

UM MODELO DE NEGOCIAÇÃO DE SERVIÇOS COM QoS FIM-A-FIM NA INTERNET

Rodrigo Pedatella, Edmundo Madeira e Maurício Magalhães

Resumo - Considerando a ascendente popularização que a Internet vem acumulando no papel de importante veículo de comunicação, a obtenção de qualidade de serviço (QoS) representa uma necessidade eminente atual. Conseqüentemente, o estudo de soluções que tornem este ambiente cada vez mais adaptado para o tráfego de aplicações que exijam QoS tem recebido grande destaque. Neste contexto, a negociação automática e dinâmica de serviços apresenta grande importância na medida em que certamente contribui para garantir a QoS necessária às aplicações atuais. Este artigo propõe a especificação de um modelo de negociação de serviços para a Internet capaz de garantir a realização de transmissões fim-a-fim com requisitos de QoS.

Palavras-chave: Internet, Qualidade de serviço, DiffServ, MPLS

Abstract - Considering the increasing Internet popularity as an important communication vehicle, the acquirement of Quality of Service (QoS) represents an eminent necessity. Consequently, the study of solutions that can make this environment suitable for the traffic of applications that demand QoS have received great attention. In this context, automatic and dynamic negotiation of services presents great importance since it certainly contributes in acquiring the necessary QoS to the current applications. This paper proposes the specification of a service negotiation model for use in the Internet which is intended to be capable of supporting end-to-end transmissions with QoS requirements.

Keywords: Internet, Quality of Service, DiffServ, MPLS

1. INTRODUÇÃO

Nos últimos anos temos assistido a uma progressiva transformação da Internet em uma ferramenta comprometida com os interesses do mercado corporativo. A popularização das aplicações multimídia tem contribuído em especial para este processo. Entretanto, o modelo tradicional baseado no serviço do melhor esforço tem-se mostrado ineficiente para esse tipo de aplicação, que em geral produz tráfegos os quais exigem garantias de qualidade de serviço (QoS) para que possam apresentar o desempenho e efeito esperados.

Diversas tecnologias têm sido propostas com o objetivo de adaptar a Internet para o oferecimento de serviços em diferentes níveis de qualidade. Dentre as soluções propostas, cer-

tamente merecem destaque o Modelo de Serviços Diferenciados (DiffServ) [10] e a Comutação baseada em Rótulos Multiprotocolar (MPLS) [5]. A primeira se preocupa com a disponibilização escalável de diferentes tipos de serviços com requisitos de QoS particulares. A segunda busca otimizar o uso dos recursos da rede através de técnicas de controle e de roteamento. As pesquisas atuais indicam que estas duas, utilizadas em separado ou devidamente integradas, poderão constituir um padrão para a obtenção de QoS na Internet.

Do ponto de vista gerencial, a disponibilização dos recursos de uma rede em diferentes níveis de qualidade introduz a preocupação com a maneira como tais recursos serão fornecidos aos clientes. Os recursos, devidamente divididos em serviços com características peculiares, exigem a especificação de modelos e/ou mecanismos capazes de promover uma distribuição eficiente, justa e flexível dos mesmos. Neste contexto, a negociação de serviços por meio de contratos tem-se mostrado uma solução adequada à concessão de serviços diversos para clientes de domínios que implementam tecnologias de QoS para a Internet.

Atualmente, a negociação de serviços vem sendo tipicamente realizada de forma estática. Isso significa que existe a ação explícita do administrador da rede na reconfiguração dos equipamentos a cada contrato negociado. Os detalhes são tratados por meios tais como telefone, e-mail e outros métodos que implicam em intervenção manual. Este tipo de procedimento, além de tornar a administração complexa e pouco flexível, ainda apresenta problemas como falta de confiabilidade e aumento do tempo dedicado ao processo administrativo. Além disso, modelos de negociação estáticos não favorecem a escalabilidade pois um grande número de contratos negociados potencialmente agrava os problemas citados.

Por todos esses motivos, a proposta de um modelo de negociação automático e dinâmico tem sido recebida com grande interesse por parte da comunidade científica. Além de simplificar o processo de administração e configuração, a automatização certamente diminuiria o tempo necessário para a negociação. Uma outra vantagem é que métodos de configuração automáticos aumentam a flexibilidade do provedor de serviços em atender as necessidades dinâmicas de seus clientes.

A provisão automática e dinâmica de serviços exige a especificação de um protocolo para a definição e negociação de níveis de serviços [12]. Tal protocolo deve oferecer suporte para diversos tipos de clientes e provedores de acesso, além de garantir a perfeita interação entre eles. Em adição, este protocolo deve privilegiar funcionalidade e confiabilidade para negociações não apenas no interior do domínio, mas também entre domínios distintos.

Como a negociação de serviços é realizada por meio de

Rodrigo Pedatella e Edmundo Madeira são do Instituto de Computação da Unicamp e Maurício Magalhães é da Faculdade de Engenharia Elétrica e de Computação da Unicamp, Campinas, SP, Brasil. E-mails: {rodrigo.pedatella, edmundo}@ic.unicamp.br, mauricio@dca.fee.unicamp.br.

contratos, é necessário criar padrões para diferentes perfis de serviços. A padronização mostra-se adequada na medida em que permite reunir as necessidades gerais de diferentes tipos de serviços. Contudo, um contrato padrão não deve restringir o modelo de negociação a um determinado tipo de protocolo ou tecnologia. Ao mesmo tempo, um protocolo de negociação deve permitir a negociação de diferentes padrões de contratos. Enfim, o cliente deve possuir uma maneira de obter informações sobre os tipos de contratos que seu provedor de acesso está apto a negociar.

A capacidade de negociação entre domínios é outra característica importante e útil a um modelo de negociação. Tal característica tem como função imprescindível garantir a QoS fim-a-fim. Portanto, é bastante desejável que um modelo de negociação contenha, entre seus componentes, um modelo de comunicação entre domínios.

Neste artigo, discutiremos diversos assuntos relacionados à negociação de serviços com QoS na Internet, além de apresentar uma proposta de modelo de negociação. Na sequência, a Seção 2 apresenta diversos protocolos de negociação atualmente existentes. A Seção 3 discorre sobre técnicas para comunicação entre domínios. A Seção 4 comenta detalhes sobre contratos de serviços. A Seção 5 apresenta uma proposta de modelo de negociação. A Seção 6 busca validar o modelo proposto através de uma simulação. Finalmente, a Seção 7 apresenta conclusões e sugestões para trabalhos futuros.

2. PROTOCOLOS DE NEGOCIAÇÃO

Nesta seção apresentaremos os conceitos que embasam os principais protocolos utilizados atualmente para a negociação de serviços na Internet. Alguns deles foram confeccionados especificamente para este objetivo. Outros, apesar de não confeccionados originalmente para este objetivo, podem ser adaptados para tal com certa facilidade. A partir destes protocolos, discutiremos propostas de soluções para os diversos requisitos do modelo de negociação a ser proposto posteriormente.

2.1 O PROTOCOLO COPS

O *Common Open Policy System* (COPS) é um protocolo desenvolvido com o intuito de promover trocas de informações de políticas entre um servidor e seus clientes [7].

O protocolo é baseado no modelo cliente/servidor e possui um ambiente composto dos seguintes elementos:

- **Repositório de Dados:** armazena informações sobre as políticas de negociação.
- **Policy Enforcement Point (PEP):** ponto onde as decisões de configuração são aplicadas. Esta entidade reside no próprio cliente, sendo responsável por configurar os roteadores com base em decisões tomadas pelo PDP.
- **Policy Decision Point (PDP):** residente no gerente, esta entidade é responsável por recuperar regras de políticas do repositório e traduzi-las em parâmetros de configuração para os roteadores do domínio.

O uso do COPS como protocolo de negociação de serviços apresenta algumas motivações interessantes. Em primeiro lugar, o uso do TCP como protocolo de transporte garante confiabilidade na troca de mensagens de sinalização entre cliente(s) e servidor. Existe ainda a possibilidade de utilização de protocolos de segurança como o IPsec e o TLS. Uma outra característica é a manutenção de estados parciais até o término do processo de negociação. Por último, o protocolo permite o transporte de objetos variados sem a necessidade de modificações.

