IMPLEMENTAÇÃO E SIMULAÇÃO DE UMA APLICAÇÃO DE GERENCIAMENTO UTILIZANDO AGENTES MÓVEIS

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

Grupo de Teleinformática e Automação, Universidade Federal do Rio de Janeiro, COPPE/EE
CP 68504 - 21945-970 - Rio de Janeiro, RJ - Brasil

RESUMO

Este trabalho analisa o desempenho de agentes móveis no gerenciamento de redes, comparando-o com o desempenho do SNMP. Protótipos de uma aplicação de gerenciamento foram implementados em uma rede local e simulações para topologias semelhantes à da *Internet* foram realizadas. Os resultados indicam que o agente móvel possui um desempenho melhor do que o do SNMP quando o número de elementos gerenciados se encontra entre dois limites determinados pelo número de mensagens que passam por um *backbone* e pelo tamanho do agente que cresce com os dados coletados.

ABSTRACT

This paper analyzes mobile agent performance in network management, comparing it with the SNMP performance. Prototypes of a management application were implemented on a LAN and simulation studies were performed on topologies similar in shape to the Internet. Results show that the mobile agent performs better than the SNMP when the number of managed elements ranges between two limits determined by the number of messages that pass through a backbone and the size of the mobile agent that grows with collected data.

1. INTRODUÇÃO

O protocolo de gerenciamento SNMP (*Simple Network Management Protocol*) [1] baseia-se no paradigma cliente-servidor, no qual o gerente centraliza as informações e fornece a ordem para a execução de ações corretivas, e o agente interage com a MIB (*Management Information Base*) e executa as ordens do gerente. Nesses protocolos, as operações disponíveis à estação de gerenciamento para o acesso à MIB são de baixo nível e é necessário o acesso a uma enorme quantidade de dados, que é obtida através de varredura.

A interação de granularidade fina e a varredura periódica geram um tráfego intenso que sobrecarrega a estação de gerenciamento, resultando em problemas de escalabilidade. O gerenciamento de redes pode ser distribuído e escalado através do uso de agentes móveis, que são programas que ajudam um usuário a realizar tarefas na rede, agindo em interesse desse usuário. Esses agentes podem mover-se para o lugar onde os dados estão armazenados e selecionar as informações que o usuário necessita; economizando-se banda passante, tempo e dinheiro.

Este trabalho visa avaliar o desempenho dos agentes móveis no gerenciamento de redes. Algumas análises semelhantes estão sendo realizadas. Baldi et al. [2] avaliam o uso de paradigmas de códigos móveis no gerenciamento de redes através de um modelo quantitativo. Bohoris et al. [3] apresentam uma comparação de desempenho entre agentes móveis, CORBA e Java-RMI, utilizando um elemento em uma rede ATM. Gavalas et al. [4] e Sahai e Morin [5] analisam o desempenho do SNMP e do agente móvel quanto à utilização da banda passante e ao tempo de resposta na busca de variáveis da MIB do SNMP em uma rede local.

A motivação deste trabalho surgiu do fato de que nenhuma das pesquisas descritas anteriormente trata do problema da escalabilidade do gerenciamento com agentes móveis de uma rede complexa de muitos nós, com uma topologia semelhante à da *Internet*. Dois protótipos de uma aplicação que obtém variáveis da MIB-II [6], um com agentes móveis e outro somente utilizando o SNMP, são implementados e testados em uma rede local *Ethernet*. A partir da obtenção de parâmetros relativos às implementações, são obtidos resultados para topologias semelhantes à da *Internet*. Alguns resultados preliminares de implementação e de simulação são apresentados em [7, 8].

Este trabalho está organizado da seguinte forma. Na Seção 2, são apresentados os principais sistemas de gerenciamento de redes. Na Seção 3, são descritos os protótipos implementados e os resultados das medições. Na Seção 4, são descritos os resultados para topologias semelhantes à da *Internet*. Por último, as conclusões são descritas na Seção 5.

