Evaluating Tradeoffs of Mobile Agents in Network Management

Marcelo G. Rubinstein — Otto Carlos M. B. Duarte

Grupo de Teleinformática e Automação (GTA)
PEE-COPPE/EE, Universidade Federal do Rio de Janeiro
P.O. Box 68504, 21945-970, Rio de Janeiro, RJ, Brazil
Email : {rubi, otto}@gta.ufrj.br
Web : www.gta.ufrj.br/{~rubi, ~otto}

ABSTRACT. *This paper aims at investigating tradeoffs of mobile agents in network management. Mobile agents performance is compared with the SNMP one in management task simulations. Response time results show that the mobile agent is less sensitive to the latency and the bandwidth of a bottleneck link that connects the management station to the ones to be managed, but is more influenced by the task to be performed. Moreover, the mobile agent paradigm performs better when the number of managed network elements ranges between two limits: one associated with the number of messages that traverse the bottleneck link and the other related to incremental size of the mobile agent, and a significant improvement is achieved when it returns to the management station after visiting a fixed number of nodes.*

RÉSUMÉ. *Ce papier présente une étude des performances liées à lútilisation dágents mobiles dans le cadre de la gestion de réseaux. Les performances obtenues avec la technologie agents mobiles sont comparées à celles obtenues avec des technologies client/serveur, telles que celle utilisée par le protocole SNMP. Pour cela, un ensemble de tâches de gestion ont été simulées. Les résultats concernant le temps de réponse montrent que lápproche agents mobiles est moins sensible au délai et à la bande passante du lien représentant le goulot d'étranglement entre la station gestionnaire et les stations à gérer, mais quélle est plus sensible au type de tâche à effectuer. De plus, le paradigme des agents donne de meilleurs résultats lorsque le nombre d'équipements à gérer est compris entre deux limites, la première étant fonction du nombre de messages traversant le lien goulot et láutre étant fonction de la taille incrémentale de lágent mobile ; une nette amélioration est obtenue lorsque lágent revient à la station gestionnaire après avoir visité un nombre fixe de nœuds.*

KEY WORDS : *mobile agents, network management, performance evaluation*

MOTS-CLÉS : *agents mobiles, gestion de réseaux, performances*

1. Introduction

Due to the extraordinary development of the Internet, the volume of information available in the world, the data moving cost, and the little experience on computing of lots of users have been reason of concern by the network community and application developers. The great volume of information available in different sites has not been followed by the user's capacity of searching, selecting, and retrieving data. Moreover, the movement of great quantities of data on the network has a significant cost. At last, the number of users with little knowledge on computing has grown and will continue to grow very fast. One possible solution to face these problems is the use of the agent paradigm.

Agents are programs that help users to perform tasks, acting on behalf of these users. If these agents can execute their tasks on the network, they are called mobile agents. They can move to the place where data are stored and select information the user wants by using intelligence, thus, saving bandwidth, time, and money, inasmuch as the total volume of data from the various sources is not transferred. Agents also work as smart user interfaces that can detect and solve problems, learn user's preferences and habits, and automate everyday actions.

Network management is one of the mobile agent applications that can benefit from its use, due to the fact that, in a general manner, it is based on a centralized paradigm, as SNMP (Simple Network Management Protocol) [STA 98, STA 96] and CMIP (Common Management Information Protocol) [YEM 93, SLU 89] are. The rapid grow in the size of networks has brought into question the scalability of the centralized model. At the same time, the computational power of the managed elements has grown, making it possible to perform management functions in a distributed way [MEY 95]. In order to provide scalable solutions, the management system must be able to handle large numbers of elements without imposing performance problems or computer resource constraints. Furthermore, the components of a management system must be able to monitor and gather data from the managed network with limited information loss and without placing too much demand on the elements of the network [HAG 98].

Mobile agents decentralize processing and control, and, as a consequence, reduce the traffic around the management station, turn asynchronous agent-management communication (useful when there are unreliable or lossy links), distribute processing load, and increase the flexibility of the management agents' behavior.