Existem dois modelos de utilização do COPS: delegação e provisionamento. A diferença entre os dois está na maneira com que o PEP obtém a política do PDP. No primeiro, cada evento no PEP leva a uma decisão por parte do PDP indicando a política a ser aplicada ao evento. No segundo, as políticas são instaladas no PEP antes que este receba eventos.

A comunicação entre PDP e PEP ocorre através de trocas de mensagens. A Figura 1 mostra essa interação, ilustrando as principais mensagens do protocolo:

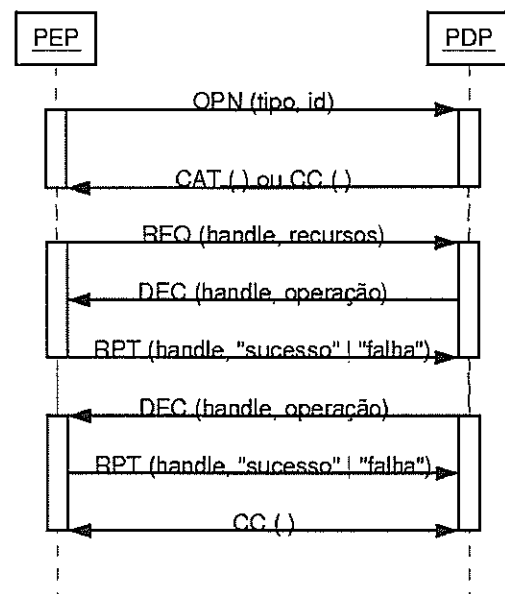


Figura 1. Troca de mensagens no protocolo COPS.

- **OPN:** mensagem enviada pelo PEP ao PDP com o intuito de iniciar uma sessão.
- **CAT:** mensagem enviada pelo PDP ao PEP com o objetivo de informar o segundo sobre a aceitação da proposta de início de sessão.
- **REQ:** mensagem enviada pelo PEP que reúne informações sobre algum tipo de requisição a ser feita ao PDP.
- **DEC:** mensagem de resposta a uma solicitação de requisição feita anteriormente.
- **RPT:** mensagens deste tipo carregam informações sobre sucesso ou falha na instalação de uma decisão. Toda mensagem DEC exige o envio de uma mensagem RPT por parte do PEP.

- **CC:** mensagem responsável por encerrar uma sessão entre PEP e PDP. Pode ser enviada a qualquer momento por ambas as partes.

O COPS originalmente não foi confeccionado para uso em negociações ou reservas de recursos. Contudo, sua adaptabilidade para o transporte de informações de políticas faz com que o protocolo apresente alguns conceitos interessantes para aplicações que tenham este objetivo.

2.2 O PROTOCOLO BGRPP

O *Border Gateway Routing Protocol Plus* (BGRPP) [11] é uma arquitetura originada do BGRP [8] que oferece um protocolo de negociação de recursos baseado na agregação de reservas. As reservas são negociadas entre os chamados agentes BGRP, os quais são posicionados nos roteadores de borda de cada domínio. Deste modo, cada domínio é responsável por realizar o controle de admissão levando em consideração os recursos disponíveis.

A Figura 2 mostra um cenário típico de utilização do BGRPP. O tráfego ingressa/egressa dos domínios através de dois diferentes tipos de roteadores:

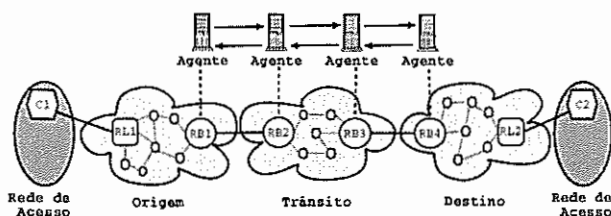


Figura 2. Cenário típico de utilização do protocolo BGRPP.

- **Roteador Limite (RL):** responsável por conectar um domínio a uma rede cliente.
- **Roteador Borda (RB):** conecta um domínio a outro domínio que também faça parte da infra-estrutura BGRPP.

Uma vez efetuado o pedido de reserva, supõe-se que o domínio de origem é capaz de executar controle de admissão levando em consideração os recursos disponíveis localmente. Contudo, ele não tem informações sobre a disponibilidade de recursos ao longo do caminho para o domínio de destino. Desta forma, o domínio de origem deve determinar um roteador borda para a negociação com o respectivo agente BGRPP. Através do protocolo, os agentes são capazes de determinar se os recursos solicitados estão disponíveis em cada domínio do fim-a-fim. Cada agente é responsável pelos recursos de seus vizinhos imediatos. O roteamento entre os domínios é realizado pelo *Border Gateway Protocol* (BGP).

A sinalização do BGRPP é realizada através das seguintes mensagens:

- **PROBE:** disparada por um agente do domínio de origem, esta mensagem é repassada aos agentes de domínios intermediários até o domínio de destino. Ao receber esta mensagem, o agente certifica-se da validade do contrato e atualiza informações de roteamento

na mesma, a fim de que o caminho de volta seja idêntico ao de ida.

- **GRAFT:** indica a disponibilidade de recursos até o domínio de destino. Esta mensagem é enviada em resposta a uma mensagem do tipo PROBE.
- **REFRESH:** contém a indicação de largura de banda necessária no momento. Esta mensagem é enviada por um agente a outro de domínio subsequente a fim de diminuir uma quantidade de recursos solicitada anteriormente. Uma mensagem do tipo REFRESH com valor de banda nulo é utilizada para cancelar a reserva.
- **ERROR:** indica a ocorrência de algum erro. Contém uma descrição do tipo de erro.

A principal característica do BGRPP é a escalabilidade. O protocolo faz uso de *sink trees*¹, conforme descrito pelo BGRP, para promover a agregação de reservas. Desta maneira, a quantidade de informações de estado armazenada pela rede pode ser reduzida. Além disso, o protocolo busca diminuir sobrecargas causadas pelo excesso de mensagens de sinalização através de um mecanismo de resposta prematura a mensagens de reserva. Este mecanismo é denominado *quiet grafting* e possibilita que as mensagens nem sempre tenham que trafegar por todos os domínios do fim-a-fim².

2.3 O PROTOCOLO SIBBS

O *Simple Inter-domain Bandwidth Broker Signalling* (SIBBS) [2] é o protocolo de sinalização proposto pelo grupo QBONE/Internet2 para promover a negociação de recursos em redes DiffServ.

O protocolo sugere que a negociação seja baseada em *Bandwidth Brokers* (BB's) e busca a manutenção da simplicidade inerente ao modelo DiffServ³. Ele define a troca de mensagens de sinalização do tipo requisição/resposta apenas entre BB's de domínios vizinhos imediatos. Tais mensagens carregam as informações essenciais sobre a negociação a ser executada.

Existem dois tipos de mensagem no SIBBS:

- **Resource Allocation Request (RAR):** mensagem de solicitação de recursos. Carrega em seu interior as informações necessárias para a alocação.
- **Resource Allocation Answer (RAA):** representa a resposta a um pedido de alocação feito anteriormente. A resposta pode ser positiva, negativa, ou parcialmente positiva. Neste último caso, a mensagem carrega informações sobre a quantidade de recursos que pode ser

¹*sink trees* são estruturas de dados na forma de árvores que promovem distribuição/compartilhamento de informações e, com isso, ganho de memória.

²O *quiet grafting* é uma estratégia que prevê que domínios intermediários guardem informações sobre recursos disponíveis em domínios vizinhos, dispensando em diversas situações o envio de mensagens de PROBE para todos os domínios do fim-a-fim e com isso criando uma perspectiva de grande diminuição no número de mensagens de sinalização.

³Um *Bandwidth Broker* (BB) representa um *software* com a função de configurar, gerenciar e distribuir os recursos disponíveis em uma rede.

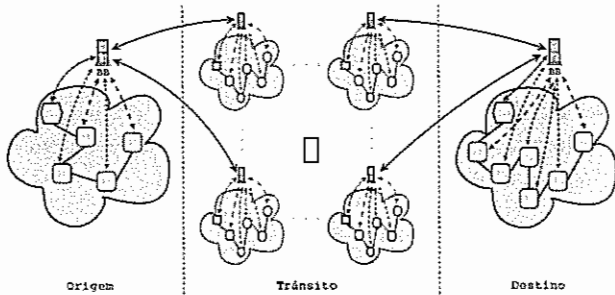


Figura 3. Funcionamento geral do protocolo SIBBS.

disponibilizada (mesmo que não seja o que foi solicitado).