2. O GERENCIAMENTO DE REDES

No SNMP, as operações disponíveis à estação de gerenciamento para o acesso à MIB são de baixo nível, logo a interação agente-gerente de granularidade fina não possui uma boa escalabilidade devido ao tráfego intenso e à sobrecarga computacional na estação de gerenciamento [2].

Um gerenciamento de redes mais descentralizado é adotado no SNMPv2 [1] (SNMP versão 2), no qual podem existir múltiplas estações de gerenciamento no mais alto nível de hierarquia, chamadas servidores de gerenciamento. Cada servidor é responsável por gerenciar agentes, porém responsabilidades podem ser delegadas a um gerente intermediário, também chamado agente procurador. A versão 3 do SNMP, o SNMPv3, incorpora um novo esquema de segurança [1] e foi criada para ser utilizada com as versões anteriores do protocolo SNMP.

O RMON (*Remote MONitoring*) [9] utiliza equipamentos para realizar o monitoramento pró-ativo de redes locais em segmentos locais ou remotos. Esses monitores provêem informações sobre enlaces, conexões entre estações, modelos de tráfego e status dos nós da rede.

Essas propostas de descentralização levam a uma redução do tráfego ao redor da estação de gerenciamento. Porém, como o poder de cálculo dos nós da rede tem aumentado, torna-se possível delegar funções de gerenciamento ainda mais complexas aos nós. Além disso, de modo a satisfazer as diversas necessidades das redes atuais, novos sistemas de gerenciamento de redes que podem analisar dados, tomar decisões e obter medidas pró-ativas para manter a qualidade de serviço da rede devem ser desenvolvidos. Por isso, os agentes móveis podem ser utilizados para descentralizar o processamento e o controle da estação de gerenciamento.

Tanto o RMON quanto o SNMPv2 possuem algumas desvantagens se comparados a um gerenciamento de redes que utiliza agentes móveis. O RMON tipicamente utiliza um monitor para cada segmento de rede, logo o gerenciamento de múltiplos segmentos gera um crescimento considerável do custo [4]. Além disso, tanto o RMON quanto o SNMPv2 só permitem que as operações de controle sejam criadas ou alteradas quando da configuração dos componentes de gerenciamento.

As principais vantagens que podem justificar a utilização de agentes móveis no gerenciamento de redes são a redução de custo através de uma compressão semântica a nível global ou de domínio, que filtra e seleciona somente as informações relevantes; o processamento assíncrono que permite um desacoplamento do nó de origem; a flexibilidade, uma vez que o comportamento do agente pode ser alterado em tempo real, e a autonomia pois o agente pode tomar decisões, realizando um gerenciamento reativo com delegação de tarefas.

Devido à pequena difusão do SNMPv2 em relação à do SNMP e como o gerenciamento de redes no SNMP não é escalável quando o tamanho ou a complexidade da rede cresce, por causa da centralização do processamento e do controle, os agentes móveis podem ser utilizados para aumentar a escalabilidade do gerenciamento de redes, sendo importante descobrir sob que condições esses agentes aumentam a eficiência do gerenciamento.

3. A IMPLEMENTAÇÃO DE UMA APLICAÇÃO DE GERENCIAMENTO

Dois protótipos de uma aplicação que obtém variáveis (informações de gerenciamento) da MIB-II [6] foram implementados, um baseado em agentes móveis e outro somente baseado no SNMP.

A infra-estrutura Mole [10] é utilizada na implementação do protótipo com agentes. Nesse sistema, os agentes podem mover, se comunicar entre si e interagir com o subsistema de comunicação. Dois tipos diferentes de agentes são providos: os agentes do sistema e os agentes do usuário. Os agentes do sistema são normalmente interfaces para recursos fora da infra-estrutura e possuem mais direitos do que os agentes do usuário, porém não podem migrar. Os agentes do usuário podem migrar mas só têm acesso aos recursos disponibilizados através da infra-estrutura.

