

“AVALIANDO O DESEMPENHO DE REDES ATRAVÉS DO SIMULADOR NETWORK SIMULATOR”

Mauro Margalho Coutinho

Universidade da Amazônia
Universidade Federal do Pará
margalho@unama.br

João Crisóstomo Weyl

Albuquerque Costa
Universidade Federal do Pará
jweyl@ufpa.br

Carlos Renato Lisboa

Francês
Universidade Federal do Pará
rfrances@ufpa.br

Universidade Federal do Pará
Campus Universitário do Guamá
Programa de Pós-Graduação em engenharia Elétrica
Rua Augusto Corrêa, 01
CEP 66075-110 - Caixa postal 479
PABX +55 91 211-2121 - Belém - Pará - Brasil

RESUMO

Este artigo busca descrever, de forma sucinta, um dos simuladores de redes de computadores mais utilizados atualmente, o Network Simulator, apresentando ao leitor suas características básicas enfocadas através de exemplos ilustrados graficamente.

1. INTRODUÇÃO

Diante da carência de recursos para se criar laboratórios de redes de computadores bem estruturados, diversos segmentos acadêmicos têm optado por testar e certificar propostas através de técnicas alternativas de avaliação de desempenho, em especial a simulação. Este artigo propõe mostrar uma visão geral de uma das ferramentas de simulação mais utilizadas atualmente no segmento de redes de computadores, o "Network Simulator" (NS), evidenciando como a mesma pode ser utilizada em cursos de graduação e extensão para aguçar a espírito de investigação científica nos alunos. Desenvolvida com o *knowhow* da equipe de Berkeley, gratuita, com código fonte aberto e disponível em diversas plataformas, ela rapidamente se popularizou e vem, a cada nova versão, recebendo contribuições significativas dos mais variados centros de pesquisa. Atualmente o NS, que encontra-se em sua versão 2.1b9a, lançada em julho de 2002, possui suporte inclusive para emulação, o que permite a interação do simulador com o tráfego de uma rede viva tornando os experimentos mais consistentes. [1]

2. O PROJETO VINT

O simulador NS, alcançou repercussão internacional através do projeto VINT (Virtual InterNetwork Testbed) [3]. A proposta básica era criar uma ferramenta que, além de disponibilizar um conjunto básico de elementos de redes (protocolos, políticas de fila, enlaces, roteadores, etc), oferecesse suporte para que outros fossem agregados e interagissem com os já existentes. O principal produto desse esforço coletivo foi um simulador que vem ganhando, cada vez mais, credibilidade. A figura 1 sintetiza a estrutura do Network Simulator (NS) onde, inicialmente, o usuário define seus modelos de tráfego e topologia, através de *scripts* de simulação, utilizando-se de um *frontend*, que no caso do NS é a linguagem OTCL (Object-oriented Tool Command Language) e depois o submete ao núcleo do NS, que é escrito em C++, para ser executado.