Research activities of mobile code in network management are recent [PHA 98]. Magedanz et al. [MAG 96a, MAG 96b] have been one of the first researches to present issues for the deployment of mobile agents in telecommunications and network management. They enhance management agent functionality by developing an agent execution environment to enable the realization of "mobile management applications", implemented by mobile agent scripts. Labetoulle et al. [CHE 98] present a state-of-the-art of intelligent agents in network management, from the management by delegation paradigm [GOL 98] to the mobile agent one, while Karmouch and Pham [PHA 98] describe the current use of mobile agents in telecommunications and network management. Baldi et al. [BAL 98, FUG 98] evaluate mobile code design paradigms in network management applications by developing a quantitative model for traditional and mobile code design of management functionalities. Sahai and Morin [SAH 98a, SAH 98b] perform measurements of bandwidth utilization and response times of mobile agent and client-server applications on an Ethernet LAN, while Rubinstein and Duarte [RUB 99b, RUB 99a] simulate management tasks performed by mobile agents and SNMP ones, comparing both the approaches. Many architectures for network management with mobile and intelligent agents have been proposed in the literature [GOL 98, BAL 97, SAH 98a, BIE 98, CHE 99, KU 97, PUL]. Management by Delegation (MbD) [GOL 98, GOL 95, SCH 97] has been the first paradigm to address decentralization and automation of management tasks by dynamically delegating management functions to the agents. Some authors [BAL 97, SAH 98a, SAH 98c, BIE 98] develop management architectures that use at the same time mobile agents and SNMP ones, providing access to legacy agents. Pagurek et al. [BIE 98, SCH 98] implement a code mobility infrastructure and a suite of simple tools that interact with agents located on network components. They also propose an intelligent network model where behavior and state are part of the model, and both may be dynamically updated [WHI 99, WHI 98]. Many researches [CHE 99, KU 97, BIE 98, GüR 98, HOO 98] apply intelligent agent techniques to network management architectures. Ku et al.'s architecture [KU 97] also supports mobile agents and the MIB-II standard [MCC 91], while Ji et al. [HOO 98] describe an intelligent agent that processes information collected by SNMP agents and uses it to detect the network anomalies that typically precede a fault. Some authors [KU 97, PUL , PUL 99, BIE 98] use Java [ARN 96] as the implementation language of mobile agent infrastructures.

This work aims to investigate tradeoffs of mobile agents in network management. We analyze the performance of a management system and compare mobile agent and SNMP approaches. The performance is evaluated for latency and data rate of the bottleneck link that connects the management station to the ones to be managed, initial size of mobile agent, task to be performed, and gain due to the unload procedure of the mobile agent after visiting a fixed number of nodes.

This paper is organized as follows. In Section 2, the use of mobile agents in network management is presented. In Section 3, we evaluate simulation results concerning the applicability of mobile agents in performing management tasks. At last, concluding remarks are presented in Section 4.

2. Mobile Agents for Network Management

Currently, most network management systems use SNMP and CMIP protocols, proposed by the Internet Engineering Task Force (IETF) and the International Organization for Standardization (ISO), respectively. These protocols are based on the client-server model, where the management station acts as a client that provides a user interface to the network manager and interacts with agents, which are servers that manage remote access to their local information. These data are stored in a Management

Information Base (MIB) [1], a hierarchical base of information managed by each agent. The operations available to the management station for accessing the MIB are very low level. This fine grained client-server interaction, called micro-management does not scale well because of the generation of intense traffic and computational overload on the management station [BAL 98].

Some steps towards decentralization have already been taken by IETF and ISO organizations [KAH 97]. In event notification, SNMP agents notify the management station upon the occurrence of a few significant events. These agents use traps, i.e., messages sent without an explicit request from the management station, to decrease the intensive use of polling. ISO uses more complex agents that have higher processing capacity. In both the approaches, the agent is only responsible for the notification of an event. SNMPv2 decentralizes processing and control because it provides multiple top-level management stations, called management servers. Each such server is responsible for managing agents, but it can delegate responsibility to an intermediate manager. The intermediate manager, called proxy agent, plays the role of a manager in order to monitor and control the agents under its responsibility; and also works as an agent to provide information and to accept control from a higher-level management server. SNMPv3 makes additional minor functional changes to the SNMPv2 and incorporates a new security approach [STA 98]. IETF has also proposed RMON (Remote MONitoring) [WAL 95] that uses network monitoring devices called monitors or probes to perform proactive LAN monitoring on local or remote segments [CHE 98]. These probes provide information about links, connections among stations, traffic patterns, and status of network nodes. They also detect failures, misbehaviors, and identify complex events even when not in contact with the management station.

These proposals do not yet provide the desired level of decentralization needed to cope with large networks [BAL 98]. In order to satisfy the diverse needs of today's network, new network management systems that can analyze data, make decisions, and take proactive measures to maintain the Quality of Service (QoS) of the network must be developed [GüR 98].

Mobile agents can decentralize processing and control and improve management efficiency. Lots of advantages [NWA 96] may justify agent utilization in network management, such as:

— Reduced cost: inasmuch as management functions require the transfer on the network of a great volume of data, it may be better to send an agent to perform the task directly on management agents, where data are stored.

— Asynchronous processing: an agent can be sent through the network, performing its tasks on other nodes. While the agent is out of its home node, this node can be out of operation.

— Distributed processing: low capacity computers can be efficiently used in order to perform simple management tasks, distributing processing previously concentrated on the management station.

1. CMIP uses the term Management Information Tree (MIT). We ignore the difference for the sake of simplicity.

— Flexibility: a new behavior for the management agents can be determined by the management station, which sends a mobile agent with a new execution code, substituting the old one in real-time.