A plataforma Mole utiliza o protocolo TCP para transferir os agentes móveis, que são implementados na linguagem Java. Um esquema de migração fraca é provido na Mole, ou seja, somente o estado relativo aos dados do agente é transferido quando da migração do agente; conseqüentemente, o programador é responsável por codificar o estado de execução do agente em variáveis do programa. Essa migração é implementada através do uso da serialização de objetos do Java. Depois que um *thread* do agente chama o método *migrateTo()*, todos os *threads* pertencentes ao agente são suspensos. É feita então a serialização do agente, na qual uma representação independente do sistema é criada e enviada para o destino que reinstancia o agente. Um novo *thread* é iniciado e logo que esse *thread* assume o controle do agente, uma mensagem é enviada à fonte que finaliza todos os *threads* pertencentes ao agente e remove-o do sistema.

Os dois protótipos implementados, um com agentes móveis e o outro sem, utilizam o protocolo SNMP para obter variáveis da MIB-II. A biblioteca SNMP da AdventNet [11] contém interfaces de programação de aplicações (APIs) que tornam mais simples a implementação de aplicações que utilizam o protocolo SNMP. Foi utilizada a versão 2.2 do AdventNet SNMPv1.

O *daemon* snmpd, que vem com o Linux Red Hat, foi utilizado como um agente SNMP. As versões utilizadas do pacote foram a 3.5.3 (para as máquinas com Red Hat 5.2) e a 4.0.1 (para o Red Hat 6.x).

3.1. As Duas Implementações

A implementação baseada em agentes móveis (Figura 1) consiste em um agente móvel, que migra para todos os elementos de rede a serem gerenciados, em um agente SNMP, que acessa as variáveis da MIB-II, e em um agente tradutor, que converte o pedido do agente móvel para o formato do protocolo SNMP. O agente móvel migra para um elemento de rede (arco 1 da Figura 1) e se comunica via chamada remota de procedimentos com o agente tradutor (arco 2). Esse agente tradutor envia um pedido (PDU GetRequest do SNMP) para o agente SNMP (arco 3) e obtém a resposta (arco 4) que é repassada ao agente móvel (arco 5). O agente móvel então segue para o próximo elemento (arco 6) e recomeça a sua execução. Ao terminar sua tarefa, que consiste na visita a todos os elementos de rede, o agente móvel retorna à estação de origem (arco n).

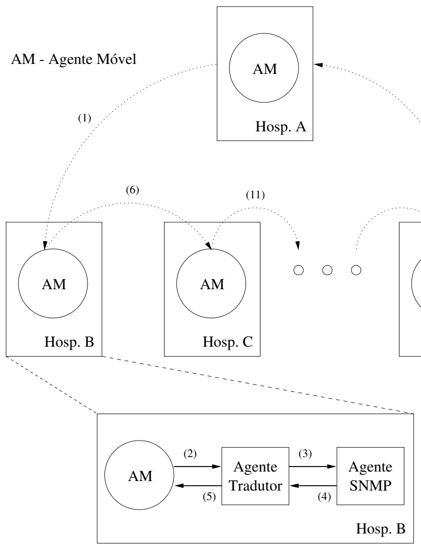

Figura 1: O gerenciamento por agente móvel.

Na implementação que só utiliza o SNMP, o modelo tradicional desse protocolo foi utilizado. O gerente envia um pedido para um agente SNMP que responde a esse gerente. São enviados pedidos a todos os elementos da rede, um após o outro, ou seja, um novo pedido é iniciado após receber a resposta do anterior, até que o último elemento de rede receba um pedido e envie a resposta ao gerente. Esse gerente foi implementado na linguagem Java.

3.2. O Estudo Experimental

Um estudo experimental é realizado, de modo a avaliar a escalabilidade das duas abordagens descritas anteriormente.

