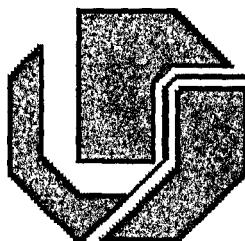


UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



**Um Estudo Sobre a Obtenção de Qualidade de
Serviço Fim-a-Fim em Redes IP com DiffServ e
Engenharia de Tráfego Baseada em MPLS**

Antonio Carlos de Oliveira Júnior

Dezembro

2003

MOV
691.3
04825
TES/MEM

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA ELÉTRICA
PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**Um Estudo Sobre a Obtenção de Qualidade de
Serviço Fim-a-Fim em Redes IP com DiffServ e
Engenharia de Tráfego Baseada em MPLS**

Antonio Carlos de Oliveira Júnior

Dissertação apresentada por Antonio Carlos de Oliveira Júnior à Universidade Federal
de Uberlândia para a obtenção do título de mestre em Ciências aprovada em 19/12/2003
pela Banca Examinadora.

Prof. Anilton Salles Garcia, Dr (UFES)

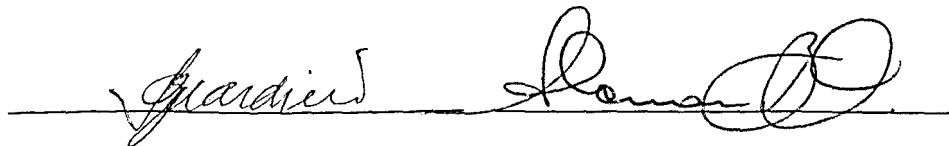
Prof. Gilberto Arantes Carrijo, Dr (UFU)

Prof. Paulo Roberto Guardieiro, Dr - Orientador (UFU)

**Um Estudo Sobre a Obtenção de Qualidade de
Serviço Fim-a-Fim em Redes IP com DiffServ e
Engenharia de Tráfego Baseada em MPLS**

Antonio Carlos de Oliveira Júnior

Dissertação apresentada por Antonio Carlos de Oliveira Júnior à Universidade Federal de Uberlândia como parte dos requisitos para obtenção do título de Mestre em Ciências.



Prof. Dr. Paulo Roberto Guardieiro

Orientador

Prof. Dr. Alcimar Barbosa Soares

Coordenador do curso de Pós-Graduação

DEDICATÓRIA

Aos meus grandes pais, Antonio Só e Adenildes,
aos meus irmãos Thiago e Aline,
pelo apoio, compreensão, paciência e amor,
demonstrados em todos os momentos de minha vida.

*"A grande arma
é o conhecimento e a compreensão."*

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus por todas as graças recebidas ao longo de minha vida.

À Faculdade de Engenharia Elétrica da Universidade Federal de Uberlândia pelos recursos oferecidos para a execução deste trabalho.

Ao meu orientador Prof. Dr. Paulo Roberto Guardieiro pela orientação competente e por todo apoio recebido.

À CAPES Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior pela ajuda financeira recebida no decurso do meu trabalho de pós-graduação.

Aos colegas e amigos do Laboratório de Redes de Computadores, Johann, Clidenor, Daniela, Eliane, Haroldo Zattar, Leandra, Solange, Érica, Karine, Liliana, Rodrigo e Neumar pelo companheirismo, paciência e amizade a mim dedicados.

Ao amigo Roberto Finzi e sua mãe Fátima pelo incentivo e apoio na minha jornada.

Aos amigos Luiz Leonardo, Sandro, Eduardo Manco, Hugney, Kaisson, Ameno, Fabiana, Fábio Gatão, Alexandre Quatão, Rui, Fabão, Guilherme, Marcelo Fandi (*in memoriam*), pelo apoio e incentivo que sempre me dispensaram.

Aos funcionários Joana, Nei e Marli pela presteza com que sempre me atenderam.

A todos os meus amigos que, de alguma forma, contribuíram nesta caminhada, meus sinceros agradecimentos.

RESUMO

Oliveira Jr, A. C., Um Estudo Sobre a Obtenção de Qualidade de Serviço Fim-a-Fim em Redes IP com DiffServ e Engenharia de Tráfego baseada em MPLS, FEELT-UFG, Uberlândia - Brazil, 2003, 115p.

