Mobile Agents for Network Management: When and When Not!

Huy Hoang To
School of Computer Science and Software Engineering
Monash University
Melbourne, Australia
IDD + 61 3 9903 2787
hoang.tohuy@dts.com.vn

Shonali Krishnaswamy
School of Computer Science and Software Engineering
Monash University
Melbourne, Australia
IDD + 61 3 9903 1967
Shonali.Krishnaswamy@infotech.monash.edu.au

Bala Srinivasan
School of Computer Science and Software Engineering
Monash University
Melbourne, Australia
IDD + 61 3 9903 1333
Bala.Srinivasan@infotech.monash.edu.au

ABSTRACT
In order to fully realise the potential of mobile agent technology to address the needs of the network management domain, it is imperative to establish the conditions where mobile agent-based (MA) NMS or SNMP-based NMS should be employed to achieve the optimal performance in term of overhead traffic generated. This paper presents mathematical models to approximate the overhead traffic created by the MA-based NMS and SNMP-based NMS on the production network, based on the complexity of the management task involved. Through our analysis and experimentation, we establish that there is range (of the number of nodes involved) wherein it is advantageous to use mobile agents. Furthermore, we demonstrate through our analytical model that this range can be estimated a priori, thereby facilitating a decision on when to deploy mobile agents and when not to.

Keywords
Mobile Agents, Network Management, Performance Optimisation

1. INTRODUCTION
Telecommunication networks using Internet Protocol (IP) have become a critical factor for the success of the organizations. This increasing importance of IP networks has raised the need for more efficient tools and techniques to manage these networks. Since current IP network management systems are typically built based on centralized client-server architectures, with the Simple Network Management Protocol (SNMP) as its core management protocol, serious drawbacks have been experienced when the size and complexity of the network increases [8]. Thus, the need for decentralized and distributed network management architectures is more important and necessary than ever before [8, 3].

Deemed as a promising approach for distributed applications, the mobile agent paradigm has been the focus of recent research to implement decentralized network management systems. Mobile agents, being defined as an active and autonomous software entities that move from host to host to perform a pre-defined task, help delegate several management operations to be perform at managed devices. With the unique ability to have autonomy in decision-making and mobility during its life span, mobile agents promise to provide scalability and flexibility for distributed network management systems. The suitability of the mobile agent paradigm for network management has been investigated extensively in research prototypes [12, 5, 6, 7]. As a result, the computational burden at the management station is reduced [8, 3]. Advantages of using mobile agents to decentralise network management system include:

- Flexibility: Since network management functions are to be built in different mobile agents, adding a new management function is easy. There is no need to upgrade or have changes at managed nodes. All that is required is to have the new mobile agent with the new management function loaded into the management station and the whole system is ready to provide new management functionality [8, 3].
- Potential bandwidth saving: Mobile management agents are usually designed to be able to perform a predefined management task at the managed devices with minimum interaction with the management station. The traffic between managed nodes and management station is possibly reduced. As a result, network bandwidth can be saved for production purposes [8, 3].
- Distributed process load: As mentioned earlier, with the help of mobile agents, some management functions can now be carried out at managed devices, reducing the workload on the management station [8, 3].

However, despite the benefits that mobile agents promise to bring to network management domain, it is obvious that there will be a certain amount of overhead traffic incurred [2, 9] when mobile agents are employed for network management purposes. This is because the systems have to support mobility and navigation models required by mobile agents to traverse the network performing network management tasks. In previous studies in performance evaluation and comparison of MA-based Network Management Systems with traditional SNMP systems [2, 4, 9, 10, 11], it was established that MA-based network management system outperforms the traditional SNMP-based counterpart under certain conditions. These conditions called for the size of the mobile agent to be kept small enough through its lifespan and the size of the management domain to be kept lower than a certain boundary. In addition, it is suggested in [9] that, for MA-based NMS to produce better performance, the size of the management domain must also be above a certain lower bound. Contrary to this, in [1] concludes with results that show that as the number of nodes increases, so does the performance gained by using mobile agents.