A topologia utilizada consiste em uma estação de gerenciamento (estação A) e em dois elementos gerenciados de rede (estações B e C), interconectados através de uma rede local *Ethernet* de 10 Mbps. As estação A é um Pentium MMX 233 Mhz, com 128 Moctetos de memória e rodando Linux Red Hat versão 6.2. As estações B e C são Pentiuns II 350 MHz, respectivamente com 64 Moctetos e 128 Moctetos de memória e rodando também Linux Red Hat versões 6.1 e 5.2.

De modo a avaliar o desempenho do gerenciamento para um grande número de elementos de rede, os dois elementos B e C são repetidos alternadamente; por exemplo, se o número de estações a serem gerenciadas é cinco, o itinerário a ser seguido pelo agente é B, C, B, C , B e A.

O parâmetro de desempenho considerado é o tempo de resposta na obtenção da variável da MIB-II *ifInErrors*, a qual representa o número de pacotes recebidos e descartados por causa de erros.

O JDK (*Java Development Kit*) 1.1.7 versão 3 foi utilizado. Todas as medidas foram realizadas cedo pela manhã ou à noite, de modo a limitar as variações de desempenho da rede, que influenciam os resultados. As implementações foram testadas sob as mesmas condições, utilizando um mesmo itinerário preestabelecido. Os testes foram realizados com os ambientes de execução dos agentes móveis sendo executados sem interrupções. O número de elementos gerenciados varia entre 1 e 250. Para as medições realizadas, foram calculados intervalos de confiança de 99% relativos à média de dez medidas. Esses intervalos estão representados nos gráficos através de barras verticais.

O efeito do número de elementos gerenciados no tempo de resposta foi analisado nas medições realizadas. Em todas as figuras são apresentadas as médias das medidas.

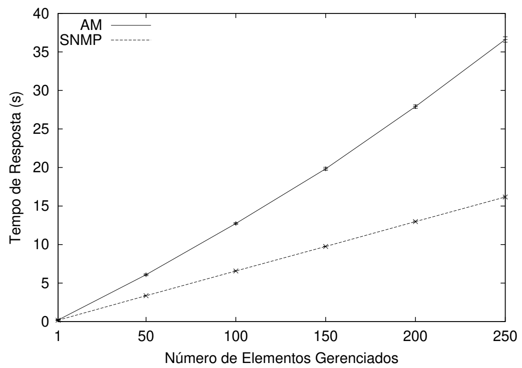

Figura 2: Tempo de resposta para o agente e para o SNMP.

O tempo de resposta para o SNMP cresce proporcionalmente ao número de elementos gerenciados, pois o tempo para gerenciar um elemento é praticamente o mesmo para todos os elementos (Figura 2). Para o agente móvel, o tempo de resposta cresce mais rápido conforme o número de elementos aumenta, por causa do tamanho do agente móvel que cresce com as variáveis coletadas a cada elemento de rede. Na topologia utilizada, o desempenho do SNMP é bem melhor do que o do agente móvel.

Para melhor analisar os resultados, foram realizadas medidas dos tempos de acesso às MIBs (envio/recebimento de PDUs GetRequest/GetResponse do SNMP) e desse tempo acrescido aos tempos de comunicação agente móvel-agentes tradutores e agentes tradutores-agente móvel (Figura 1).

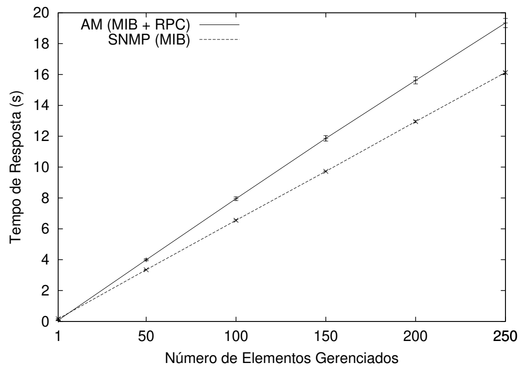

Figura 3: Tempo de resposta relativos aos acessos às MIBs.