A Figura 3 mostra um exemplo de utilização do protocolo SIBBS. Também para este protocolo a principal preocupação é com a escalabilidade. A idéia é que a negociação exclusivamente entre domínios administrativos vizinhos imediatos consiga diminuir drasticamente o número de mensagens de sinalização. Deste modo, são especificadas apenas as interfaces para comunicação entre tais domínios, deixando-se em aberto detalhes sobre contratos de serviços, assim como os detalhes de como cada um dos domínios gerencia seus recursos. O objetivo é que diferentes implementações de modelos de gerenciamento de recursos e contratos possam ser utilizadas neste mesmo protocolo sem a necessidade de grandes modificações.

3. COMUNICAÇÃO ENTRE DOMÍNIOS

Um modelo de comunicação entre domínios define a maneira como os gerentes se relacionam com o objetivo de negociar serviços para os domínios que representam. Existem atualmente três propostas de modelos de comunicação entre domínios: Modelos Estrela, Hierárquico e Cascata.

O Modelo Estrela caracteriza-se pela negociação centralizada realizada pelo domínio de origem. Neste caso, um gerente negocia contratos de serviços diretamente com os gerentes de domínios com os quais deseja manter relações. A grande vantagem do Modelo Estrela é o controle administrativo que se ganha pela centralização das negociações no domínio solicitante. Por outro lado, este modelo apresenta um grande problema de escalabilidade. Dependendo da topologia entre os domínios de um determinado ambiente, o número de contratos de negociação pode crescer consideravelmente. O modelo Estrela está ilustrado na Figura 4.

O Modelo Hierárquico promove a inserção de uma nova entidade com o objetivo de centralizar as negociações. Esta entidade, denominada *Service Exchange* (SE) [16], passa a ser responsável pela definição e negociação de serviços com base em informações fornecidas pelos gerentes de cada domínio. As informações compreendem serviços oferecidos, recursos disponíveis, topologia e características de enlaces entre domínios. Toda negociação ocorre por meio da SE. Periodicamente, os domínios enviam à SE informações sobre compra e venda de serviços. De posse destes dados, tal entidade realiza "rodadas de negociação". O resultado é a permissão total, a permissão parcial, ou mesmo a negação dos serviços.

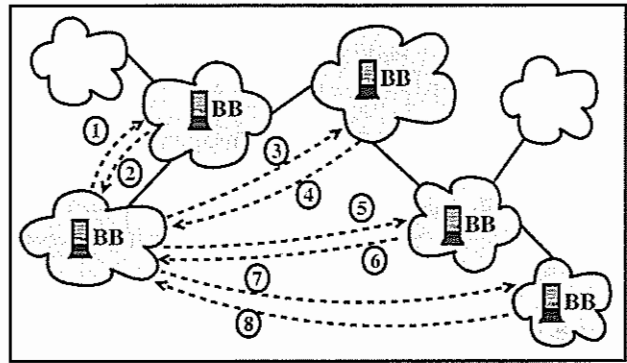


Figura 4. Modelo Estrela.

O Modelo Hierárquico apresenta como vantagens escalabilidade e, principalmente, justiça na alocação de recursos. Como as decisões são tomadas por uma única entidade, pode-se pensar em critérios sofisticados para uma concessão justa de recursos. Em contra-partida, a aplicação deste modelo apresenta problemas de ordem prática. É pouco provável que um provedor de acesso concorde em informar a uma entidade externa detalhes importantes sobre sua organização interna. Menos provável ainda é sua concordância com que essa entidade tome as decisões de alocação de recursos em nome de seu domínio. A Figura 5 ilustra o funcionamento do Modelo Hierárquico.

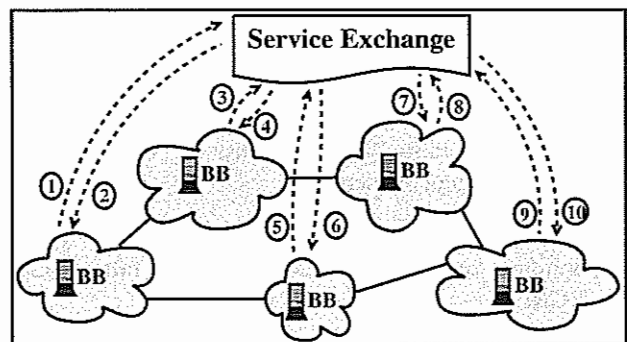


Figura 5. Modelo Hierárquico.

O Modelo Cascata caracteriza-se pela negociação indireta de recursos. Neste modelo, o domínio de origem negocia apenas com domínios adjacentes. Quando é necessário negociar um contrato com um domínio não adjacente, essa tarefa é delegada a um domínio adjacente, que fará o mesmo se ainda não for o vizinho imediato do domínio de destino. Esse processo se repetirá até que o vizinho imediato do domínio de destino seja alcançado quando, neste caso, a negociação ocorre de fato e de maneira coletiva. O Modelo Cascata apresenta como vantagens escalabilidade e simplicidade. A grande crítica ao modelo é a dificuldade de implementação de mecanismos de justiça na alocação de recursos, uma vez que as decisões são descentralizadas. Podemos visualizar um esquema que representa a organização do Modelo Cascata na Figura 6.

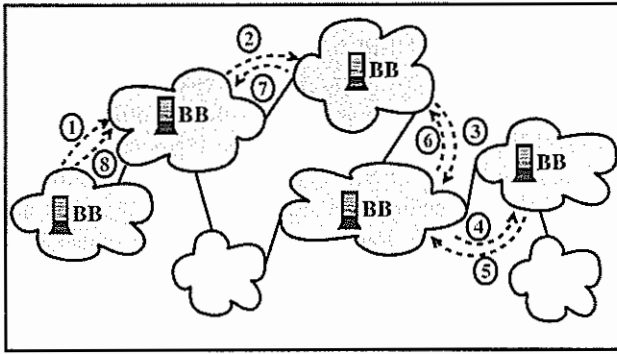


Figura 6. Modelo Cascata.

4. CONTRATOS DE SERVIÇOS

Service Level Specification (SLS) pode ser definido como a representação de um conjunto de parâmetros técnicos os quais descrevem o tratamento a ser dispensado a um determinado fluxo de pacotes em um domínio de transporte, entre as interfaces de ingresso e egresso [14].

Na prática, tais entidades representam contratos realizados entre duas entidades em um modelo de negociação, sendo estas gerente/cliente ou gerente/gerente. Contratos armazenam os detalhes técnicos da negociação.

4.1 REQUISITOS DE UM CONTRATO

O projeto TEQUILA [19] apresenta uma boa proposta sobre as informações que um contrato de serviços deve possuir. A idéia é especificar um contrato de serviços genérico o suficiente para se tornar um padrão. Com base nesse padrão, outros contratos poderiam ser criados para necessidades específicas.

Em [6], os requisitos de um contrato de serviços são divididos em: escopo, identificação de fluxo, parâmetros de conformidade de tráfego, tratamento de excesso, parâmetros de desempenho, disponibilidade e confiabilidade.

A informação de escopo de um SLS especifica a região topológica sobre a qual a qualidade de serviço deve ser garantida. O escopo é representado por informações de identificação dos nós de fim-a-fim: endereços IP ou mesmo qualquer informação capaz de identificar tais nós.

A identificação de fluxo de um SLS indica para quais pacotes o tratamento diferenciado deve ser executado. A informação de identificação é tipicamente representada pelo campo reservado à especificação da classe a que pertence o pacote. Este campo pode estar contido no cabeçalho do pacote IP de uma rede DiffServ ou no rótulo do pacote de uma rede MPLS. Outras informações de identificação são: portas de origem/destino a que se referem os pacotes e versão do protocolo IP.

Os parâmetros de conformidade de tráfego são utilizados para o mecanismo de policiamento do mesmo. São representados por parâmetros que especificam valores referenciais a que um tráfego deve estar subordinado e por um algoritmo de conformidade de tráfego. Os mais utilizados são o algoritmo do Balde Furado e o do Balde de Fichas. Parâmetros para estes algoritmos são: taxa de pico (em bps), taxa de fichas do

balde (em bps), profundidade do balde (em bits), unidades de transferência máxima e mínima (em bits).

O tratamento de excesso descreve como o gerente deve tratar excessos detectados nos mecanismos de policiamento. Existem três possíveis tratamentos: descarte, condicionamento ou remarcação. No caso dos dois últimos, deve haver informações adicionais sobre a operação. Por exemplo, a nova classe de pacotes para casos de remarcação, ou o tamanho do *buffer* dos condicionadores para casos de condicionamento.