A mobile agent can move to the place where data are stored and filter information by using intelligence (e.g.: artificial intelligence). It can also interrupt execution and migrate from a machine to another, carrying code and data about its state that include information obtained from previous task executions. In this work, we compare mobile agent performance with the SNMP one by evaluating the traffic around the management station and response time in retrieving SNMP variables. The size of a mobile agent's code is not small, e.g.: 5 kbytes, compared to an SNMP packet size. Mobile agent size depends on the task to be carried out and on the language used to implement the agent, thus, a mobile agent does not perform well if the number of elements to be managed is small. But it performs better than the SNMP when there are bottleneck links that connect the management station to the ones to be managed, due to the fact that the SNMP uses lots of messages that traverse these links, while a mobile agent can move to the network area that contains the stations to be managed and carry out its tasks locally, crossing just two times (one to go to the region of interest and the other to go back to the management station) the bottleneck links. However, as the number of visited nodes grows, mobile agent size also grows, turning migration harder. A mobile agent performs better than SNMP when the number of managed elements does not exceed a superior bound. One possible solution to this problem is to visit a fixed number of nodes, return to the agent home or send all data to it (reducing mobile agent size), and start the task again on the remaining nodes.

2.1. *Simulation Model*

The applicability of mobile agents in carrying out network management tasks is assessed by comparing its performance with the SNMP one. The topology used in these simulations consists of a LAN of Managed Network Elements (MNEs) connected to the management station by a bottleneck link (Figure 1). The considered performance parameters are bandwidth consumption on the bottleneck link, evaluated by the number of bytes transmitted and received by a management station, and response time in retrieving SNMP variables.

The Network Simulator (NS) is used in this work [FAL 99]. This discrete-event simulator provides mechanisms to simulate computer networks, by node and link abstractions and several implemented protocols. This work uses Ethernet LAN and UDP functionalities. Some UDP modules have had to be changed in order to effectively receive a packet.

The simulation model assumes that links and nodes have no load, links are error-free, and processing time in application layer is not taken into account. UDP is used in all simulations and tests performed with SNMP applications provided the considered packet sizes. UDP and IP headers are also taken into account and the Maximum Transmission Unit (MTU) used in all links is 1500 bytes, thus, there is no fragmentation of

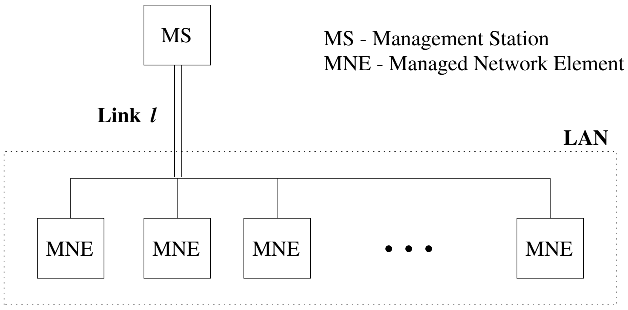

Figure 1. *Topology used in all simulations.*

SNMP messages because they are small. Every request/response of a variable is sent on a different message and the mobile agent follows a predefined itinerary [RUB 98].

3. Simulation Results

Various experiments are carried out in order to evaluate the effect on management performance of latency and data rate of the bottleneck link, initial size of mobile agent, task to be performed, and gain due to the unload procedure of the mobile agent after visiting a fixed number of nodes.

3.1. *Effect of Latency and Data Rate of the Bottleneck Link*

In this section, we evaluate the effect of latency and data rate of the bottleneck link, when retrieving an SNMP variable named *ifInErrors*, which denotes the number of received packets discarded because of errors, from interfaces of LAN stations.

Three different values for the latency of the bottleneck link are used, as in Figure 2. Data rate of the link is 2 Mbps and mobile agent size is 5 kbytes.

Figure 2. *Values for latency of the bottleneck link.*

Bottleneck Link Values	Latency (ms)	Data Rate (Mbps)
v_1	10	2
v_2	4	2
v_3	0.1	2

The number of MNEs is *n*. SNMP sends *n* requests (one after receiving the other), while the mobile agent goes to an MNE, obtains the variable, and visits all other MNEs, retrieving all variable values. After finishing, the mobile agent returns to the management station.