Na Figura 3, são apresentados os tempos de resposta para os acessos às MIBs e para as RPCs relativas às comunicações agente móvel-agentes tradutores. Neste experimento, em relação ao SNMP, pode-se dizer que para um número de estações gerenciadas igual a 250, aproximadamente 99,6 % do tempo total é gasto nos acessos às MIBs. Para o agente móvel, quando o número de estações é 250, os acessos às MIBs e as RPCs ocupam 52,8 % do tempo total. Pode-se afirmar que as consultas às MIBs crescem de modo linear, em função do número de elementos gerenciados e gastam aproximadamente 65 ms para cada elemento para o caso do SNMP. Essas consultas às MIBs acrescidas das RPCs agente móvel-agentes tradutores também crescem linearmente e gastam aproximadamente 78 ms para cada elemento.

O tempo restante do agente móvel é calculado através da diferença do tempo total do agente móvel e do tempo das MIBs e das RPCs e é apresentado na Figura 4. Como neste experimento o tempo de transmissão do agente é muito pequeno em relação aos demais tempos envolvidos, o tempo restante corresponde a gastos relativos à infra-estrutura, como a serialização/desserialização, a criação de *threads* e o envio de mensagens internas. Pode-se dizer que o tempo de resposta cresce exponencialmente com o número de elementos gerenciados, logo a curva da Figura 4 pode ser aproximada por:

$$y = a^x, \qquad (1)$$

onde a = 1,01176 (Figura 4). Essa aproximação foi escolhida por permitir, de maneira simples, a sua utilização nas simulações realizadas para topologias mais gerais (Seção 4).

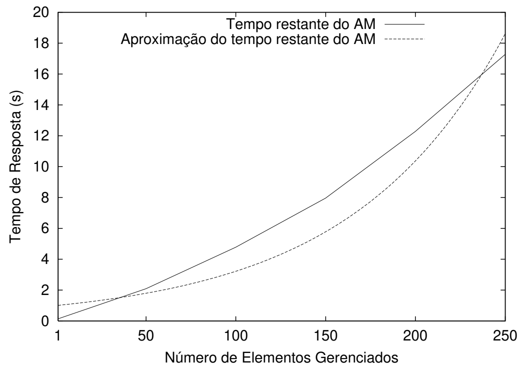

Figura 4: Tempo restante do agente móvel.

4. A ANÁLISE DO DESEMPENHO DO GERENCIAMENTO ATRAVÉS DE SIMULAÇÕES

A análise do uso de agentes móveis no gerenciamento de redes também foi realizada através de simulações.

O simulador de rede *ns* (*Network Simulator*) é utilizado neste trabalho [12]. Esse simulador de eventos discretos oferece uma infra-estrutura para simulações em redes de computadores com abstrações para nós e enlaces e possui diversos protocolos já implementados. Nas simulações deste trabalho, foram utilizadas as funções de rede local *Ethernet*, de topologias similares à da *Internet* e dos protocolos TCP e UDP. Alguns módulos dos protocolos TCP e do UDP foram alterados, para permitirem o envio de agentes móveis.

O *ns* trata o envio de pacotes através de uma rede e geralmente não considera o tempo de processamento da camada aplicação em cada nó. Por isso, vários parâmetros relativos ao gerenciamento de redes foram adicionados ao modelo de simulação, de modo a tornar os resultados da simulação mais fiéis a uma implementação real. Apesar dos parâmetros das implementações serem dependentes da infra-estrutura de agentes, do sistema operacional e da utilização das máquinas, o uso desses parâmetros permite a obtenção de resultados de simulação mais perto dos reais. Nas simulações foram considerados os parâmetros mostrados na Tabela 1.