Os parâmetros de desempenho descrevem o comportamento desejado, ou seja, as garantias de QoS que o gerente está oferecendo ao cliente. Os parâmetros de desempenho são: atraso (em ms), variação estatística de atraso (em ms), descarte (em percentual) e vazão (em Kbps). Garantias de desempenho podem ser quantitativas ou qualitativas. Se pelo menos um dos parâmetros de desempenho for quantificado, então as garantias de desempenho do contrato são ditas quantitativas. Caso contrário, elas são ditas qualitativas. Parâmetros qualitativos são especificados em três faixas: alta, média ou baixa. Apenas o parâmetro de vazão não pode ser qualitativo.

A disponibilidade indica o período de tempo em que determinado serviço estará acessível. A disponibilidade é representada pelos parâmetros de horários de funcionamento: durante o dia, a semana, o mês e o ano.

Finalmente, os parâmetros de confiabilidade indicam a quantidade máxima de tempo em que o serviço pode não estar disponível, assim como a quantidade máxima de enlaces e de nós que podem falhar simultaneamente.

4.2 REPRESENTAÇÃO E ARMAZENAMENTO DE CONTRATOS

A troca de informações de contratos no contexto da negociação de serviços exige a especificação de padrões para a representação e o armazenamento dos mesmos. Existem atualmente algumas propostas para a execução de ambas as tarefas.

Com relação à representação de informações de contratos, diversos modelos têm sido propostos. As informações contidas em um contrato podem ser representadas por estruturas de dados específicas através de *Policy Information Bases (PIB's)*. Uma PIB deve ser descrita através da *Abstract Syntax Notation One (ANS.1)* para efeitos de padronização. Outras iniciativas, apoiadas pelo IETF, são o *Policy Core Information Model (PCIM)* [4] e o *QoS Policy Information Model (QPIM)* [13]. Tratam-se de modelos de informação para uso específico de políticas e QoS, respectivamente. O PCIM estende o *Class Information Model (CIM)* para a representação de informações de políticas. De modo análogo, o QPIM estende o modelo PCIM para a representação de políticas aplicáveis a mecanismos de QoS.

Todos os modelos citados são baseados no paradigma de orientação a objeto e podem ser conceitualmente descritos como um conjunto de classes. Uma nova versão de modelo pode ser facilmente estendida a partir de um destes para a representação de informações de contratos de serviços.

Com relação ao armazenamento das informações de con-

tratos, existem propostas para diferentes tipos de necessidades. A forma mais simples de se armazenar os dados de contratos é através de arquivos texto. Neste caso, o acesso e a interpretação das informações devem ser feitos por meio de um *parser*. Tal procedimento não possui grandes funcionalidades para o gerenciamento das informações e deve ser utilizado quando o número de contratos é pequeno e a quantidade de informações de cada contrato é reduzida.

Um proposta de solução mais robusta para o armazenamento de informações de contratos é a *Extended Markup Language (XML)*. Através da criação de *tags* personalizadas pode-se facilmente implementar regras para o armazenamento de informações de políticas. Neste caso, também é necessária a utilização de um *parser*. Contudo, trata-se de um *software* já otimizado e específico para esta função. Além disso, a estrutura organizada do XML permite o gerenciamento de uma quantidade bem maior de contratos de serviços com informações diversas.

De qualquer modo, a proposta mais adequada para o armazenamento de contratos segundo o IETF é o *Lightweight Directory Access Protocol (LDAP)*. Diretórios LDAP são bastante apropriados para o armazenamento seguro, com acesso controlado e com alta eficiência para dados compartilhados na Internet [18]. Devido à sua estrutura distribuída, possuem capacidade de gerenciar grande quantidade de informações de forma robusta.

5. PROPOSTA DE MODELO DE NEGOCIAÇÃO

A estratégia utilizada para a confecção do modelo de negociação foi o aproveitamento de idéias e conceitos de protocolos existentes, com a devida adaptação destes para o contexto da negociação dinâmica, automática e transparente entre domínios distintos.

Dividimos a especificação do modelo em quatro partes. Inicialmente trataremos da questão operacional, representada por um protocolo de envio e recebimento de mensagens. A seguir será apresentado o plano gerencial, representado pela estratégia da negociação de atacado/varejo. Logo depois serão apresentados detalhes sobre políticas de configuração e, por último, a especificação de contratos adotados para o modelo.

5.1 O PROTOCOLO SLA-COPS

Para a definição do protocolo de negociação foram aproveitados conceitos encontrados nos protocolos COPS, BGRPP e SIBBS.

Do protocolo COPS foi utilizada toda a arquitetura básica, com as entidades PEP, PDP e o repositório de dados. O conceito de administração baseada em políticas de configuração foi igualmente incorporado ao protocolo, assim como formato e semântica das mensagens de sinalização. O uso de políticas garante à gerência do sistema grande automatização, limitando a configuração essencialmente à especificação das mesmas. Além disso, a idéia de transparência nas informações transportadas pelas mensagens possibilita que contratos de diferentes padrões sejam utilizados, o que garante

adaptabilidade e dinamismo ao novo protocolo.

Com relação ao BGRPP, a principal herança diz respeito a escalabilidade. A estratégia do *quiet grafting* contribui de forma decisiva para a negociação escalável entre diferentes domínios. Deste modo, um esquema bastante semelhante ao *quiet grafting* foi incorporado ao protocolo, recebendo a denominação de **negociação de atacado/varejo**. Por estar bastante vinculado ao nível gerencial do modelo, esta estratégia será detalhada na subseção 5.2.

Para a comunicação entre domínios distintos foi utilizado o Modelo Cascata apresentado na Seção 3. Este modelo é igualmente utilizado pelo protocolo SIBBS, o qual apresenta uma proposta detalhada para sua utilização (ver [2]). O protocolo aqui especificado se valeu desta especificação para implementar a comunicação entre domínios e viabilizar QoS fim-a-fim para o modelo de negociação.

Como toda a infra-estrutura do protocolo foi retirada do COPS, iremos denominá-lo **SLA-COPS**. As entidades PEP e PDP serão denominadas, respectivamente, **SLA-PEP** e **SLA-PDP**. A operação do protocolo envolve estas duas entidades e é dividida em três fases:

- **Fase de Inicialização:** o SLA-PEP, após receber pedido do cliente, envia ao SLA-PDP uma mensagem de abertura de negociação (OPN). Este último verifica a validade de tal requisição e, se não houver empecilhos (políticas restritivas), envia uma mensagem de aceitação de cliente (CAT) em resposta. Caso contrário, envia uma mensagem de fechamento de cliente (CC) e encerra o processo de negociação.
- **Fase de Configuração:** caso tenha sido aceito, o SLA-PEP envia nova mensagem ao SLA-PDP, com o intuito de negociar os parâmetros modo de negociação⁴, tempo para renegociação e contratos SLS predefinidos. O SLA-PDP analisa tais parâmetros, realiza as modificações necessárias nos mesmos e envia uma mensagem do tipo DEC à entidade SLA-PEP com os novos valores para os parâmetros analisados. O SLA-PEP envia ao SLA-PDP uma mensagem do tipo RPT para confirmar o recebimento de DEC.
- **Fase de Negociação:** o SLA-PEP envia ao SLA-PDP uma nova mensagem do tipo REQ, desta vez salientando os recursos que o cliente deseja obter. O SLA-PDP analisa o pedido do SLA-PEP, aplica as políticas relacionadas àquele cliente e envia uma resposta ao SLA-PEP, numa mensagem do tipo DEC, frisando o que pode ser oferecido a partir do que foi pedido (concessão total, parcial ou negação da solicitação). O SLA-PEP analisa a resposta do SLA-PDP, comparando os resultados obtidos com o que foi pedido e, finalmente, envia uma mensagem do tipo RPT ao SLA-PDP informando-o se o cliente concorda ou não com os termos propostos para a negociação.

Efetuada as três etapas satisfatoriamente, o processo de negociação está finalizado. SLA-PDP ou SLA-PEP podem,

⁴Modos de negociação são: predefinido (gerente restringe parâmetros do SLS) e não predefinido (o próprio cliente restringe parâmetros do SLS).

a qualquer momento, cancelar o contrato enviando ao outro uma mensagem do tipo CC. As penalidades para estes procedimentos podem ser negociadas em contrato.