The conditions established in previous work have focused on mobile agent size, size of management domain and other parameters separately. Those parameters, when taken together, constitute the complexity of the management task. Since these parameters are strongly related to each other, it is observed that the relationship between these parameters within a management task ultimately determines the conditions, under which MA-based NMS or SNMP-based NMS should be employed in order to achieve the best performance. This paper investigates the conditions when mobile agents (MA) based network management systems perform better than SNMP-based network management. The condition considered in our approach is the complexity of the management task, which is determined based on the relationship between the number of managed devices involved in the management task and other constraints such as the size of the SNMP message, the size of the mobile agent, the number of SNMP messages, the number of mobile agents and the incremental size of the mobile agent. We also establish that there is range (of the number of nodes involved) wherein it is advantageous to use mobile agents. Furthermore, we demonstrate through our analytical model that this range can be estimated a priori, thereby facilitating a decision on when to deploy mobile agents and when not to.

The paper is organised as follows. Section 2 presents our theoretical model for evaluation of when it is appropriate to use mobile agents for network management and when SNMP-based approaches offer maximum benefits in terms of performance. Section 3 presents experimental evaluation of our model. Section 4 concludes the paper. The proceedings are the records of the conference.

2. OUR MODEL FOR PERFORMANCE EVALUATION OF NMS

In this section, we derive a mathematical model to set up the conditions to determine what NMS, MA-based or SNMP-based, should be employed, in order to minimise the overhead traffic generated by NMS on the production network. The model is expressed as the mathematical relationship between the number of managed devices involved in the management task and other parameters constituting the complexity of the management task such as the size of the SNMP message, the number of SNMP messages involved, the size of the mobile agent, the incremental size of the mobile agent and the number of mobile agents involved in the management task.

2.1 Overhead Traffic Model for SNMP-based NMS

SNMP-based NMS carries out a management task by having the manager send a request to the managed devices. The manager then waits for the response from the managed devices, indicating the results of the instruction. To derive a mathematical formula for estimating the overhead traffic generated by SNMP-based NMS, we consider the following cases with increasing levels of interactions.

Case 1: The management task involves only one managed node and requires only one instruction from the network management station. In this simplest case, the network management station sends one SNMP request to the managed device, instructing the stationary agent located at the managed device to perform a specific task, which may be setting a configuration variable of the managed device or getting the status of a configuration variable of the managed device. When the required task is completed, the managed device communicates the results back to the network management station through a SNMP response. As a result, the total overhead traffic generated by NMS to complete this type of management task is equal to the size of one SNMP request message and the size of one SNMP response message. In most implementations of SNMP-based NMS today, SNMP messages are implemented with approximately the same size, which is 2usually smaller than 1500 bytes [7] in order to avoid fragmentation. Hence, if the size of a SNMP message is Ssnmp, the overhead traffic of generated by SNMP-based NMS in order to complete this type management task is: OTsnmp = 2*Ssnmp.

Case 2: The management task involves one managed device and requires multiple instructions from the network management station. This scenario extends the first case by having the network management station send multiple SNMP requests to the managed node, instructing the stationary agent located at the managed device to perform a specific task. As before, the task may be setting multiple configuration variables of the managed device or getting the status of multiple configuration variables of the managed device. Since the SNMP protocol was designed with the idea of trying to relieve computing overhead from managed devices as much as possible, the stationary agents located at managed devices are often capable of doing only one instruction at one time. In other words, even though the stationary agent can receive multiple instructions through SNMP messages, the stationary agent would only be able to carry out the instructions sequentially, in the order that the instructions are received. As a result, for each and every instruction the stationary agent receives, it will reply with one SNMP response after the instruction has been carried out, in order to communicate the result back to the management station. Hence, if the number of SNMP requests that the network management station sends out is Nsnmp, the number of SNMP responses that the managed device produces would also be Nsnmp. Therefore, if the size of a SNMP message is Ssnmp, the total traffic overhead created by SNMP requests would be Ssnmp*Nsnmp and the total traffic overhead created by SNMP responses created would also be Ssnmp*Nsnmp. As a result, the total overhead traffic generated by SNMP-based NMS in this scenario would be: OTsnmp = 2*Snmp*Nsnmp.