Figure 3 shows that the mobile agent's behavior does not change with the latency

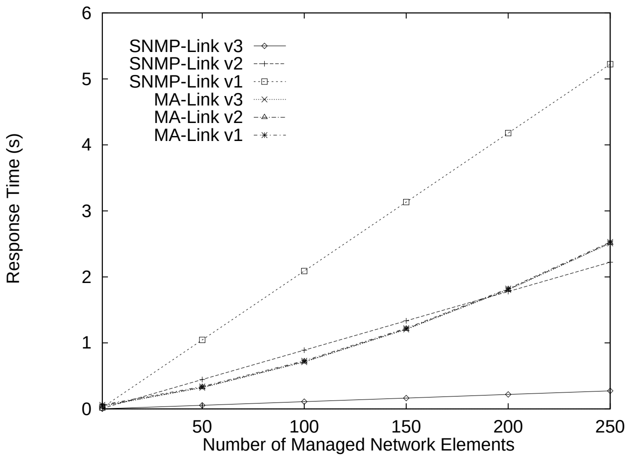

Figure 3. *Response time for the different latencies of the link.*

(all the three mobile agent curves are very close to each other). But for the SNMP, the difference of performance among the response times for the three latency values increases faster when n is large, due to the fact that SNMP packets traverse the bottleneck link oftentimes. Response time for the SNMP grows approximately linearly with n. For the mobile agent, response time increases faster when n is large because the mobile agent has a large size, which turns migration harder. For the link v_3, SNMP performs better than the mobile agent because the link latency is small, but for the link v_1, the mobile agent has a smaller response time inasmuch as the link latency is large. If the latency is medium (link v_2), the mobile agent performs better until the number of MNEs reaches a certain value (Section 3.2).

We also evaluate the effect of the data rate of the bottleneck link. Three different values for data rate of the bottleneck link are used, as in Figure 4. Latency of the link is 4 ms and mobile agent size is 5 kbytes.

Figure 4. *Values for data rate of the bottleneck link.*

Bottleneck Link Values	Latency (ms)	Data Rate (Mbps)
v_1	4	0.05
v_2	4	2
v_3	4	5

In Figure 5, for the smaller data rate (link v_1 from Figure 4), the mobile agent and the SNMP present larger response times, due to the fact that it turns difficult the transfer of the mobile agent and SNMP packets. For the other data rates, the mobile agent and the SNMP do not show a behavior different from the one previously described.

We also perform an experiment maintaining the product response time x data rate constant (Figure 6). Mobile agent size is 5 kbytes.

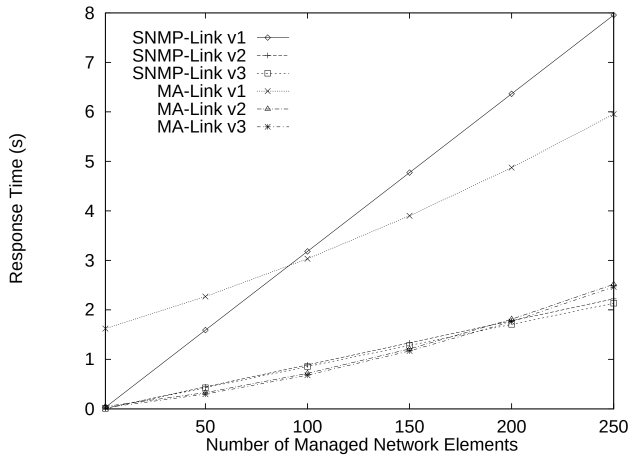

Figure 5. *Response time for the different data rates of the link.*

Figure 6. *Values for latency and data rate of the bottleneck link.*

Bottleneck Link Values	Latency (ms)	Data Rate (Mbps)
v_1	10	0.1
v_2	1	1
v_3	0.1	10

Analyzing Figure 7, we observe that the mobile agent's behavior is the same when the data rate is medium or large (links v_2 and v_3 from Figure 6), but a small data rate (link v_1) influences a lot mobile agent performance. The SNMP is mainly affected by

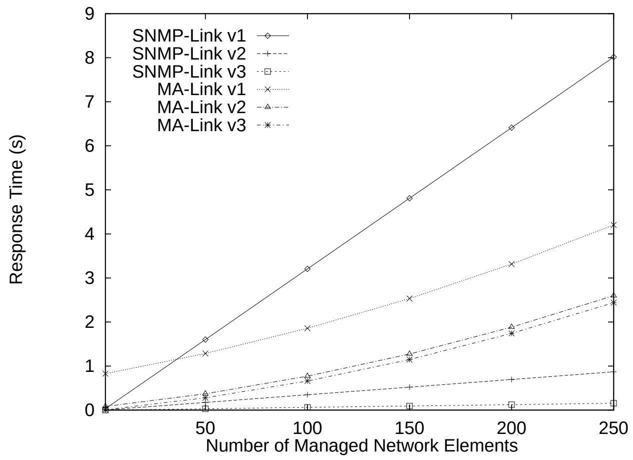

Figure 7. *Response time for the different latencies and data rates of the link.*

the small data rate and is also influenced by small and medium latencies (links v_2 and v_3).

3.2. *Effect of Initial Size of Mobile Agent*

The effect of initial size of mobile agent is also analyzed in the topology with the latency and the data rate of the bottleneck link equal to 4 ms and 2 Mbps, respectively. Five different initial sizes of mobile agent are used (1, 3, 5, 7, and 9 kbytes). The task is the same of the previous experiment.