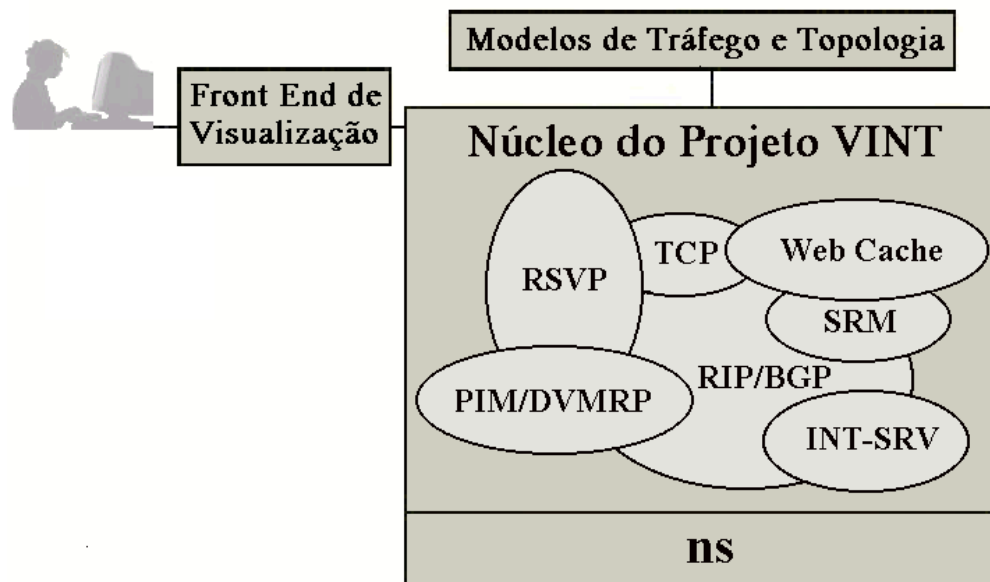


Figura 1 - Estrutura do NS

A idéia por trás de duas linguagens é permitir que os detalhes de simulação, que manipulam com bytes, pacotes, cabeçalhos, implementação de algoritmos, etc, rodem de forma compilada em uma linguagem mais robusta (C++), onde o tempo de execução é o fator primordial. Já os modelos de simulação devem ser escritos em uma linguagem interpretada, pois envolvem pequenas variações de parâmetros e configurações, onde a prioridade é o tempo de interação (mudar e executar novamente o modelo). Nesse caso a linguagem OTCL oferece essa flexibilidade. [1]

Hoje, dentre outros, o simulador oferece suporte para testes com LANs, WANs, redes móveis, redes baseadas em satélites, modelo de propagação via rádio e qualidade de serviço (QoS).

3. COMPONENTES BÁSICOS DO SIMULADOR

3.1 NÓS

Basicamente, a estrutura de um nó no NS é composta por dois classificadores conforme ilustrado na Figura 2. Um de endereços, que serve para verificar se o pacote que está chegando, através do enlace de entrada, tem como destino aquele nó ou deve ser roteado adiante e outro de portas, para entregar o pacote à aplicação correta. Pode-se vincular diversos agentes ou protocolos (TCP, UDP, etc) a um mesmo nó para tornar a simulação mais atrativa.