Case 3: The management task involves multiple managed devices and requires multiple instructions from the network management station. This case is a generalisation of the previous two simple scenarios. In this scenario, the network manager tries to perform a task with the scope involving a number of managed devices. The task may be getting the status of the entire network management domain or setting different configuration variables at different devices to pre-calculated values so that all managed devices within the management domain can be synchronised. For each managed device in this scenario, the network management stations send out multiple SNMP requests to instruct the stationary agent at the managed device to perform a subtask of the total coordinated management task. Like the previous cases, for each SNMP request, there will be a SNMP response. Hence, if the number of SNMP requests at the managed device number t is Ntsnmp and the size of a SNMP message is Ssnmp, then the traffic overhead generated by the network management station and the managed device number t is 2* Ntsnmp * Ssnmp as analysed in the previous section. Therefore, let the number of managed device involved in the management task be x, the total overhead traffic created by SNMP-based NMS in this scenario would be:

$$OTsnmp = \sum_{i=1}^{x} 2 * S_{snmp} * N^{i}_{snmp}$$

Thus, the parameters that impact on the overhead traffic generated by SNMP-based NMS after performing management task T are:

- Number of the managed elements involved in a management task T. Let this be expressed as x.
- Size of a SNMP message. Let this be expressed as Ssnmp.
- Number of SNMP messages per node involved in a management task. Let this be expressed as Nsnmp.

The overhead traffic generated by SNMP-based NMS when performing management task T is:

$$OTsnmp = \sum_{i=1}^{x} 2 * S_{snmp} * N^{i}_{snmp}$$

Further analysis on this mathematical formula concludes that OTsnmp is a linear function of x, with the assumption that Ntsnmp is the same for all t, and can be expressed as $OT_{snmp} = a * x$, where a is calculated based on Ssnmp and Nsnmp as follows: $a = 2 * S_{snmp} * N_{snmp}$.

2.2 Overhead Traffic Model for MA-based NMS

The previous section has analysed the performance of the SNMP-based NMS based on the management task complexity, and presented a mathematical formula for estimating the overhead traffic created by SNMP-based NMS. In order to compare the performance of SNMP-based and MA-based NMS, this section analyses the operation of MA-based NMS and derives a mathematical formula for computing the overhead traffic generated by MA-based NMS.

As discussed in section 3.1, the MA-based NMS carries out a management task by having the network management station send out different mobile agents to the managed devices. These mobile agents then execute the task within the environment provided by the managed devices, accessing the management information in order to complete the task required. After the task is completed, the mobile agent either returns itself to the network management station, bringing along the results, or communicates the results back to management station via a messaging scheme and terminates itself at the managed device. It is obvious that the former scheme would result in far more overhead traffic in comparison with the latter scheme. Hence, in this paper, we only consider the situation where the mobile agent sends the results back to the management station and terminates itself at the managed device, after the management task is completed.

When the management task requires a mobile agent to visit multiple managed devices, the mobile agent has two choices after completing its task at the managed device and before moving on to the next managed in its itinerary. The first choice would be the mobile agent communicates the result back to the network management station to prevent itself from increasing in size when moving to the next managed device. In this scheme, the size of the mobile agent will be kept the same through out its lifespan. The second choice would be the mobile agent bringing along the result to the next managed device as it moves from device to device until the entire management task is completed. At that point, as discussed in the previous paragraph, the mobile agent sends back the result to the network management station and terminates itself at the final managed device in its itinerary. With this scheme, the size of the mobile agent will keep increasing each time the mobile agent departs from one managed device to the next. Hence, this scheme creates more overhead traffic than the scheme of sending the result back to the management station, at the end of its operation at every managed device. However, it is perceived that, one advantage of using mobile agent is its ability to work independently from the network management station. Thus, if there is a management task, which requires the mobile agent to intelligently calculate what needs to be done when it arrives at the managed device based on results collected from the previously visited managed devices, sending back results after visiting each managed device would not be desirable. Therefore, this paper chooses to model the MA-based NMS based on the assumption that, the MA-based NMS is designed to have the mobile agent bring along the result it collected from the previously visited managed devices to the next managed device in the itinerary. The mobile agent sends back the result to the management station before self-termination, only when there is no more managed device to visit in the itinerary

Like previous section, in order to derive a mathematical formula for estimating the overhead traffic created by MA-based NMS, we consider three cases with increasing levels of interactions.