É importante destacar que o protocolo proposto apresenta um princípio funcional diferente da arquitetura de políticas em três camadas tradicional utilizada pelo COPS. O SLA-PEP não representa um aplicador de políticas nos elementos de rede, mas sim um intermediário que pode ser encarado como um "configurador de contratos" para o gerente de negociação. A partir dos contratos negociados, a idéia é que o gerente de negociação do domínio solicitante reconfigure seus roteadores, utilizando para tanto um PEP tradicional.

5.2 NEGOCIAÇÃO DE ATACADO/VAREJO

A estratégia de negociação proposta para este trabalho é denominada **negociação de atacado/varejo**. Esta estratégia visa garantir o provisionamento escalável de QoS especialmente para casos em que os nós de fim-a-fim pertençam a domínios distintos.

A estratégia de negociação diz respeito ao plano gerencial do modelo e está organizada da seguinte maneira: para obter recursos de outros domínios, cada gerente negocia contratos apenas com seus vizinhos imediatos (segundo o Modelo Cascata), pedindo a estes recursos de todos os domínios para os quais deseja ter garantias de QoS. Em contra-partida, cada gerente deve garantir a outros domínios uma quantidade de recursos adequada, a fim de que consiga atender satisfatoriamente a propostas de contratos de seus vizinhos imediatos. Busca-se com isso a montagem de "túneis" de QoS entre os domínios e uma substancial redução do número de mensagens de sinalização. Contratos desta natureza compõem a chamada **negociação de atacado**, uma vez que um único contrato simboliza a necessidade de vários clientes.

De posse de recursos provenientes de outros domínios garantidos pelos contratos de atacado, cada gerente negocia contratos locais, oferecendo a seus clientes, além de recursos próprios, os recursos obtidos na negociação de atacado. Estes contratos locais compõem a chamada **negociação de varejo**.

Cabe a cada domínio organizar coerentemente seus contratos e recursos. Enfim, é necessário um bom relacionamento entre os domínios. Eles devem se organizar com base em mecanismos de histórico de dados e perfis de tráfego.

É desejável que os contratos negociados entre os domínios apresentem uma grande quantidade de recursos, a fim de que a renegociação seja feita com uma frequência mínima. Deste modo, a duração dos contratos entre domínios tende a ser grande, apesar disso não ser uma regra.

Uma característica importante dessa estratégia de negociação é que, apesar de contratos de atacado representarem pedidos de diversos clientes, o domínio solicitado só necessita conhecer o domínio solicitante. Cabe a este último dividir os recursos obtidos entre os seus clientes. A tarefa de cada domínio é, portanto, reunir os recursos de que necessita (interna e externamente) a fim de poder realizar inteligentemente a negociação de atacado.

5.3 POLÍTICAS DE NEGOCIAÇÃO

A implementação do modelo de negociação proposto

disponibiliza minimamente as seguintes políticas de negociação de recursos:

- **Políticas de admissão:** definem os clientes autorizados a negociar com o domínio. Caso um cliente não esteja presente na lista, o protocolo de negociação o rejeita logo no início da negociação.
- **Políticas de restrição de banda:** restringem a quantidade de recursos (total e por domínio) que pode ser solicitada por cliente e por serviço. Se ausentes para um determinado cliente, significa que ele não possui qualquer restrição de banda (a não ser a capacidade do domínio).
- **Políticas de renegociação:** definem o tempo mínimo para renegociação de um contrato. Objetiva diminuir o número de contextos de negociação. Se ausentes para um determinado cliente, significa que ele é livre para renegociar o contrato a qualquer momento.
- **Políticas de aceitação de contratos:** definem o que um cliente está disposto a aceitar em termos de uma contra-proposta feita pelo domínio. Existem três possíveis comportamentos: (1) o cliente aceita qualquer contra-proposta; (2) o cliente aceita apenas o que foi originalmente solicitado; (3) o cliente aceita condicionalmente uma contra-proposta. Neste último caso, é necessário especificar a quantidade mínima de recursos que serão aceitos.

As três primeiras categorias de políticas fazem parte da configuração do provedor do(s) serviço(s) (domínio solicitado). A última categoria faz parte da configuração do cliente: sempre que estipular um contrato, ele deve escolher o comportamento desejado a partir do recebimento de uma contra-proposta.

Estas políticas são suficientes para atender as necessidades do modelo de negociação. Contudo, se para algum caso específico for preciso novas políticas, a idéia é que elas possam ser implementadas na forma de *plugins* e carregadas dinamicamente ao *software* que implementa o modelo.

5.4 ESPECIFICAÇÃO DE CONTRATO

O contrato proposto para o modelo de negociação foi especificado a partir do levantamento de requisitos apresentado na Seção 4. Tais requisitos foram divididos em unidades lógicas que reúnem parâmetros semanticamente relacionados. Foram especificadas as seguintes unidades lógicas:

- **Unidade Comum:** reúne informações de identificação e controle do contrato, como especificação de serviço(s) negociado(s), identificação do provedor, identificação do cliente, tipo de cliente, data de criação do contrato e data de validade.
- **Unidade de Topologia:** reúne informações sobre os nós envolvidos na negociação. O cliente deve preencher as informações de nós de origem e o provedor deve preencher as de nós de destino. Podem ser especificados diversos pares de nós de origem/destino. Cada par

é transformado por esta unidade em uma estrutura que representa um grafo parcial e recebe uma identificação. Posteriormente, cláusulas do contrato irão referenciar estas estruturas.

- **Unidade de QoS:** representa as cláusulas de um contrato. Existe uma unidade de QoS para cada cláusula do contrato. É formada por diversas subunidades. A subunidade de Escopo carrega informações sobre a(s) origem(ns) e o(s) destino(s) estipulados para a cláusula. A subunidade de Descritor de Tráfego possui informações sobre código de serviço, portas de origem/destino e protocolo de rede utilizado. A subunidade de Descritor de Carga armazena informações sobre parâmetros de condicionamento e policiamento de tráfego, além do algoritmo de condicionamento a ser utilizado. A subunidade de Parâmetros de QoS possui as informações de atraso, variação de atraso e descarte máximos permitidos para a cláusula do contrato. A subunidade de Disponibilidade de Serviços armazena informações sobre horários de disponibilidade para a cláusula. Finalmente, por uma questão de simplicidade, a largura de banda garantida para a cláusula é especificada no final da mesma.

Cada contrato possui a estrutura completa descrita acima. Alguns dados devem ser especificados pelo cliente. Outros, exclusivamente pelo provedor de serviços. A decisão sobre parâmetros de responsabilidade do cliente está intimamente ligada ao modo de negociação especificado na fase de configuração do protocolo SLA-COPS.

5.5 IMPLEMENTAÇÃO

A implementação do modelo de negociação foi executada como uma extensão às funcionalidades do *framework* confeccionado em [3] e [15]. De [3] foi utilizado o *software* de BB, com os gerentes de recursos e de políticas, bem como o componente de aplicação das políticas nos roteadores. Já de [15] foi utilizada uma plataforma adaptada ao uso conjunto das tecnologias MPLS e DiffServ, além das modificações necessárias ao BB para suporte a esta plataforma.

O *software* do BB proposto em [3] foi implementado em linguagem de programação Tcl, com uso das extensões iTcl e Combat para suporte, respectivamente, a orientação a objetos e CORBA. O mapeamento de CORBA para iTcl, realizado pelo pacote Combat, é obtido através das interfaces *Dynamic Invocation Interface* (DII) e *Dynamic Skeleton Interface* (DSI). Por uma questão de compatibilidade e reaproveitamento de código, a linguagem Tcl e ferramentas citadas foram mantidas para a implementação do modelo de negociação. A Figura 7 ilustra os componentes do BB e o relacionamento entre eles.

A plataforma MPLS-DS apresentada em [15] foi desenvolvida sobre o simulador de redes *Network Simulator* (NS) [20], com o uso das extensões DS-NORTEL [9] e MNS [1] para suporte a DiffServ e MPLS. Ela tem como objetivo validar, por meio de simulações, o uso de políticas de configuração para a obtenção de QoS fim-a-fim em redes IP. A opção pela simulação proporciona grande liberdade na criação de cenários de utilização.