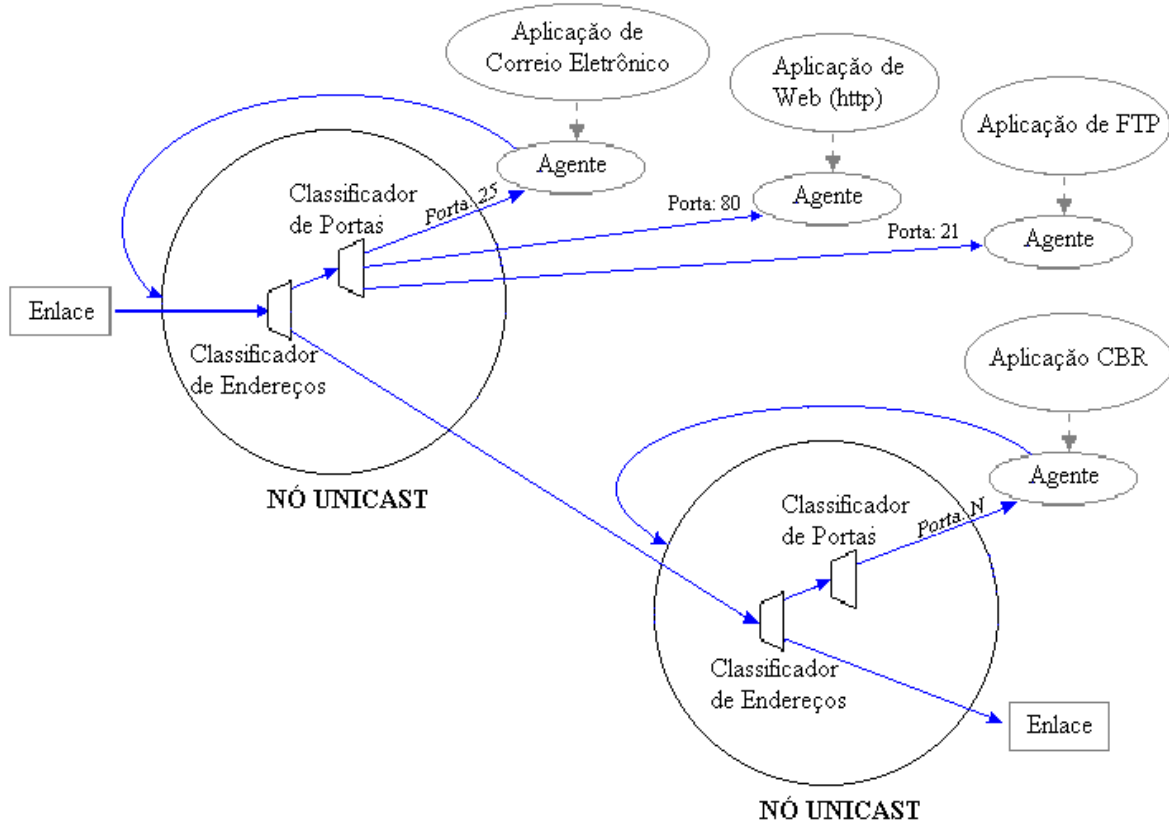


Figura 2 - Estrutura de um nó no NS

3.2 ENLACES

No NS, conforme mostrado na Figura 3, o componente fila está ligado ao enlace, podendo ser parametrizado pelo usuário. Existem diversas políticas já implementadas, sendo que as principais são DROP TAIL, que é a política padrão, SFQ (Stochastic Fair Queueing), WFQ (Weight Fair Queueing) e RED (Random Early Discard). Ao entrar no nó, um pacote é, inicialmente, direcionado para uma fila. Ao sair dela, ele pode ser tanto descartado, o que normalmente ocorre em situações de congestionamento generalizado, como atrasado pelo simulador, conforme especificação do usuário no parâmetro *delay*. O tipo do enlace (simplex ou duplex) também pode ser definido no *script* de simulação reforçando a flexibilidade da ferramenta. [4]

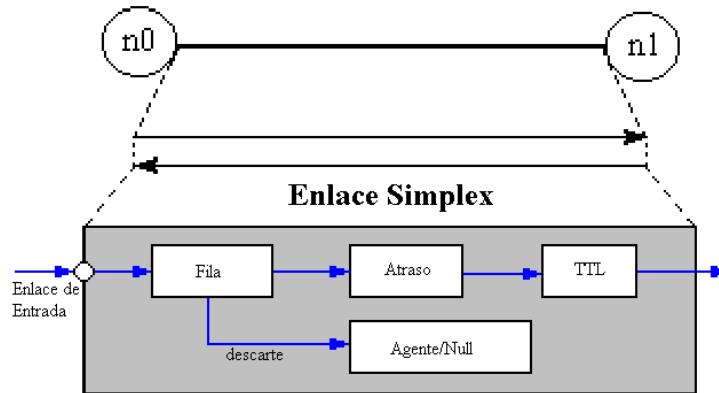


Figura 3 - Estrutura de um enlace no NS

A seqüência de eventos incluída no enlace se inicia com uma fila. Em caso de descarte o pacote é enviado a um agente nulo. De outra forma ele será atrasado e encaminhado ao controle de TTL (Time to Live) que limita o número máximo de saltos antes que a rede descarte o pacote.

4. DEFININDO O CENÁRIO E O TRÁFEGO

Alguns procedimentos são requeridos ao se propor uma simulação. Dentre outros, deve-se criar um cenário e definir a especificidade de tráfego nesse cenário. O exemplo mostrado na Figura 4 apresenta um cenário composto por quatro nós referenciados como **n0**, **n1**, **n2** e **n3**. Entre os nós **n0** e **n2** existe um enlace com velocidade de 2 Megabits por segundo onde o atraso é de 10 milissegundos.