Tabela 1: Os parâmetros utilizados nas simulações.

Parâmetro	Valor
Tamanho inicial (código) do agente	1500 octetos
Tamanho do pedido para *ifInErrors*	42 octetos
Tamanho da resposta para *ifInErrors*	51 octetos
Tempo da MIB para o agente por nó	78 ms
Tempo da MIB para o SNMP por nó	65 ms
Relativo ao tempo restante do agente	1,01176

O modelo das simulações assume que os enlaces e nós não possuem carga e que os enlaces não tem perdas. O tamanho máximo de segmentação (*Maximum Segmentation Size* - MSS) utilizado nas simulações é igual a 1500 octetos; com isso, não há fragmentação de mensagens SNMP por essas serem pequenas. Já para o agente móvel, como o seu tamanho inicial é de 1500 octetos, logo após a visita ao primeiro elemento, o seu tamanho estará maior do que a MSS, o que fará com que o agente seja fragmentado e enviado em diferentes pacotes, prejudicando o seu desempenho. Cada pedido de uma variável é enviado em uma mensagem diferente. Em todas as simulações, o agente móvel segue um itinerário preestabelecido. O agente móvel utiliza como protocolo de transporte o TCP-Reno, por esse ser atualmente de grande utilização na *Internet* e o protocolo UDP é usado nas simulações do SNMP.

Foram utilizados dois tipos de topologias em todas as simulações. O primeiro tipo consiste em elementos dispostos em uma rede local *Ethernet* com 250 nós, banda passante igual a 10 Mbps e latência de 10 μs. O segundo tipo procura ser próximo das estruturas encontradas na *Internet*. Esse tipo de topologia é chamado *transit-stub*, pois cada domínio de roteamento na *Internet* pode ser classificado em *transit* ou *stub*. O propósito dos domínios *transits* é interconectar domínios *stubs* de modo eficiente. Um domínio *transit* compreende um conjunto de nós do *backbone*, nós que estão tipicamente conectados entre si. Em um domínio *transit*, cada nó do *backbone* também pode conectar domínios *stubs*.

Esse tipo de topologia pode ser utilizado no gerenciamento de redes de uma organização do tipo matriz-filiais, na qual uma matriz quer gerenciar suas filiais espalhadas geograficamente. A estratégia de gerenciamento utilizada neste trabalho para topologias *transit-stubs* considera que a estação de gerenciamento está localizada em um nó de um domínio *stub* e que os elementos a serem gerenciados pertencem aos outros domínios *stubs* (Figura 5). No caso do conjunto matriz-filiais, a estação de gerenciamento da matriz gerencia os roteadores das filiais, sendo que cada filial (representada por um *stub*) contém vários roteadores.

O parâmetro de desempenho considerado é o tempo de resposta na obtenção da variável da MIB-II [6] *ifInErrors*.

Foram feitas simulações utilizando a topologia de rede local de modo a comparar o modelo de simulação aos resultados da implementação da Seção 3.

Na Figura 6 são apresentados os tempos de resposta para o agente móvel e para o SNMP, na implementação e na simulação. Observa-se que os modelos simulados reproduzem os modelos implementados, havendo uma pequena diferença no tempo do agente móvel devido à aproximação utilizada para o tempo restante do agente (Seção 3.2).

O desempenho do agente móvel em uma situação mais próxima da encontrada na *Internet*, na qual as latências são

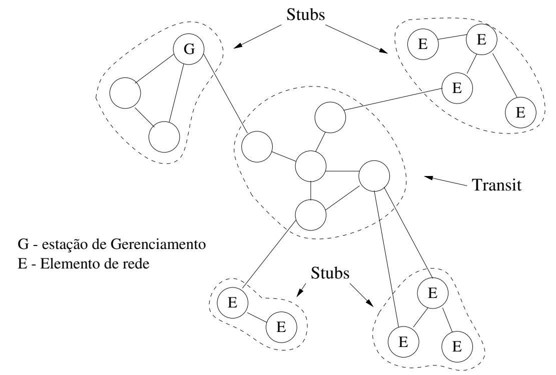

Figura 5: O gerenciamento em uma topologia *transit-stub*.