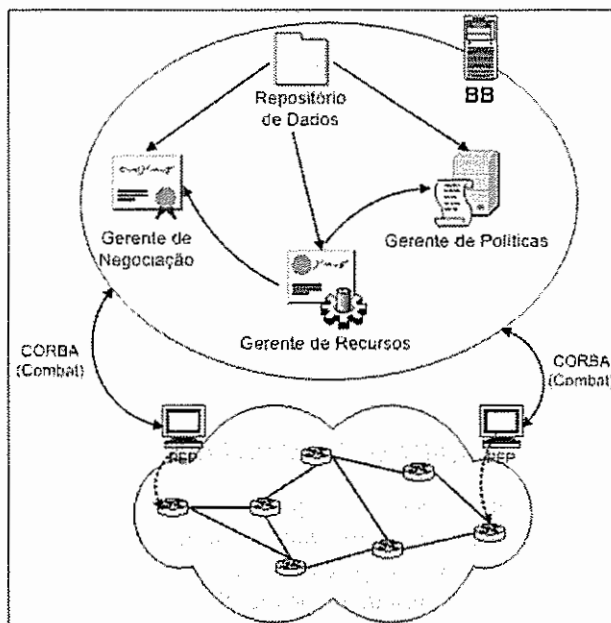


Figura 7. Implementação do BB

O BB é originalmente composto por três módulos estruturais: o primeiro implementa as funcionalidades de um gerente de recursos; o segundo implementa um gerente de políticas; o terceiro módulo implementa as funcionalidades previstas para os pontos de aplicação de políticas, vinculados aos dispositivos de nível físico. A estes componentes foi integrado um quarto módulo, denominado Gerente de Negociação e implementado segundo o modelo proposto neste artigo.

6. VALIDAÇÃO DO MODELO

A fim de demonstrar as funcionalidades do modelo proposto, apresentamos um estudo de caso para obtenção de QoS fim-a-fim em um ambiente que inclui 4 domínios administrativos.

A Figura 8 mostra a topologia das redes simuladas. Nesta figura, cada círculo representa um roteador MPLS/Diffserv. Os hexágonos representam os clientes. Todo enlace da topologia mostrada possui capacidade total de 3.456 Kbps. Os roteadores MPLS/DiffServ possuem 4 filas para cada enlace com o qual possuem ligação. As filas são capazes de armazenar até 1000 pacotes, independente de seu tamanho. O BB do domínio A (BB-A) atua sobre os roteadores R01 a R06, o do domínio B (BB-B) atua sobre os roteadores R07 a R10, o do domínio C (BB-C) atua sobre os roteadores R11 a R14 e o domínio D (BB-D) sobre R15 a R20.

Foram realizadas medições de atraso e descarte médios com base em acumuladores e medidores disponibilizados pelo NS. O descarte de um LSP representa a média de descartes de todos os roteadores que o compõem. De maneira análoga, o atraso de um LSP é calculado com base no atraso médio dos pacotes no percurso realizado entre pares de roteadores do LSP. Estas medições foram realizadas com frequência de 0,1 segundo.

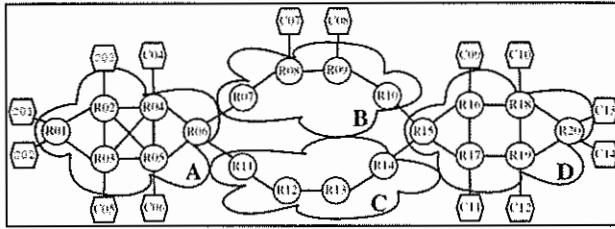


Figura 8. Topologia da simulação

6.1 TRÁFEGO GERADO

Foram selecionados dois tipos de tráfegos para a simulação. O primeiro incorpora as características de tráfego de voz, como taxa de geração constante e tamanho de pacote reduzido. O segundo tipo simula as características de uma aplicação de tráfego exponencial. A Tabela 1 mostra detalhes sobre os tipos de tráfego.

Cada cliente da topologia mostrada gera um dentre os dois tipos de tráfego. O cliente C01 gera tráfego de voz para o cliente C13 e o cliente C02 gera tráfego exponencial para o cliente C14. Os demais clientes geram apenas tráfego local. Os clientes C03, C04, C05 e C06 trocam tráfego exponencial. Já os clientes C07 e C08 trocam tráfego de voz. Os clientes C09, C10, C11 e C12 também trocam informações por meio de tráfego de voz.

Tipo de tráfego	Tamanho do pacote	Taxa
Exponencial	512 bytes	432 Kbps
Voz	66 bytes	66 bytes/20ms

Tabela 1. Características dos tráfegos gerados

6.2 CLASSES DE SERVIÇO

Foram estipuladas três classes de serviços concordadas entre os domínios: *Best Effort* (BE), *Assured Forwarding* (AF) e *Expedited Forwarding* (EF).

A classe AF oferece baixo descarte e alta probabilidade de entrega de pacotes. É adequada ao tráfego de aplicações que exijam requisitos de segurança. A classe EF apresenta prioridade de tratamento sobre as demais classes. Suas características são baixos atraso, descarte e variação de atraso. A classe BE acomoda o tráfego que não precisa receber tratamento diferenciado. A Tabela 2 reúne informações sobre atraso e descarte máximo para cada classe.

Classe	Atraso	Descarte
<i>Best Effort</i> (BF)	-	-
<i>Assured Forwarding</i> (AF)	4 s	20%
<i>Expedited Forwarding</i> (EF)	210 ms	20%

Tabela 2. Características das classes de serviços

As classes de serviço foram simuladas através de 4 filas do tipo *Droptail*, utilizando o escalonador *Weighted Round Robin* (WRR). São três filas físicas, sendo que uma dessas filas é dividida logicamente em duas filas virtuais. Cada classe

de serviço ocupa uma fila. A classe AF ocupa as duas filas virtuais.

No início da simulação, todo tráfego pertence à classe BE e recebe o mesmo tratamento, independente de tipo ou origem. Durante a simulação ocorrem os processos de negociação, levando a um tratamento diferenciado dos tráfegos referenciados em contratos.

6.3 POLÍTICA DE ATUAÇÃO

Foi definida uma única política de atuação para a simulação. Sempre que acionada, esta política é automaticamente traduzida pelo BB correspondente em parâmetros de configuração para os roteadores do domínio.

A política se refere à atualização do peso das filas dos roteadores para uso do WRR. Esta política é acionada sempre que um determinado fluxo de dados sofre descarte maior que o estipulado para a sua classe. No início da simulação, as filas de todas as classes possuem o mesmo peso.

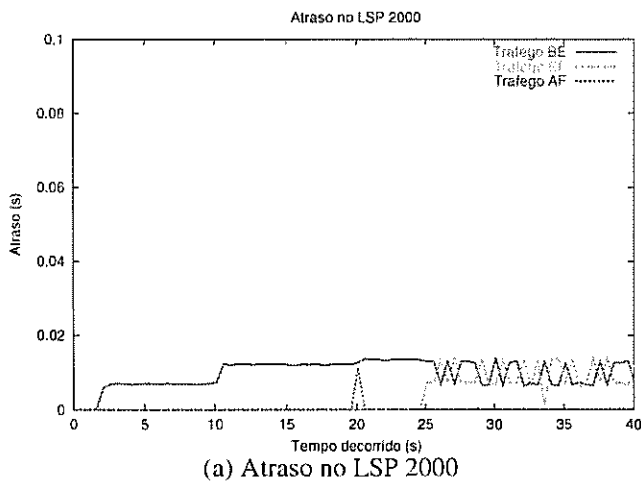
6.4 A SIMULAÇÃO

Quatro LSP's pré-configurados são criados no início da simulação. O LSP 2000 é criado pelo BB-A e segue a rota R01-R02-R04-R06. O LSP 2001 é criado pelo BB-B, seguindo a única rota possível: R07-R08-R09-R10. Similarmente, o BB-C cria o LSP 2002, que segue a rota R11-R12-R13-R14. Finalmente, o BB-D cria o LSP 2003, que segue a rota R15-R16-R18-R20.