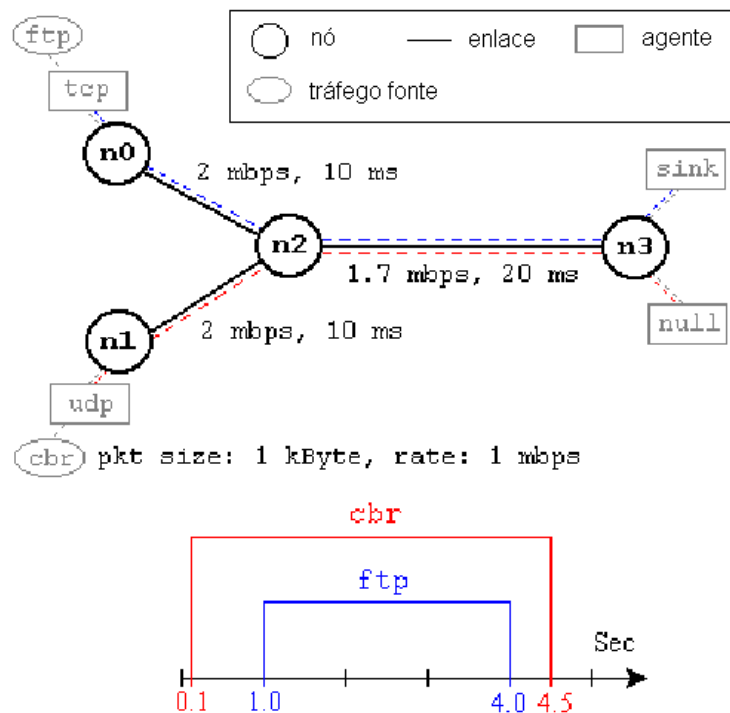


Figura 4 - Cenário de uma simulação

Os mesmos parâmetros servem para o enlace entre os nós **n1** e **n2**. Já entre os nós **n2** e **n3** o enlace é de 1.7 Megabits por segundo com um atraso de 20 milissegundos, o que já se configura como um gargalo da rede. Esses parâmetros são configurados no simulador conforme mostrado na Figura 5, onde `rt_belem` representa o nó origem e `rt_brasília` o destino:

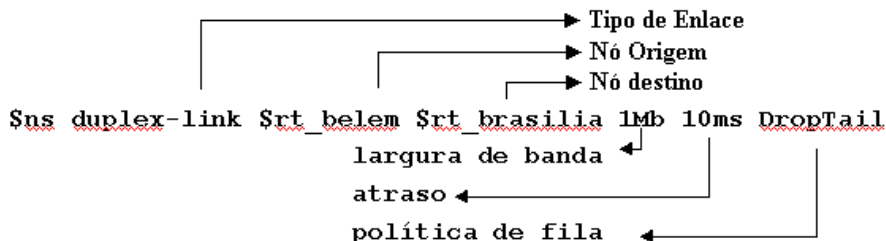


Figura 5 - Parâmetros de configuração dos enlaces no NS

No NS, todo e qualquer protocolo de transporte é definido através de um agente. No exemplo mostrado na figura 4, temos quatro agentes definidos. Um agente TCP (Transmission Control Protocol) vinculado ao nó **n0**, o que indica que desse ponto poderá ser feita uma transmissão orientada a conexão. Um agente UDP (User Datagram Protocol) vinculado ao nó **n1** e dois agentes de recepção vinculados ao nó **n3**. Os agentes de recepção servem para receber os pacotes enviados pelos protocolos de transporte. O agente **null** recebe os pacotes oriundos do protocolo UDP e os anula, enquanto que o agente **sink** recebe os pacotes do protocolo TCP e gera os pacotes de reconhecimento (ACK).