Case 1: The management task involves only one managed device. In this scenario, the network management station sends one mobile agent to the managed device to perform a specific task. This may be setting a configuration variable of the managed device or getting the status of a configuration variable of the managed device. When the mobile agent arrives at the managed device, it is provided with access to the management information stored at the managed device. The mobile agent then executes within the environment provided to complete the task required. After the task is completed, the result is sent back to the network management station via a messaging scheme before the mobile agent terminates itself. Thus, the total overhead traffic generated by MA-based NMS to complete this type of management task is the initial size of the mobile agent and the size of message sent back to the network management station containing the result. Hence, if we denote the initial size of the mobile agent as Sma and the size of the message containing the result is Ima, the overhead traffic generated by MA-based NMS in this case would be: OTma = Sma + Ima.

Case 2: The management task involves multiple managed devices but requires only one mobile agent. In this scenario, the network management station sends out only one mobile agent with a predefined itinerary, which lists out all managed devices the mobile agent need to visit to complete the management task. For each managed device in the itinerary, the mobile agent arrives and performs the required task for that managed device. After the required task for that managed device is completed, the mobile agent moves to next managed device listed in its itinerary, brings along the result it just collected. Since the result of a management task is either a set of values or a SUCCESS/FAIL indicator of the operation just carried out, these values would typically range below 50 bytes in size [10] in general. Hence, we make the assumption that, the results that mobile agent collects at different managed devices are of the same size and let it be Ima. Since the mobile agent carries the result along when it moves to the next device, the size of the mobile agent increases (t-1)*Ima bytes when it finishes its operation at the managed device number (t-1) and about to visit the managed device number t. Thus, if we denote the initial size of the mobile agent be Sma, the overhead traffic that MA-based NMS generates when it's about to visit managed device number t would be (Sma + (t-1)*Ima). Therefore, if we denote x as the number of managed devices that the mobile agent has to visit in the management task, the total traffic overhead created by MA-based NMS would be:

$$OT_{ma} = S_{ma} + (S_{ma} + I_{ma}) + ... + (S_{ma} + x * I_{ma})$$

$$OT_{ma} = \sum_{j=0}^{x} (S_{ma} + j * I_{ma})$$

The management task involves multiple managed devices and requires multiple mobile agents from the network management station. In this scenario, the management task to be carried out may be getting the status of the entire network management domain or setting different configuration variables at different devices to pre-calculated values so that all managed devices within management domain can be synchronised. However, instead of sending out only one mobile agent to do the task, the network manager divides the management task and sends out multiple mobile agents to speed up the process. For this paper, it is assumed that the management task is divided such that there is no communication between mobile agents and all mobile agents would work independently of each other. The only communication that the mobile agents have is with the network management station. The management task is presumed completed when all mobile agents communicate the results back to the network management station. It is also assumed that the management task is divided such that each mobile agent has to visit an equal number of managed devices. In other words, if the number of managed devices involved in this management task is x and the number of mobile agents sent out is Nma, each mobile agent has to visit (x/Nma) managed devices. Therefore, if the initial size of the mobile agent is Sma, the size of the result collected is Ima, the overhead traffic that MA-based NMS generated for each mobile agent, based on case 2, is: $\sum_{j=0}^{x/Nma-1} (Sma + j * Ima) + Ima$

Hence, the total traffic overhead created by MA-based NMS would be:

$$OT_{ma} = \sum_{i=1}^{N_{ma}} \sum_{j=0}^{x/N_{ma}-1} (S_{ma} + j * I_{ma}) + I_{ma}$$

Thus, the parameters that impact on the overhead traffic generated by MA-based NMS after performing management task T are:

- Number of the managed elements involved in a management task T. Let this be expressed as x.
- Initial size of a mobile agent message. Let this be expressed as Sma.
- Number of mobile agents involved in the management task. Let this be expressed as Nma.
- Incremental size of the mobile agent after visiting a particular managed device. Let this be expressed as Ima
- The overhead traffic generated by MA-based NMS when performing management task T is:

$$OT_{ma} = \sum_{i=1}^{N_{ma}} \sum_{j=0}^{x/N_{ma}-1} (S_{ma} + j * I_{ma}) + I_{ma}$$

Further analysis on this mathematical formula concludes that OTma is a quadratic function of x and can be expressed as:

OTma = c*x2 + d*x + e, where c, d, and e are calculated based on Sma, Ima and Nma as follows: $c = I_{ma} / 2$, $d = S_{ma} + I_{ma} * N_{ma} / 2$, $e = I_{ma} / N_{ma}$

2.3 Performance Comparison

The previous two sections have presented the formulae for estimating traffic overhead created by MA-based and SNMP-based NMS. In this section, we establish the conditions thereby improve performance by selecting the appropriate management system to be deployed. The overhead traffic generated by SNMP-based NMS when performing management task T is:

$$OTsnmp = \sum_{i=1}^{x} 2 * Ssnmp * Nsnmp ,$$

where:

- x is the number of the managed elements involved in a management task T
- Ssnmp is the size of a SNMP message
- Nsnmp is the number of SNMP messages per node involved in the management task

OTsnmp can be further analysed as a linear function of x and can be expressed as: $OT_{snmp} = a * x$, where a is calculated based on Ssnmp and Nsnmp as follows: $a = 2 * S_{snmp} * N_{snmp}$.

The overhead traffic generated by SNMP-based NMS when performing management task T is:

$$OT_{ma} = \sum_{i=1}^{N_{ma}} \sum_{j=0}^{x/N_{ma}-1} (S_{ma} + j * I_{ma}) + I_{ma}$$

where:

- Sma is the initial size of the MA.
- Nma is the number of concurrent MAs involved in the management task
- Ima is the incremental size of MA after visiting each managed device.
- x is the number of the managed devices involved in the management task

OTma can be further analysed as a quadratic function of x and can be expressed as:

OTma = c*x2 + d*x + e, where c, d, and e are calculated based on Sma, Ima and Nma as follows: $c = I_{ma} / 2$, $d = S_{ma} + I_{ma} * N_{ma} / 2$, $e = I_{ma} / N_{ma}$

Thus, for MA-based system to perform better than the traditional SNMP-based systems, for a given management task, MA-based system needs to produce less overhead traffic. Hence, the condition required is: OT ma < OTsnmp

=> c*x2 + d*x + e < a*x + b

=> c*x2 + (d-a)*x + (e-b) < 0

=> x1 < x < x2

where x1 and x2 are calculated based on a, b, c, d, and e which in turns, are calculated based on Ssnmp, Nsnmp, Sma, Ima and Nma

Similarly, for SNMP-based NMS to perform better than the MA-based NMS, for a given management task, SNMP-based system needs to produce less overhead traffic. Hence, the condition required is: OTsnmp < OTma

=> c*x2 + d*x + e > a*x + b

=> c*x2 + (d-a)*x + (e-b) > 0

=> x < x1 OR x2 < x

where x1 and x2 are calculated based on a, b, c, d, and e which in turns, are calculated based on Ssnmp, Nsnmp, Sma, Ima and Nma.

3. EXPERIMENTAL EVALUATION

In order to verify the validity of the mathematical formulae derived through the analytical model, two main scenarios were evaluated. For each scenario, we determine the size of SNMP messages, the initial size of MA, the number of SNMP messages, the number of MAs sent out and the incremental size of MA. We then employ the mathematical formulae presented in section 2 to calculate the range for the number of managed device involved in the management task. The experiment establishes that, if the number of managed devices is between that calculated range, MA-based NMS produces less overhead traffic. Otherwise, SNMP-based NMS generates less overhead traffic.

We used EtherDetect Packet Sniffer (http://www.etherdetect.com/)] to capture and analyse the network traffic, in order to estimate the actual size of the traffic overhead generated. EtherDetect is an easy-to-use packet sniffer and network protocol analyser, which provides a connection-oriented view for analysing packets effectively. Using EtherDetect, we were able to capture SNMP traffic and mobile agent traffic and then save them to hard disk. The size of the file containing the captured data is the actual size of the traffic overhead generated by SNMP-based NMS or MA-based NMS.

The network used in the experiments composed of four HP workstations running Windows 2000 professional operating systems, interconnected through a 10 MB Ethernet LAN. One workstation is dedicated to be the network management station. The other three act as managed devices. In order to simulate a large-scaled network, we used parallel threads to query the managed devices.