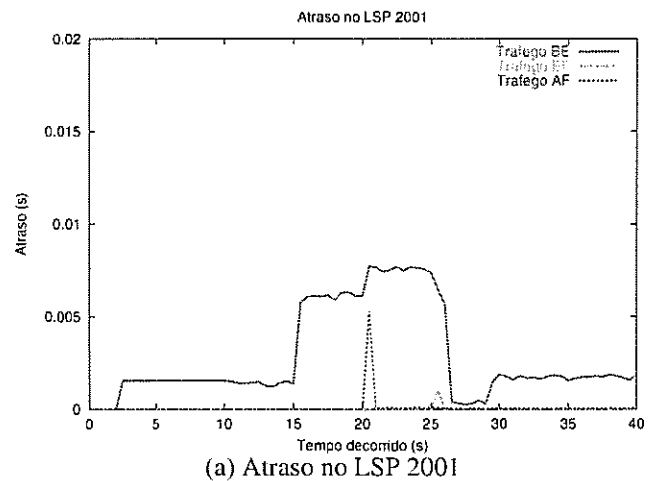
Em $t = 1$ s, o BB-B negocia satisfatoriamente um contrato com o BB-D. Ele negocia 1780 Kbps para fluxos de tráfego da classe AF. No instante $t = 1,5$ s, é a vez do BB-C negociar um contrato com o BB-D. Neste contrato, ele assegura 890 Kbps para fluxos da classe EF com destino ao domínio D. Ainda no instante $t = 1,5$ s, inicia-se o tráfego dos clientes C01 a C13 e C02 a C14. Aos 2 segundos da simulação, o BB-A negocia um contrato com o domínio B. Este contrato prevê 1000 Kbps para a classe AF com destino ao domínio D. Finalmente, o ciclo de negociações de atacado se encerra aos 2,5 s de simulação, quando o BB-A negocia 600 kbps de serviço EF para o domínio D com o BB-C.

Aos 10 segundos, inicia-se o tráfego exponencial entre os clientes C03, C04, C05 e C06. Em $t = 15$ s, inicia-se o tráfego de voz entre os clientes C07 e C08. Os domínios A e B começam a ter seus enlaces saturados e os clientes C01 e C02 vêem seus tráfegos ameaçados de atraso e descarte. No instante $t = 20$ s, o cliente C02 decide realizar uma negociação com o BB-A. Ele negocia 500 Kbps de fluxos AF com destino a C14. O tráfego de C02 passa pelo domínio B no caminho a C14. Em $t = 20,1$ s o BB-A é informado sobre excesso de descarte de tráfego AF nos roteadores do LSP 2000 (Figura 8-b). Neste momento, ele executa a política de reconfiguração das filas referentes ao tráfego AF nos roteadores do LSP 2000. O descarte para este tráfego volta, então, a índices aceitáveis.

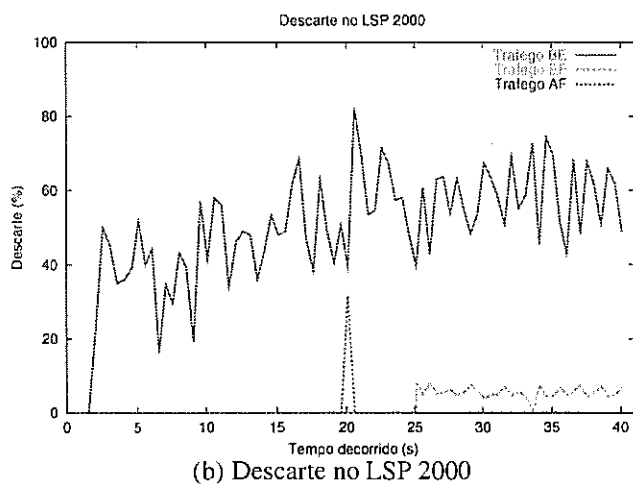
Em $t = 20,5$ s, o BB-B é notificado sobre um descarte de 22,72% em seu tráfego AF, ou seja, o tráfego de C02. Para honrar o contrato de atacado realizado com o BB-A, o primeiro executa a política de reconfiguração de filas para a



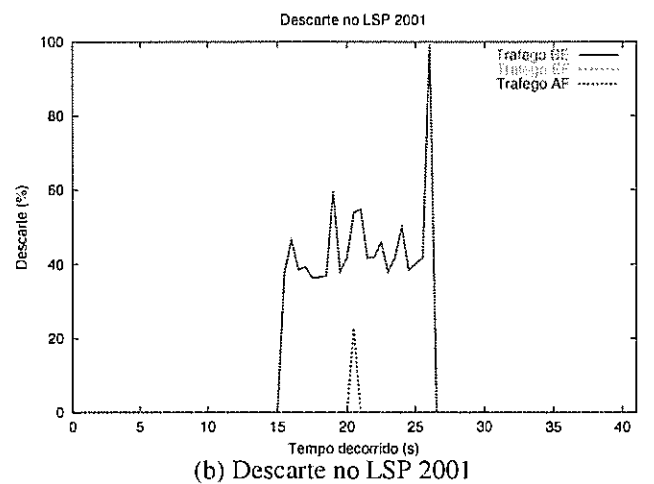
(a) Atraso no LSP 2000



(a) Atraso no LSP 2001



(b) Descarte no LSP 2000



(b) Descarte no LSP 2001

Figura 9. Gráficos do LSP 2000.

Figura 10. Gráficos do LSP 2001.

classe AF, o que faz com que o descarte neste domínio volte a valores compatíveis com os contratos existentes (Figura 9-b).

Aos 25 segundos de simulação, o cliente C01 negocia com o BB-A 480 Kbps de fluxos EF para C13. O tráfego de C01 é desviado para o domínio C no caminho até o destino. Como o domínio C estava ocioso até o momento, o grande descarte que vinha ocorrendo no tráfego de C01 cessa imediatamente, restabelecendo para este cliente a qualidade de serviço contratada.

Quando a simulação alcança os 30 segundos, é iniciado o tráfego de voz entre os clientes C09, C10, C11 e C12. É a vez do domínio D ter seus enlaces saturados (Figura 10-b). Em $t = 30,5$ s, o tráfego EF do LSP 2003 sofre descarte de 24,79%. Em $t = 31$ s, o tráfego AF sofre descarte de 31,81%. O BB-D executa, então, a política de reconfiguração de filas para ambas as classes. Essas duas medidas são suficientes para trazer de volta a estabilidade ao domínio D. Em $t = 40$ s, a simulação é finalizada.

O gráfico da Figura 12 confirma o sucesso da simulação ao apresentar valores de atraso reduzidos para os monitores de fim-a-fim durante todo o tempo. Estes resultados se mostram compatíveis com os atrasos parciais apresentados nas Figuras 8-a, 9-a e 10-a. Para o LSP 2002, os parâmetros de QoS permaneceram anulados durante toda a simulação. Este era o comportamento esperado, uma vez que a capacidade dos

enlaces e das filas dos roteadores superava consideravelmente a necessidade real do tráfego transportado neste LSP.

Um outro fato que merece ser notado é a estabilidade alcançada pelo sistema através da aplicação das políticas. Parâmetros que extrapolaram os contratos foram em todos os casos rapidamente corrigidos pela aplicação das mesmas.

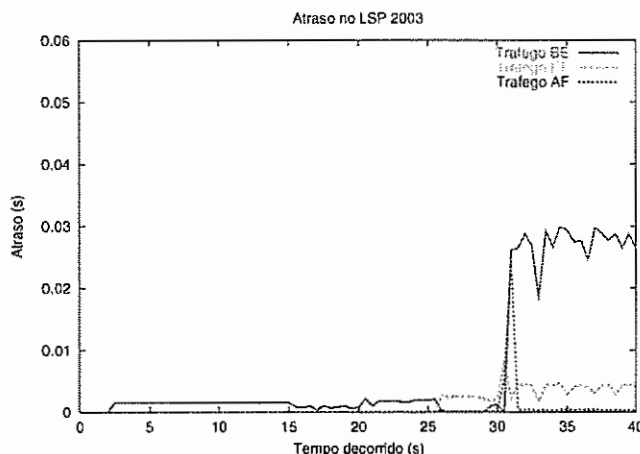
7. CONCLUSÕES

Apresentamos neste artigo um modelo de negociação de serviços que propõe substituir as negociações estáticas, atualmente praticadas na maioria dos sistemas que oferecem algum tipo de QoS, numa busca por automação e dinamismo.