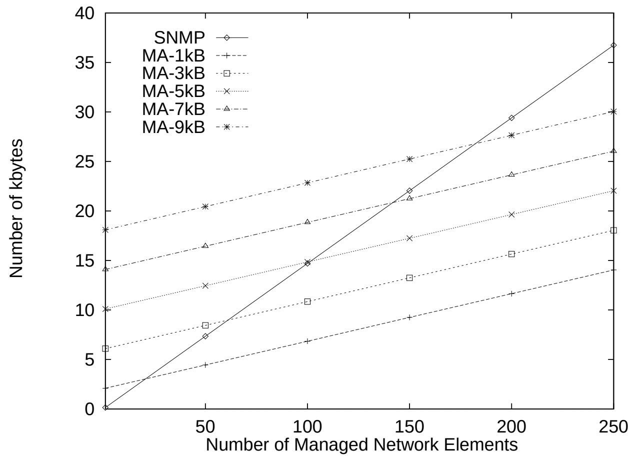

Figure 8. *Number of bytes for the SNMP and different mobile agent sizes.*

For a small number of MNEs, SNMP uses fewer bytes than the mobile agent to perform the task, but as the number of MNEs increases, the overhead (PDU GetRequest, UDP, and IP headers) associated with the number of retrievals turns mobile agent utilization more suitable (Figure 8). As mobile agent size increases, the number of bytes associated with the management station increases proportionally. For the response time, as the number of MNEs increases, the performance difference among the various mobile agent sizes increases faster (Figure 9). Analyzing the curves MA-5kB and SNMP, for a small number of MNEs, the mobile agent works better, but when the number of MNEs is greater than 200, SNMP performance is better inasmuch as mobile agent size is significant.

3.3. *Effect of Task to Be Performed*

An analysis related to mobile agent and SNMP's behavior for different tasks is performed. Three tasks are used: task t_1 is the same task used in previous sections,

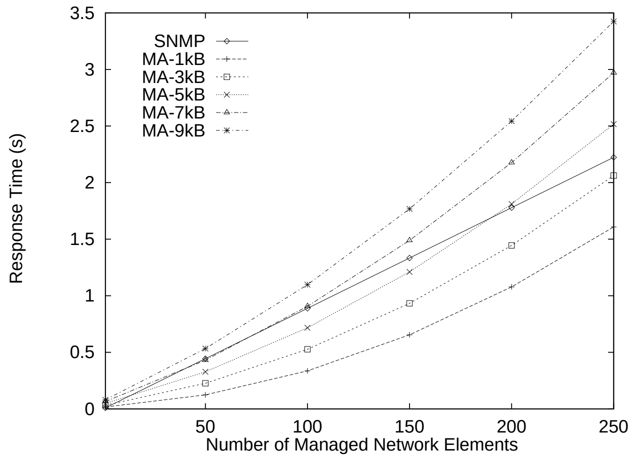

Figure 9. *Response time for the SNMP and different mobile agent sizes.*

task t_2 is related to the retrieval of the SNMP variable named *sysContact*, which is the name of contact person and how to contact them, and task t_3 is related to the SNMP variable called *sysDescr*, which is a textual description of the system. Request and response packet sizes for the tasks, including all overhead (PDU GetRequest/GetResponse, UDP, and IP headers) are presented in Figure 10. The values for the bottleneck link are the same of the previous experiment, and the mobile agent size is 5 kbytes.

Figure 10. *Request and response packet sizes for the tasks.*

Task	Request (bytes)	Response (bytes)
t_1	71	76
t_2	69	91
t_3	66	173

Figure 11 shows that the number of bytes varies with the task to be performed because the greater the number of bytes of every request, the larger the total number of bytes transmitted and received by the management station. All mobile agent curves start at 10 kbytes because agents traverse the bottleneck link two times and their initial size is 5 kbytes. In Figure 12, response time varies in a different way when using SNMP or the mobile agent. For the SNMP, the task does not have a great influence due to the fact that the number of bytes exchanged between the management station and the MNE is small, but for the mobile agent, as the number of exchanged bytes increases (e.g., from task t_1 to task t_3), the response time also increases.

