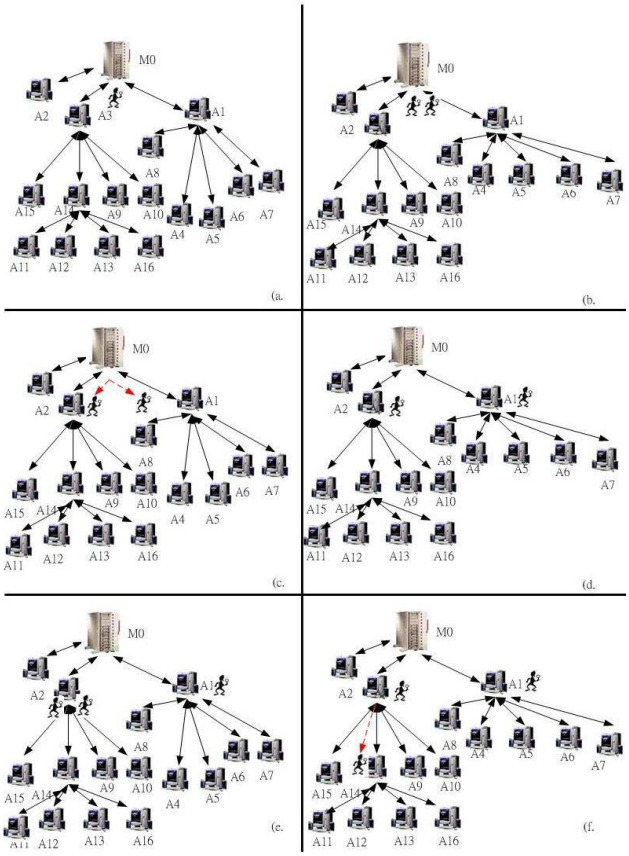

Figure 7 MMA location process

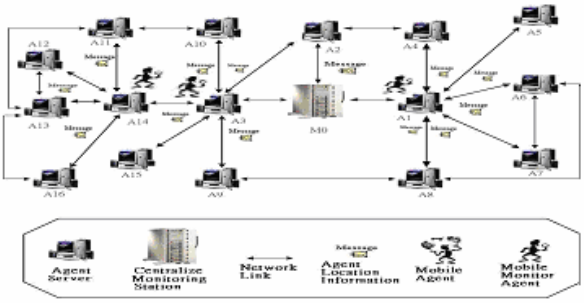

Figure 8 is finial the MMA within the network

- MMA self-relocation algorithm

A simple example, illustrating MMA self-regulation in response to a link failure, is depicted in Figure 9/

Figure 10. In this case, it follows a loss of connectivity between Agent serverA12 and A14. A new monitoring path is established between Agent serverA12 and A14. As a result, the central node for the system partition comprising agent servers (14, 11, 12, 13, and 16) becomes agent server A13. Hence, the agent, originally located in agent serverA14, will relocate to agent serverA13, to bring the system back for optimality.

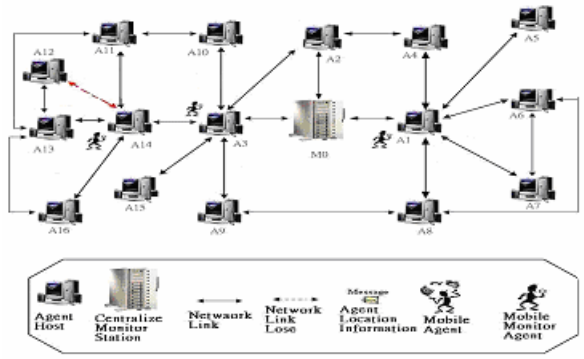

Figure 9 Agent server link failure

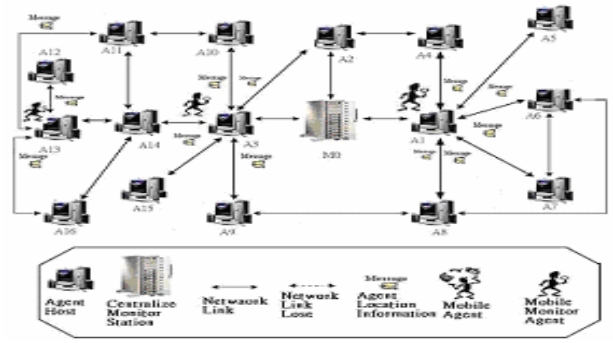

Figure 10.MMA self-regulation in response to a link failure

5. Conclusion and future works

The above mentioned mechanism is to distribute MMA dynamic to solve the hierarchical monitoring mechanism scalability issue, and still keeps the advantages of hierarchical monitoring mechanism to decrease information processing bottleneck issue which the centralized monitoring mechanism encounters.

Although the presented paper rises that the mechanism solves centralized monitoring question, it will also have some rooms to research.

The evaluation for dynamic monitoring mechanism is how every agent server becomes a region monitoring station, and how it still keeps the service, which originally provides. This will be the future research direction. By the way, we'll focus on the security issues for agent servers because agent servers provide the MMA working environment. How do we confirm the security between MMAs and the Servers? That is a direction for us in the future.

References

[1] Baumann, J., Hohl, F., Rothermel, K., Straber, M.," Mole – Concepts of a Mobile Agent System," WWW Journal, Special issue on Applications and Techniques of Web Agents, volume 1, no 3, 1998

[2] D'Agents, http://agent.cs.dartmouth.edu/

[3] Mitsubishi Electric ITA,"Concordia Developer's Guide,"October 1998

[4] IKV++ GmbH,"Grasshoper Programmer's Guide," Release 2.2, March 2001

[5] IBM Aglet, http://www.trl.ibm.co.jp/agles

[6] Ye Din-Yuan, Wang T.I,"The Mechanism of Tracking Mobile Agent" Taiwan Cheng Kung University Department of Engineering Science Master paper ,2000

[7] FIPA Agent Management Specification (Sc00023J) http://www.fipa.org/specs/fipa00023/sc00023j.html