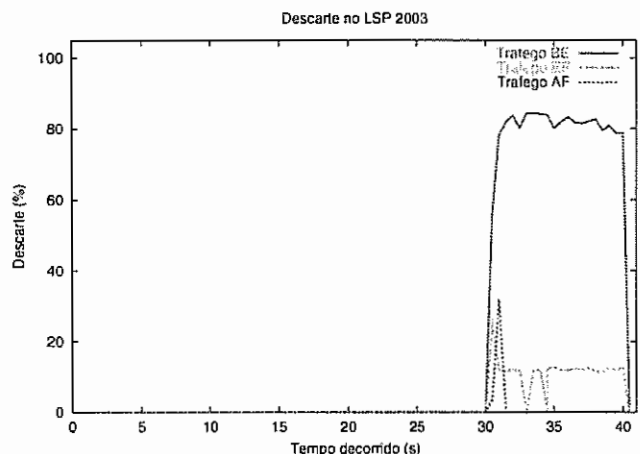
O modelo de negociação proposto é formado, em nível operacional, por um protocolo baseado em políticas de configuração e adaptado ao transporte de contratos heterogêneos. O uso de políticas certamente contribui para a obtenção da automação e do dinamismo almejados. As tarefas administrativas ficam restritas à especificação de políticas, as quais definem o comportamento esperado do sistema quando do recebimento de propostas de contratos. De posse das políticas especificadas, o sistema é capaz de responder automaticamente a cada uma delas. Além disso, a infra-estrutura baseada no protocolo COPS possibilita o tratamento transparente de contratos, fato que confere portabilidade ao modelo.

A ênfase mais imediata deste trabalho foi a de identificar os cenários de políticas que fazem sentido para este tipo de problema e aprimorar o contexto da negociação de atacado/varejo. Desta forma, a simulação serve como um cenário inicial e típico, ainda que simples, para validar estes conceitos. O cenário é típico porque simula o comportamento desejado de alguns tráfegos largamente encontrados nas redes atuais. Uma variação do cenário apresentado neste artigo pode ser encontrado em [17].

Trabalhos futuros certamente compreendem testes em cenários mais complexos de simulação e de ambiente real. Neste sentido, o uso de CORBA oferece um certo grau de transparência do ambiente de simulação. Isso facilita, em tese, que no futuro as técnicas desenvolvidas possam ser testadas em ambientes de rede reais.



(a) Atraso no LSP 2003



(b) Descarte no LSP 2003

Figura 11. Gráficos do LSP 2003.

REFERÊNCIAS

[1] G. Ahn and W. Chun. Overview of MPLS Network Simulator: Design and Implementation. <http://flower.ce.cnu.ac.kr/fog1/mns/index.htm>, 1999.
[2] QBone Bandwidth Broker Architecture. <http://www.internet2.edu/qos/wg/documents-informational/20020709-chimento-et-al-qbone-signaling/>. 2001.
[3] L. Costa. Desenvolvimento de um Bandwidth Broker para a arquitetura DiffServ. Dissertação de mestrado, Unicamp, Março 2001.

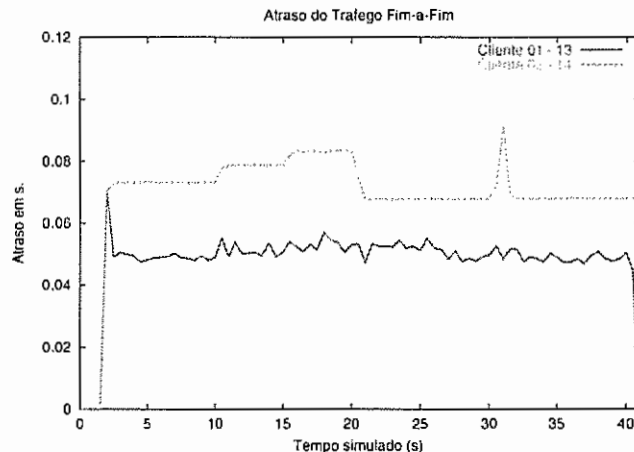


Figura 12. Gráfico de atraso fim-a-fim.

[4] B. Moore et al. Policy Core Information Model - Version 1 Specification. Internet RFC 3060, IETF, Fevereiro 2001.
[5] D. Awduche et al. Requirements for Traffic Engineering over MPLS. Internet draft, IETF, draft-ietf-mpls-traffic-eng.00.txt, Outubro 1998.
[6] D. Goderis et al. Service Level Specification Semantics and Parameters. Internet draft, IETF, draft-tequila-sls-00.txt, Novembro 2000.
[7] J. Boyle et al. The COPS (Common Open Policy Service) Protocol. Internet RFC 2748, IETF, Janeiro 2000.
[8] P. Pan et al. BGRP: Sink-Tree-Based Aggregation for Inter-Domain Reservations. *Journal of Communications and Networks*, Junho 2000.
[9] P. Piedad et al. A Network Simulator Differentiated Services Implementation. Internet draft, Nortel Networks, draft-tequila-sls-00.txt, Julho 2000.
[10] S. Blake et al. An Architecture for Differentiated Services. Internet RFC 2475, IETF, Dezembro 1998.
[11] S. Salsano et al. Inter-domain QoS Signaling: the BGRP Plus Architecture. Internet draft, IETF, draft-salsano-bgrpp-arch-00.txt, Maio 2002.
[12] T. Nguyen et al. COPS-SLS: A Service Level Negotiation Protocol for the Internet. *IEEE Communications Magazine*, Maio 2002.
[13] Y. Snir et al. Policy Framework QoS Information Model. Internet draft, IETF, draft-ietf-policy-qos-info-model-01.txt, Abril 2000.
[14] Y. T. Joens et al. Service Level Specification and Usage Framework. Internet draft, IETF, draft-manyfolks-sls-framework-00.txt, Outubro 2000.
[15] A. M. S. Guillén. MPLS-DS: Uma plataforma para validação de políticas no contexto das redes MPLS/DiffServ. Dissertação de mestrado, Unicamp, Dezembro 2001.
[16] C. Kamiński and D. Sadok. Chameleon: uma Arquitetura para Serviços Avançados Fim a Fim na Internet com QoS. *Simpósio Brasileiro de Redes de Computadores (SBRC 2001)*, Maio 2001.
[17] R. Pedatella, E. Madeira, and M. Magalhães. Um Framework para Obtenção de QoS Fim-a-Fim na Internet. *Simpósio Brasileiro de Redes de Computadores (SBRC 2003)*, Maio 2003.
[18] M. A. Siqueira. Uma Arquitetura de Políticas Para Gerência de Redes MPLS. Dissertação de mestrado, Unicamp, Abril 2002.
[19] TEQUILA Project. <http://www.ist-tequila.org>. 2000.
[20] The Network Simulator. <http://www.isi.edu/nsnam/ns>. 2001.

Rodrigo Pedatella, Edmundo Madeira e Maurício Magalhães **Um Modelo de Negociação de Serviços com QoS Fim-a-Fim na Internet**

Rodrigo Neiva Pedatella é Engenheiro de Software pela empresa Aker Security Solutions. Graduou-se Bacharel em Ciência da Computação pela Universidade Federal de Goiás em Dezembro de 2000. Atualmente está finalizando o curso de Mestrado em Ciência da Computação pela Universidade Estadual de Campinas (UNICAMP). Suas áreas de interesse incluem: qualidade de serviço na Internet, segurança em redes de computadores e desenvolvimento de sistemas distribuídos.

Edmundo Roberto Mauro Madeira é Professor Associado do Instituto de Computação da Universidade Estadual de Campinas (UNICAMP). Graduou-se em Engenharia Civil em 1980, obteve o grau de mestre em Ciência da Computação em 1985 e o grau de doutor em Engenharia Elétrica em 1991, sempre pela UNICAMP. Foi o coordenador do Comitê de Programa do IEEE LANOMS'03. Suas áreas de interesse incluem: oferta de Qualidade de Serviço na Internet, desenvolvimento de Plataformas Distribuídas e Tolerância a Falhas.

Mauricio Ferreira Magalhães é Professor Titular da Faculdade de Engenharia Elétrica e de Computação (FEEC) da Universidade Estadual de Campinas (UNICAMP). Graduou-se em Engenharia Elétrica em 1975 pela Universidade de Brasília (UnB), obteve o grau de mestre em Engenharia Elétrica em 1979 pela Unicamp e o título de Doutor Engenheiro pelo Instituto Politécnico de Grenoble (INPG) em 1983. Suas áreas de interesse incluem: Engenharia de Tráfego e Qualidade de Serviço na Internet, Gerência de Redes e Sistemas de Tempo Real.