The SNMP-based NMS employed in the experiment consisted of a network manager software, namely SNMPUTIL, running at the network management station. SNMPUTIL is a command line utility that allows the querying of the MIB information from the network management station. With the SNMPUTIL program, we were able to access the SNMP OID and get information from the queries that we posted. For all managed devices, the standard SNMP agent that comes as part of Windows 2000 was enabled. The windows SNMP agent is able to answer to SNMP queries and to send traps to the network manager (SNMPUTIL) located at the network management station.

The MA-based NMS employed in this experiment consists of a network manager together with a set of mobile agents, located at the network management station. For the purpose of our experiment, this MA-based NMS has been developed with minimal functions in order to minimise the initial size of the mobile agent. The Aglets Software Development Kit (ASDK), which includes the Aglets API, extensive documentation, numerous examples, source code, and Tahiti, an aglet server/viewer, was chosen to be the development kit for the MA-based NMS. Both the network manager software and mobile agents employed in this experiment are Aglets based. For each managed device, in addition to enabling the standard SNMP agent within the Windows 2000 operating system, there is an Aglets server running to provide the execution environment for mobile agents. Furthermore, each managed device is also equipped with a SNMPUTIL located at a specified local directory. The purpose of having SNMPUTIL at each and every managed device is to enable mobile agents to access the SNMP MIB without having mobile agents carry the code for accessing the SNMP MIB while travelling. The mobile agent invokes SNMPUTIL locally to enable itself with the ability to speak SNMP, through the use of the Java RunTime class, once it arrives at the managed device. This, again, minimises the size of the mobile agent while it is moving between hosts.

3.1 Scenario 1

In this scenario, the management task is simple, which is getting the CPU usage of a number of managed devices at a particular time. The mobile agent employed in the MA-based NMS is optimised in size. To carry out this task by SNMP-based NMS, we set the size of the SNMP message to its typical size of 1500 bytes. The reason that 1500 bytes is the typical SNMP size is because 1500 bytes is the Maximum Transmission Unit (MTU) and SNMP message is often not bigger than the MTU of the path that the message travel to avoid fragmentation. Since the task is to get only the CPU usage of managed devices at a particular time, the number of SNMP request messages needed per managed device is one. With Ssnmp = 1500 and Nsnmp = 1, based on the formula from section 2.1, the overhead traffic generated by SNMP-based NMS is:

$$OT_{snmp} = 2 * S_{snmp} * N_{snmp} * x$$

$$=> OT_{snmp} = 2 * 1500 * 1 * x$$

$$=> OT_{snmp} = 3000 * x$$

As mentioned above, for MA-based NMS, the mobile agent is optimised in size in this scenario. Thus, we set the initial size of the mobile agent to 2000 bytes, which is the smallest size of a mobile agent that we were able to generate. The number of mobile agents used in this scenario is one and the incremental size of the MA is set to 50 bytes. Therefore, with Sma = 2000, Nma = 1 and Ima = 50, based on the formula from section 3, the overhead traffic generated by the MA-based NMS is:

$$OT_{ma} = \sum_{j=0}^{x-1} (S_{ma} + j * I_{ma}) + I_{ma}$$

$$OT_{ma} = \sum_{j=0}^{x-1} (2000 + j * 50) + 50$$

$$OT_{ma} = 25 * x^2 + 1975 * x + 50$$

To set up the condition when MA-based NMS to perform better than SNMP-based NMS, we compare the overhead traffic generated by both systems: $OT_{snmp} > OT_{ma}$

$$=> 3000 * x > 25 * x^2 + 1975 * x + 50$$

=> 0.48 < x < 40.9

Hence, to perform the management task presented in this scenario, the conditions established by our formulae states that: if the number of managed devices is between 1 and 40 inclusive, then MA-based NMS will generate less overhead traffic than the SNMP-based NMS. Otherwise, the SNMP-based NMS will create less overhead traffic. To validate this statement, we ran two experiments with the number of managed devices involved set to 6 and 50 respectively. When the number of managed devices involved in the task was set to 6, the experiments are carried out 15 times and the overhead traffic of both MA-based NMS and SNMP-based NMS were captured and presented in the figure 1 below. When the number of managed devices involved in the task was set to 50, the experiments are carried out 15 times and the overhead traffic of both MA-based and SNMP-based NMS were captured and presented in the figure 2 below. From the figures 1 and 2, it can be seen that, when the number of managed devices is 6, which is between the range of 1 and 40, the MA-based NMS produces less overhead traffic on the production network. When the number of managed devices is 50, which is outside the range of 1 and 40, the SNMP-based NMS creates less overhead traffic on the production network. This validates our hypothesis.