Deve-se também definir o tráfego fonte. No exemplo da figura 4 existem dois desses componentes. Uma aplicação FTP (File Transfer Protocol) ligada ao agente TCP do nó **n0** e uma aplicação CBR (Constant Bit Rate) ligada ao agente UDP do nó **n1**. O tempo de simulação foi fixado em 5 milissegundos, sendo que a aplicação CBR inicia sua transmissão no momento 0.1 e a encerra no momento 4.5 enquanto a aplicação FTP inicia sua transmissão no momento 1.0 e a encerra no momento 4.0.

5. VISUALIZANDO OS RESULTADOS

Há um recurso, especialmente importante, para se acompanhar o andamento do processo de simulação, pois permite a visualização, através de animações, de toda a seqüência de simulação. Pacotes sendo descartados, enlaces tornando-se não operacionais, enfim, todo o comportamento dos fluxos de aplicações pode ser acompanhado através deste módulo do NS chamado NAM (Network Animator) como mostra a figura 6.

Uma aplicação adicional para o NAM é usá-lo como recurso didático no ensino da disciplina redes de computadores, tanto nos curso de graduação como de pós-graduação. Diversos conceitos que exigem uma grande capacidade de abstração por parte do aluno podem ser melhor assimilados diante de um instrumento visual tão poderoso. [5]