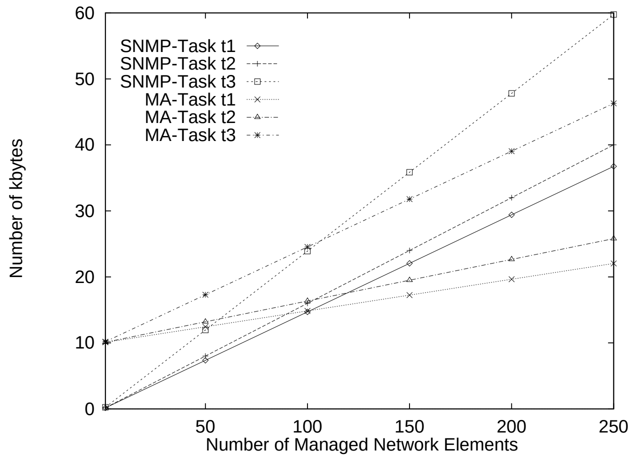

Figure 11. *Number of bytes for different tasks.*

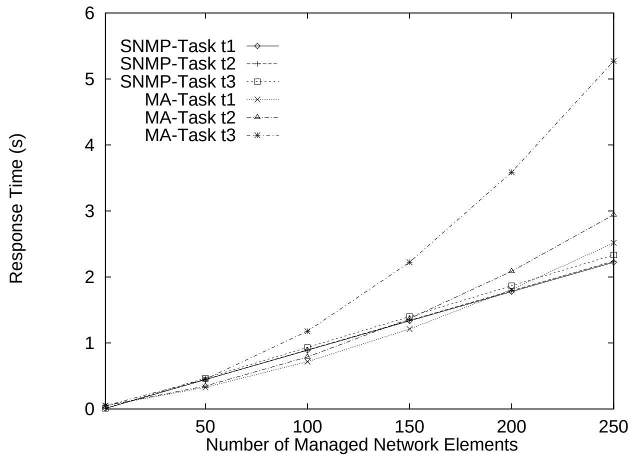

Figure 12. *Response time for different tasks.*

3.4. *Effect of Returning to the Management Station*

It has been shown in the previous sections that the mobile agent size increases with the number of visited nodes and, as a consequence, migration becomes difficult. In this section, we evaluate the performance gain of the strategy of returning to the management station when the mobile agent size becomes large. The performance analysis consider that the mobile agent visits a fixed "number of nodes per trip", i.e., the number of visited MNEs before returning to the management station to "unload". The considered tasks are the three previous one, but their purpose is to retrieve the variables from all 250 MNEs. The values for the bottleneck link value and the mobile agent size are the same of the previous experiment. The number of visited nodes per trip varies from 1 to 250.

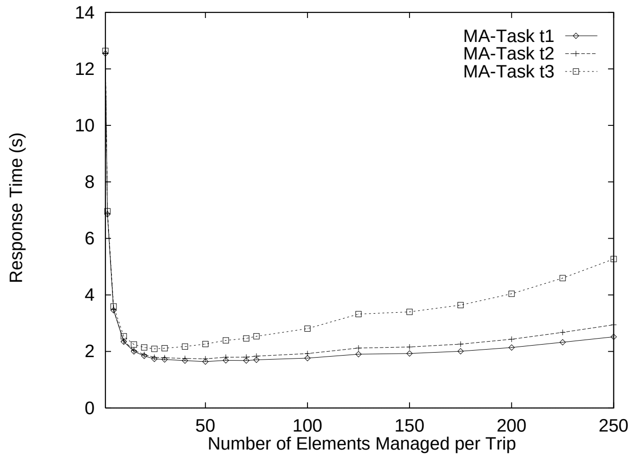

Figure 13. *Response time when returning to the management station.*

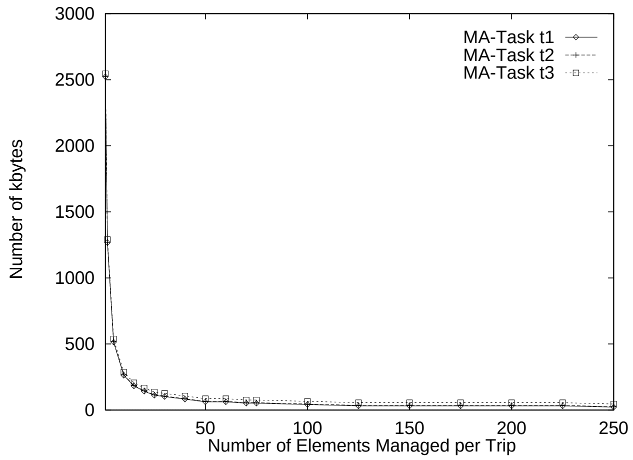

Figure 14. *Number of bytes when returning to the management station.*

Figure 13 shows that the response time decreases sharply when the number of elements managed per trip is small because the agent visits few nodes and returns to the management station. As the number of nodes visited per trip goes on raising, response time decreases up to a point when response time starts to increase due to agent migration difficulty related to its size. The "optimum point" varies with the task; for tasks t_1 and t_2, visiting 50 nodes and then returning to the management station gives the best result, while for task t_3, 25 is the number of nodes to be visited per trip. For the number of bytes, it decreases when the number of elements managed per trip increases, as in Figure 14, because the more the mobile agent returns to the management station, the larger the number of bytes transmitted and received by this station.

4. Conclusion

This work has analyzed tradeoffs of mobile agents in network management tasks. The performance of mobile agents has been compared with the SNMP one in several simulations and has been evaluated by the number of bytes transmitted and received by a management station and the response time in retrieving SNMP variables.