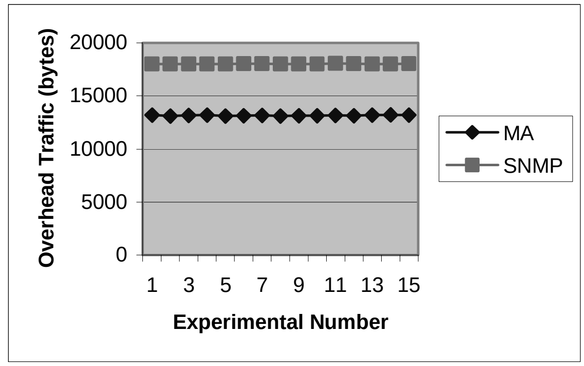

Figure 1. Results for 6 Managed Devices

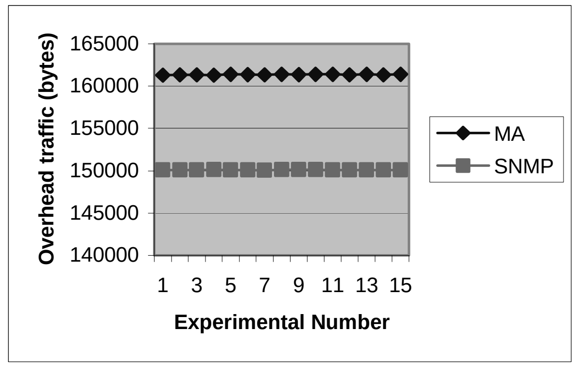

Figure 2. Results for 50 Managed Devices

3.2 Scenario 2

The management task is to get values of CPU usage of all managed devices ten times over a period of time for statistical purposes. The mobile agent employed in the MA-based NMS is not optimised in size. To carry out this task by SNMP-based NMS, we set the size of the SNMP message to 1500 bytes. Since the task is to get the CPU usage of managed devices ten times over a period of time, the number of SNMP request messages needed per a managed device is ten. With Ssnmp = 1500 and Nsnmp = 10, based on the formula from section 2.1, the overhead traffic generated by SNMP-based NMS is: $OT_{snmp} = 30000 * x$.

Since the purpose of this scenario is to carry out a complex task when the size of mobile agent is not optimal, we kept all parameters of MA-based NMS the same as those of the scenario 2. With Sma = 8000, Nma = 1 and Ima = 50, based on the formula from section 2.2, the overhead traffic generated by the MA-based NMS would be: $OT_{ma} = 25 * x^2 + 7975 * x + 50$.

To set up the condition when MA-based NMS to perform better than SNMP-based NMS, we compare the overhead traffic generated by both systems: $OT_{snmp} > OT_{ma}$

=> 0.0022 < x < 880.1.

Hence, to perform the management task presented in this scenario, the conditions established by our formulae states that: if the number of managed devices is between 1 and 880 inclusive, then

MA-based NMS will generate less overhead traffic than the SNMP-based NMS. Otherwise, the SNMP-based NMS would generate less overhead traffic. To validate this statement, we ran one experiment with the number of managed devices involved set to one. When the number of managed devices involved in the task is set to 1, the experiments are carried out 15 times and the overhead traffic of both MA-based and SNMP-based NMS were captured and presented in the figure 3 below. From figure 3, it can be seen that the MA-based NMS produce less overhead traffic than the SNMP-based NMS when the number of managed devices is one, between the range of 1 and 880. This validates our hypothesis. Since only four workstations were employed for this project, we failed to run the experiment when the number of managed devices set to 900, outside the calculated range. The mobile agent stopped responding after approximate 200 iterations. This is resulted from the limited number of workstations and excessive parallel threads utilised to simulate a large-scale network. It is perceived that, if there are more workstations available, the experiment will succeed even when the number of managed devices is large.