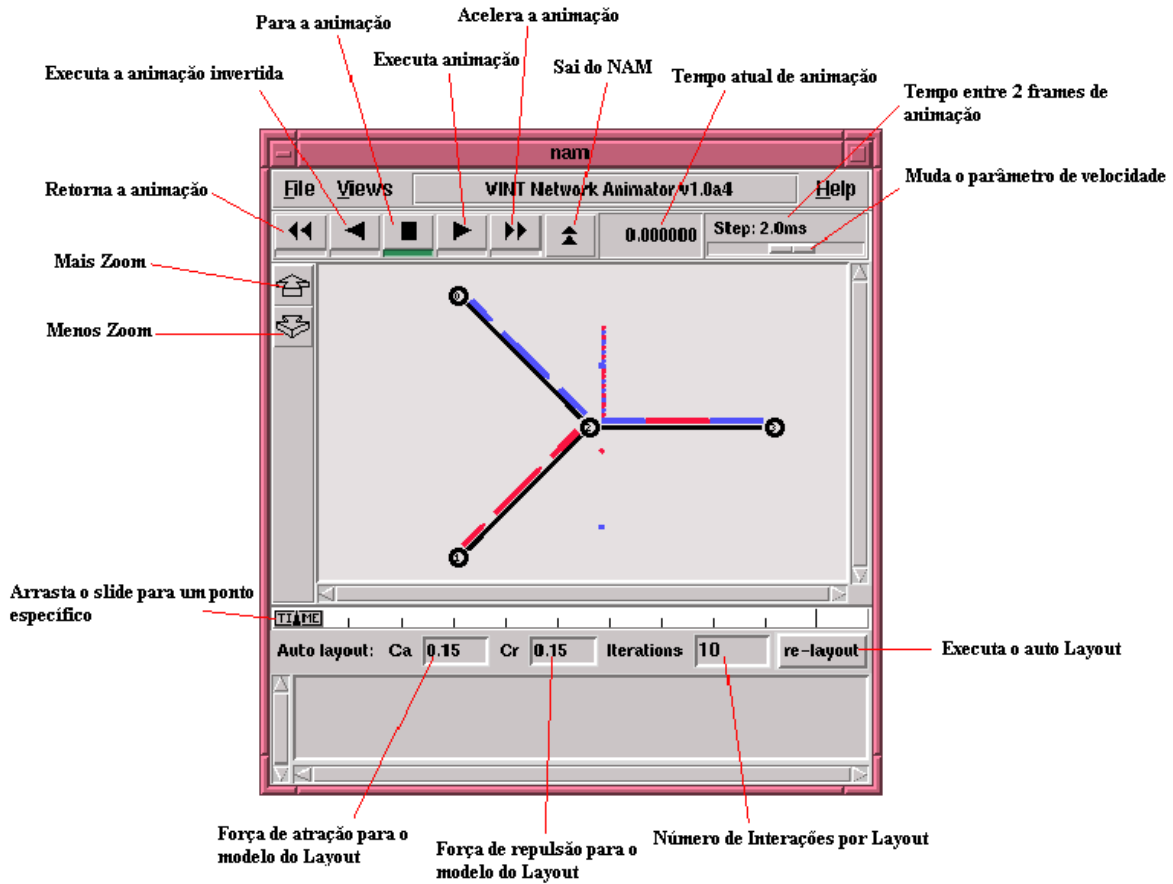


Figura 6 - NAM (Network Animator)

Na figura 7 pode-se visualizar uma simulação que reproduz, em parte, a topologia do projeto Interlegis (interligação dos Poderes Legislativos do Brasil). [9]

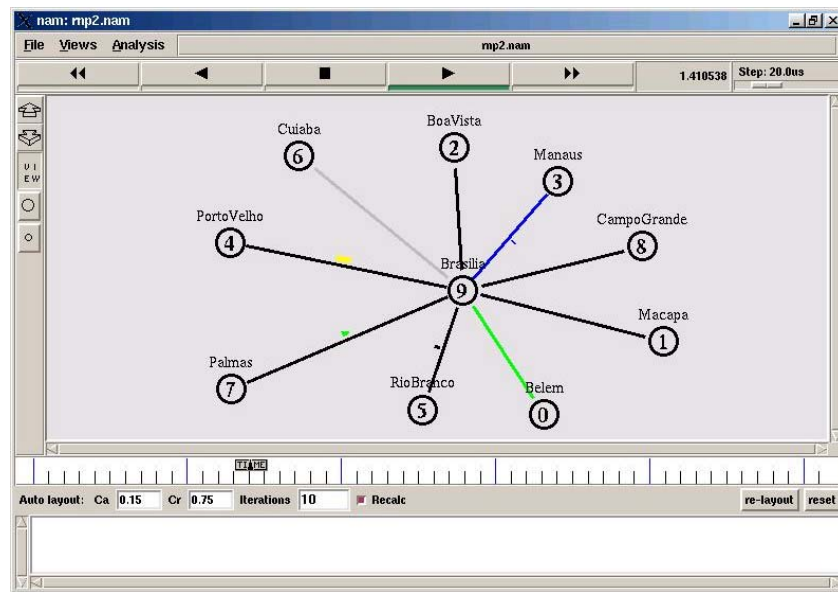


Figura 7 - Topologia parcial do projeto INTERLEGIS

Nesse projeto cada Assembléia Legislativa possui um enlace de 384 kbps com o prédio central, em Brasília. Durante a realização de vídeo conferências, por exemplo, os equipamentos lá instalados, refletem todo o tráfego para os demais estados simulando um roteamento *multicast*. Uma vez que estes mesmos enlaces provêm acesso a Internet, existe a necessidade de se introduzir Qualidade de Serviço (QoS) nessa rede para que os pacotes de vídeo conferência sejam priorizados. [10]

6. AVALIANDO O DESEMPENHO

A ilustração de um estudo comparativo, através de gráficos, constitui-se em uma das principais contribuições do NS. Isso é possível através do módulo XGRAPH.