Response time results show that the mobile agent is less sensitive to the latency and the bandwidth of the bottleneck link that connects the management station to the remote LAN, due to the fact that SNMP packets traverse the bottleneck link oftentimes. The mobile agent is more influenced by the task to be performed, inasmuch as, for the SNMP, the number of bytes exchanged between the management station and the managed network element is small. The results also show that the mobile agent performs better than the SNMP when the number of managed network elements ranges between two limits: an inferior bound associated with the number of messages that traverse the bottleneck link and a superior bound related to the incremental size of mobile agent, which turns migration difficult. Moreover, a significant improvement of the mobile agent performance is obtained when the agent returns to the management station, after visiting a fixed number of nodes. For the number of bytes, the mobile agent performs better than the SNMP when the number of managed network elements exceeds a value related to the overhead of several retrievals of the SNMP.

We conclude that the mobile agent paradigm significantly enhances the network management performance when most of the agent movement is inside high bandwidth networks; in other words, when only a small percentage of the agent movement concerns wide area links that are bottlenecks and the management station.

Acknowledgements

This work has been supported by UFRJ, FUJB, CNPq, CAPES, COFECUB, FAPERJ and REENGE.

References

[ARN 96] ARNOLD K. et GOSLING J., *The Java Programming Language*. Addison-Wesley, 1996.

[BAL 97] BALDI M., GAI S. et PICCO G. P., Exploiting Code Mobility in Decentralized and Flexible Network Management. In *First International Workshop on Mobile Agents*, p. 13–26, Berlin, Germany, April 1997.

[BAL 98] BALDI M. et PICCO G. P., Evaluating the Tradeoffs of Mobile Code Design Paradigms in Network Management Applications. In *Proceedings of the 20th International Conference on Software Engineering (ICSE'98)*, p. 146–155, Kyoto, Japan, April 1998.

[BIE 98] BIESZCZAD A., PAGUREK B. et WHITE T., Mobile Agents for Network Management. *IEEE Communications Surveys*, vol. 1, n^o 1, p. 2–9, September 1998.

[CHE 98] CHEIKHROUHOU M., CONTI P. et LABETOULLE J., Intelligent Agents in Network Management: a State-of-the-Art. *Networking and Information Systems*, vol. 1, n^o 1, p. 9–38, 1998.

[CHE 99] CHEIKHROUHOU M., CONTI P., LABETOULLE J. et MARCUS K., Intelligent Agents for Network Management: Fault Detection Experiment. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management (IM'99)*, Boston, USA, May 1999.

[FAL 99] FALL K. et VARADHAN K., NS Notes and Documentation. Rapport technique, The VINT Project, January 1999.

[FUG 98] FUGGETTA A., PICCO G. P. et VIGNA G., Understanding Code Mobility. *IEEE Transactions on Software Engineering*, vol. 24, n^o 5, p. 342–361, May 1998.

[GOL 95] GOLDSZMIDT G. et YEMINI Y., Distributed Management by Delegation. In *The 15th International Conference on Distributed Computing Systems*, Vancouver, Canada, June 1995.

[GOL 98] GOLDSZMIDT G. et YEMINI Y., Delegated Agents for Network Management. *IEEE Communications Magazine*, vol. 36, n^o 3, p. 66–70, March 1998.

[GüR 98] GÜRER D., LAKSHMINARAYAN V. et SASTRY A., An Intelligent-Agent-Based Architecture for the Management of Heterogeneous Networks. In *Proceedings of the 9th IFIP/IEEE International Workshop on Distributed Systems: Operations & Management (DSOM'98)*, Delaware, USA, October 1998.

[HAG 98] HAGGERTY P. et SEETHARAMAN K., The Benefits of CORBA-Based Network Management. *Communications of the ACM*, vol. 41, n^o 10, p. 73–79, October 1998.

[HOO 98] HOOD C. S. et JI C., Intelligent Agentes for Proactive Fault Sectection. *IEEE Internet Computing*, vol. 2, n^o 2, p. 65–72, March 1998.

[KAH 97] KAHANI M. et BEADLE H. W. P., Decentralised Approaches for Network Management. *Computer Communications Review*, vol. 27, n^o 3, p. 36–47, July 1997.

[KU 97] KU H., LUDERER G. W. et SUBBIAH B., An Intelligent Mobile Agent Framework for Distributed Network Management. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'97)*, Phoenix, USA, November 1997.

[MAG 96a] MAGEDANZ T. et ECKARDT T., Mobile Software Agents: A new Paradigm for Telecommunications Management. In *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS'96)*, Kyoto, Japan, April 1996.

[MAG 96b] MAGEDANZ T., ROTHERMEL K. et KRAUSE S., Intelligent Agents: An Emerging Technology for Next Generation Telecommunications? In *Proceedings of the IEEE INFOCOM'96*, San Francisco, USA, March 1996.

[MCC 91] MCCLOGHRIE K. et ROSE M., Management Information Base for Network Management of TCP/IP-based Internets: MIB-II. RFC 1213, March 1991.

[MEY 95] MEYER K., ERLINGER M., BETSER J., SUNSHINE C., GOLDSZMIDT G. et YEMINI Y., Decentralizing Control and Intelligence in Network Management. In *Proceedings of the 4th International Symposium on Integrated Network Management*, Santa Barbara, USA, May 1995.