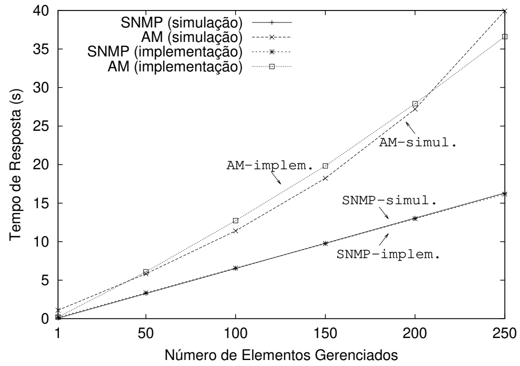

Figura 6: Tempo de resposta para o agente e para o SNMP.

bem maiores do que as das redes locais, também é avaliado. Para esse fim, foram utilizadas três topologias *transit-stubs* aleatórias criadas pelo gerador de topologias GT-ITM [13]. As topologias possuem 272 nós, banda igual a 2 Mbps e latência de algumas dezenas de milisegundos. Nessas simulações, a estação de gerenciamento controla grupos de 16 elementos de rede que é o número de nós de cada um dos *stubs*. Todos os elementos de um *stub* são acessados, após isso, o próximo *stub* é gerenciado, até que todos os 16 *stubs* sejam acessados.

O comportamento do agente móvel praticamente não muda com a topologia (Figura 7), porém para o SNMP, há uma pequena variação nos tempos de resposta para as três topologias. Essa variação se deve ao grande número de pacotes do SNMP que passam pelos enlaces do *backbone* (*transit*) e à configuração dos nós do *backbone* que varia com a topologia. Na Figura 7, os tempos médios de resposta também são apresentados. Para um número muito pequeno de elementos gerenciados, o SNMP possui um desempenho melhor pelo fato do tamanho do pacote do SNMP ser menor do que o tamanho inicial do agente móvel. Conforme o número de elementos gerenciados aumenta, o tempo

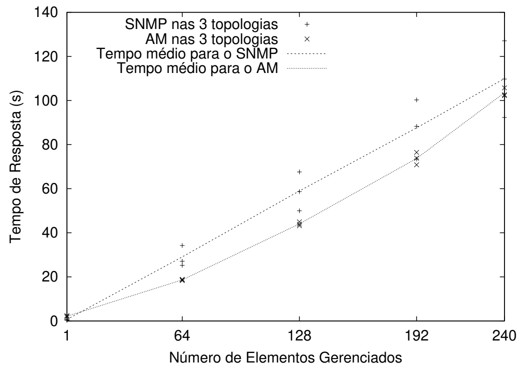

Figura 7: Tempo de resposta para o agente e para o SNMP.

de resposta para o SNMP cresce proporcionalmente, pois o tempo para gerenciar um *stub* é praticamente o mesmo para todos os *stubs*. Para o agente móvel, o tempo de resposta cresce mais rápido com o aumento do número de elementos gerenciados, devido ao tamanho crescente do agente. Extrapolando-se o resultado obtido, pode-se afirmar que o agente móvel possui um desempenho melhor do que o do SNMP quando o número de elementos gerenciados de rede encontra-se entre dois limites determinados, respectivamente, pelo número de mensagens que passam pelo *backbone* e pelo tamanho crescente do agente.

5. CONCLUSÕES

Este trabalho analisou o desempenho do gerenciamento por agentes móveis, comparando-o ao gerenciamento através do SNMP, no que se refere ao tempo de resposta.