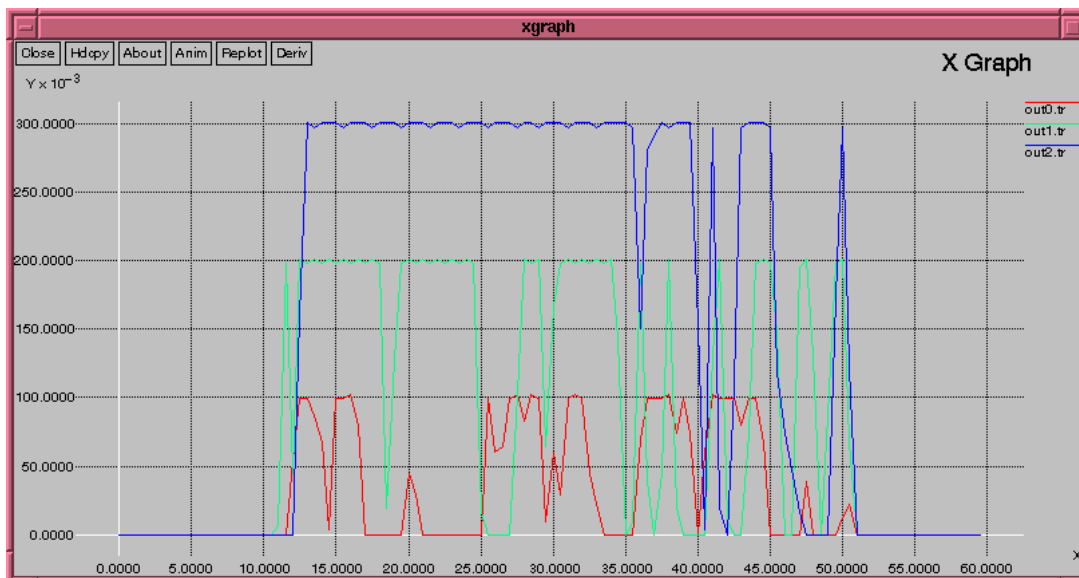


Figura 8 - XGraph

No exemplo da figura 8, criado por [2] pode-se analisar o comportamento de três fluxos em relação ao uso da banda durante um processo de simulação. Caso queira se utilizar outra ferramenta para a produção de gráficos mais sofisticados, pode-se fazê-lo através da análise do *trace* que é um arquivo textual com diversas as informações coletadas durante o processo de simulação.

7. CONCLUSÃO

Diferentes metodologias podem ser exploradas no ensino de redes de computadores. Todavia, a adoção de uma ferramenta que possibilite a construção, modificação ou mesmo a simples manipulação dos elementos de rede, gera uma motivação adicional em meio a um universo de conceitos extremamente abstratos que têm sido prática comum no meio acadêmico. Aliado a isso, o NS tem se mostrado eficiente nas mais variadas vertentes de redes e telecomunicações desde a área de comunicação móvel até as tradicionais LANs, MANs e WANs baseadas em estruturas fixas. Resta aos professores e pesquisadores difundirem o uso desta importante ferramenta de apoio.

8. REFERÊNCIAS

- [1] NS Web Site, <http://www.isi.edu/nsnam/ns/>
- [2] Greis, Mark; Tutorial NS; <http://www.isi.edu/nsnam/ns/tutorial/index.html>
- [3] VINT Project Web Site, <http://netweb.usc.edu/vint/>
- [4] Tanenbaum, Andrew S.; "Computer Networks", Third Edition, Prentice Hall, ISBN: 0-13-349945-6, 1996.
- [5] Jain, Raj; "The Art of Computer Systems Performance Analysis Techniques for Experimental Design, Measurement, Simulation and Modeling"; John Wiley & Sons Inc., ISBN: 0-471-50336-3, 1998. <http://www.cis.ohio-state.edu/~jain>
- [6] Plzak, R.; Wells, A.; Krol, E.; "FYI on Questions and Answers – Answers to Commonly Asked 'New Internet User' Questions"; RFC2664, August 1999.
- [7] IEEE - Institute of Electrical and Electronics Engineers. <http://www.ieee.org>
- [8] IETF - Internet Engineering Task Force. <http://www.ietf.org>
- [9] Projeto Interlegis. <http://www.interlegis.gov.br>
- [10] Croll, A.; Packman, E.; "Managing Bandwidth - Deploying QoS in Enterprise Networks", Prentice Hall, ISBN: 0-13-011391-3, <http://www.phptr.com>, 1999