[NWA 96] NWANA H. S., Software agents: an overview. *Knowledge Engineering Review*, vol. 11, n^o 3, p. 1–40, September 1996.

[PHA 98] PHAM V. A. et KARMOUCH A., Mobile Software Agents: An Overview. *IEEE Communications Magazine*, vol. 36, n^o 7, p. 26–37, July 1998.

[PUL] PULIAFITO A., TOMARCHIO O. et VITA L., MAP: Design and Implementation of a Mobile Agent Plataform. *Journal of System Architecture (to appear)*.

[PUL 99] PULIAFITO A. et TOMARCHIO O., Advanced Network Management Functionalities through the use of Mobile Software Agents. In *Proceedings of the 3rd International Workshop on Intelligent Agents for Telecommunication Applications (IATA'99)*, Stockholm, Sweden, August 1999.

[RUB 98] RUBINSTEIN M. G. et DUARTE O. C. M. B., Service Location for Mobile Agent Systems. In *IEEE/SBT International Telecommunications Symposium (ITS'98)*, p. 623–626, São Paulo, Brazil, August 1998.

[RUB 99a] RUBINSTEIN M. G. et DUARTE O. C. M. B., Analyzing Mobile Agent Scalability in Network Management. Rapport technique, Grupo de Teleinformática e Automação, COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil, May 1999.

[RUB 99b] RUBINSTEIN M. G. et DUARTE O. C. M. B., Evaluating the Performance of Mobile Agents in Network Management. Rapport technique, Grupo de Teleinformática e Automação, COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil, March 1999.

[SAH 98a] SAHAI A. et MORIN C., Enabling a Mobile Network Manager (MNM) through mobile agents. In *Second International Workshop on Mobile Agents*, Stuttgart, Alemanha, September 1998.

[SAH 98b] SAHAI A. et MORIN C., Mobile Agents Enhanced Thin Client Approach to Network Management. In *Proceedings of the IEEE Singapore International Conference on Networks (SICON'98)*, Kent Ridge, Singapore, June 1998.

[SAH 98c] SAHAI A. et MORIN C., Towards Distributed and Dynamic Network Management. In *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS'98)*, New Orleans, USA, February 1998.

[SCH 97] SCHÖNWÄLDER J., Network management by delegation - From research prototypes towards standards. *Computer Networks*, vol. 29, n^o 15, p. 1843–1852, November 1997.

[SCH 98] SCHRAMM C., BIESZCZAD A. et PAGUREK B., Application-Oriented Network Modeling with Mobile Agents. In *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS'98)*, New Orleans, USA, February 1998.

[SLU 89] SLUMAN C., A Tutorial on OSI Management. *Computer Networks*, vol. 17, n^o 4&5, p. 270–278, October 1989.

[STA 96] Stallings W., *SNMP, SNMPv2, and RMON: Pratical Network Management*. Addison-Wesley, 1996.

[STA 98] Stallings W., SNMP and SNMPv2: The Infrastructure for Network Management. *IEEE Communications Magazine*, vol. 36, nº 3, p. 37–43, March 1998.

[WAL 95] Waldbusser S., Remote Network Monitoring Management Information Base. RFC 1757, February 1995.

[WHI 98] White T., Pagurek B., Bieszczad A., Sugar G. et Tran X., Intelligent Network Modeling Using Mobile Agents. In *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM'98)*, p. 1082–1087, Sydney, Australia, November 1998.

[WHI 99] White T., Pagurek B. et Bieszczad A., Network Modeling for Management Applications Using Intelligent Mobile Agents. *Journal of Network and Systems Management (to appear)*, September 1999.

[YEM 93] Yemini Y., The OSI Network Management Model. *IEEE Communications Magazine*, vol. 31, nº 5, p. 20–29, May 1993.

photo

***Marcelo G. Rubinstein** was born in Rio de Janeiro, Brazil, on August 29, 1971. He received the B. Sc. degree in Electronic Engineering and the M. Sc. degree in Electrical Engineering from Universidade Federal do Rio de Janeiro, Brazil, in 1994 and 1996, respectively. He is a D. Sc. student of Electrical Engineering at the Grupo de Teleinformática e Automação (GTA) from Universidade Federal do Rio de Janeiro. His major interests are mobile agents, network management, and multimedia applications.*

photo

***Otto Carlos M. B. Duarte** was born in Rio de Janeiro, Brazil, on October 23, 1953. He received the Electronic Engineer degree and the M. Sc. degree in Electrical Engineering from Universidade Federal do Rio de Janeiro, Brazil, in 1976 and 1981, respectively, and the Dr. Ing. degree from ENST/Paris, France, in 1985. Since 1978 he is Associate Professor at Universidade Federal do Rio de Janeiro. Presently, he is heading the computer network group (Grupo de Teleinformática e Automação - GTA). His major research interests are high speed communications, multimedia protocols, QoS garantees, mobile agents, and active networks.*