Foram implementados dois protótipos de uma aplicação que obtém variáveis da MIB-II: um baseado em agentes móveis e outro somente baseado no SNMP. O desempenho do SNMP é bem melhor do que o do agente móvel, pois a topologia utilizada nas medições é bastante desfavorável ao agente móvel, uma vez que a pequena latência na rede *Ethernet* torna o tempo de transmissão desprezível em relação aos tempos de processamento envolvidos.

Simulações das duas implementações também foram realizadas no simulador de redes *ns*, de modo a se obter resultados em topologias genéricas maiores, semelhantes às encontradas na *Internet*. Os agentes móveis possuem um melhor desempenho do que o do SNMP quando o número de elementos gerenciados encontra-se entre dois limites determinados, respectivamente, pelo número de mensagens que passam pelo *backbone* e pelo tamanho do agente que cresce com as variáveis coletadas a cada elemento de rede.

De um modo geral, conclui-se que o uso de agentes móveis no gerenciamento de redes é uma boa solução no caso de haver diversas sub-redes a serem gerenciadas remotamente; principalmente se as ligações entre a estação de gerenciamento e as estações a serem gerenciadas forem de alto custo (pequena banda passante e grande latência).

6. REFERÊNCIAS

[1] W. Stallings, "SNMP and SNMPv2: The infrastructure for network management", *IEEE Communications Magazine*, vol. 36, no. 3, pp. 37–43, março de 1998.

[2] M. Baldi e G. P. Picco, "Evaluating the tradeoffs of mobile code design paradigms in network management applications", in *20th International Conference on Software Engineering (ICSE'98)*, Kyoto, Japão, pp. 146–155, abril de 1998.

[3] C. Bohoris, A. Liotta e G. Pavlou, "Software agent constrained mobility for network performance monitoring", in *6th IFIP Conference on Intelligence in Networks (SmartNet 2000)*, Viena, Áustria, pp. 367–387, setembro de 2000.

[4] D. Gavalas, D. Greenwood, M. Ghanbari e M. O'Mahony, "Advanced network monitoring applications based on mobile/intelligent agent technology", *Computer Communications*, vol. 23, no. 8, pp. 720–730, abril de 2000. Elsevier Science.

[5] A. Sahai e C. Morin, "Enabling a Mobile Network Manager (MNM) through mobile agents", in *Second International Workshop on Mobile Agents*, Stuttgart, Alemanha, setembro de 1998.

[6] K. McCloghrie e M. Rose, "Management information base for network management of TCP/IP-based internets: MIB-II". RFC 1213, março de 1991.

[7] M. G. Rubinstein, O. C. M. B. Duarte e G. Pujolle, "Evaluating the network peformance management based on mobile agents", *Lecture Notes in Computer Science*, vol. 1931, pp. 95–102, setembro de 2000. ISSN 03029743, ISBN 3540410694, Springer-Verlag.

[8] M. G. Rubinstein e O. C. M. B. Duarte, "Evaluating the performance of mobile agents in network management", in *IEEE Global Telecommunications Conference (GLOBECOM'99)*, Rio de Janeiro, Brasil, pp. 386–390, dezembro de 1999.

[9] S. Waldbusser, "Remote network monitoring management information base". RFC 1757, fevereiro de 1995.

[10] J. Baumann, F. Hohl, M. Straβer e K. Rothermel, "Mole - concepts of a mobile agent system", *World Wide Web*, vol. 1, no. 3, pp. 123–137, 1998. Baltzer Science Publishers.

[11] Advent Network Management Inc., "AdventNet SNMP release 2.0". http://www.adventnet.com, 1998.

[12] K. Fall e K. Varadhan, "NS Notes and Documentation", relatório técnico, The VINT Project, janeiro de 1999.

[13] E. W. Zegura, K. L. Calvert e M. J. Donahoo, "A quantitative comparison of graph-based models for internet topology", *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 770–783, dezembro de 1997.