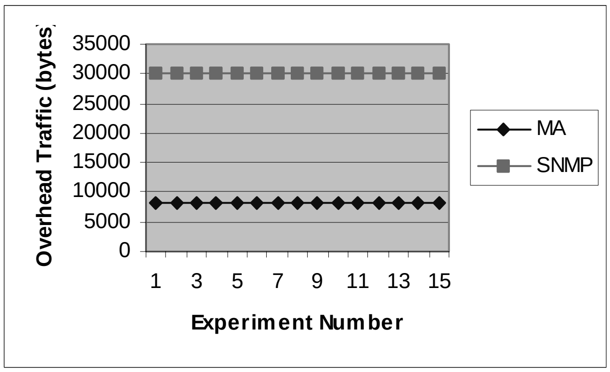

Figure 3. Results for 1 Managed Device

4. CONCLUSIONS

We have proposed and developed a mathematical model for estimating the overhead traffic generated by SNMP-based NMS as well as MA-based NMS. These two models are then analysed and compared to derive a mathematical formula to compute the conditions, based on the relationships between parameters of the management task, to determine what NMS model, MA-based or SNMP-based, should be employed, in order to minimise the overhead traffic generated on the production network. We have experimentally established the validity of our model. Our model is unique in that it takes into account a variety of parameters and their impact on performance (we have termed this the task complexity) rather than examining parameters in isolation. Furthermore, our analysis clearly that there is range (in terms of the number of nodes) within which mobile agents can be expected provide performance benefits over SNMP and that this range can be analytically determined a priori.

5. REFERENCES

[1] Adhicandra, I., Pattinson, C., and Shagouei, E., “Using Mobile Agents to Improve Performance of Network Management Operations”, Postgraduate Networking Conference (PGNET 2003), 2003, Liverpool, UK, Available Online: http://www.cms.livjm.ac.uk/pgnet2003/submissions/Paper-12.pdf

[2] Baldi, M., and Picco G, P., “Evaluating the Tradeoffs of Mobile Code Design Paradigms in Network Management Applications” Proceedings of the 20th International Conference on Software Engineering April 1998 pp146-155.

[3] Bieszczad, A., Pagurek, B., and White, T., “Mobile Agents for Network Management”, IEEE Communications Surveys, September 1998.

[4] Gavalas, D., Greenwood, D., Ghanbari, M., and O’Mahony, M., “An Infrastructure for Distributed and Dynamic Network Management Based on Mobile Agent Technology”, 1999. ICC’99. 1999 IEEE International Conference on Communications. 1999. Vol.2. pp. 1362-1366.

[5] Nicklisch, J., Quittek, J., Kind, A., and Arao, S., “INCA: An Agent-based Network Control Architecture”. Proceedings of the 2nd International Workshop on Intelligent Agents for Telecommunication Applications (IATA’98), LNCS vol. 1437, pp. 143-155, 1998.

[6] Perpetuum, “Perpetuum Mobile Procura Project” [Internet] http://www.sce.carleton.ca/netmanage/perpetuum.shtml [Accessed 13/8/2003].

[7] Project James, “A Mobile Agent Platform for the Management of Telecommunication and Data Networks” [Internet] http://james.dei.uc.pt/james2/banner.html [Accessed 23/8/2003].

[8] Puliafito, A., and Tomarchio, O., “Using mobile agents to implement a flexible network management strategies”, Computer Communication Journal, 23(8): 708-719, April 2000.

[9] Rubinstein, M, G., and Duarte, O., “Evaluating the Performance of mobile agents in Network Management” Global Telecommunications Conference, 1999. GLOBECOM’99, Vol. 1a 1999 pp. 386-390.

[10] Rubinstein, M, G., and Duarte, O., and Pujolle, G., “Improving Management Performance by Using Multiple Mobile Agents” Agents 2000, Barcelona Spain, 2000, pp. 165-166.

[11] Rubinstein, M, G., and Duarte, O., and Pujolle, G., “Evaluating tradeoffs of Mobile Agents in Network Management”, Networking and Information Systems Journal, 2(2):237-252, 1999.

[12] Zhang, D., “Network management using mobile agent”, Proceedings of the International Conference on Communication Technology, Oct. 1998 Beijing, China, Page(s): 22-24, Vol. 2.