O surgimento de novas aplicações na Internet, tais como voz sobre IP (VoIP), ensino a distância interativo, realidade virtual, videoconferência, etc, impõe a necessidade de garantias de qualidade de serviço (*QoS - Quality of Service*) fim-a-fim. Tais aplicações exigem limites no atraso, sua variação (*jitter*) e nas perdas de pacotes. O modelo de serviço atual da Internet não possibilita a obtenção dos requisitos de QoS mencionados. Assim sendo, o modelo DiffServ quando combinado com a engenharia de tráfego, provida pelo MPLS pode fornecer uma arquitetura apropriada para assegurar garantias de QoS às aplicações. Neste trabalho, considera-se um ambiente de rede, onde os tráfegos multímidia oriundos de um ISP local são classificados na borda da rede backbone em sub-classes de tráfego definidas pelo DiffServ. Em seguida, estes tráfegos são mapeados em diferentes LSPs (*Label Switched Paths*) com diferentes características de vazão, atraso e jitter. Por meio de modelagem e simulação demonstra-se que o MPLS com a Engenharia de Tráfego combinado com o DiffServ garante os limites requeridos pelas aplicações multímidia para os parâmetros de tráfego e de QoS.

Palavras-chave: QoS fim-a-fim, MPLS, Serviços Diferenciados, Engenharia de Tráfego, Aplicações multimídia.

ABSTRACT

Oliveira Jr., A. C. A Study of End-to-End Quality of Service Over IP Networks with DiffServ and MPLS Traffic Engineering, FELT-UFG, Uberlândia - Brazil, 2003, 115p.

Currently, the Internet provides only best-effort service that treats packets equally, and recently new kinds of distributed applications, like Internet telephony, video conferencing, video-on-demand have been developed which are sensitive to the end-to-end Quality of Service (QoS) received from the network. Such applications demand limits in the delay, jitter and packet loss. The DiffServ model when combined with Traffic Engineering can provide a suitable architecture for ensuring end-to-end QoS guarantees to several kinds of applications. MPLS is emerging to be the choice for the core network applying multi-service networks, and providing Traffic Engineering. In this work, the following environment is considered: in the network edge, the packets are classified in sub-classes of traffic defined by the DiffServ, with different types of delay, jitter and throughput characteristics. In the MPLS backbone, these sub-classes of traffic are mapped in the different Label Switched Paths (LSPs). This work proposes to show through modelling and simulation, Traffic Engineering over MPLS combined with DiffServ guarantees, the limits required by multimedia applications to the parameters of traffic and QoS.

Keywords: Multi-protocol Label Switching, Quality of Service, Traffic Engineering, Differentiated Services, Multimedia Applications.

Sumário

LISTA DE FIGURAS	xiv
LISTA DE TABELAS	xvi
LISTA DE ABREVIATURAS	xvii
1 Introdução	1
1.1 Descrição do Problema	5
1.2 Objetivos Propostos	6
1.3 Estrutura da dissertação	7
2 Qualidade de Serviço nas Redes IP	9
2.1 Introdução	9
2.2 O protocolo IP e a Internet	10
2.2.1 Exigência das aplicações	10
2.3 Qualidade de Serviço	11
2.3.1 Serviço de Melhor Esforço	12
2.3.2 Protocolos de Sinalização	13
2.3.3 Prioridades	13

2.3.4	Escalonamento	13
2.3.5	Controle de Filas	14
2.3.6	Congestionamento	14
2.4	Necessidade de Qualidade de Serviço	15
2.4.1	QoS Fim-a-Fim	16
2.5	Parâmetros de QoS	16
2.5.1	Largura de Banda	17
2.5.2	Vazão	17
2.5.3	Atraso fim-a-fim	17
2.5.4	<i>Jitter</i> (Variação do Atraso)	17
2.5.5	Perdas	18
2.6	Algumas Soluções de QoS	19
2.6.1	Serviços Integrados (IntServ)	19
2.6.2	Serviços Diferenciados (DiffServ)	21
2.6.3	Multiprotocol Label Switching (MPLS)	23
2.6.4	Conclusões	26
3	MPLS e Engenharia de Tráfego	27
3.1	Introdução	27
3.2	Arquitetura MPLS (<i>Multi Protocol Label Switching</i>)	29
3.2.1	Rótulo MPLS	29
3.2.2	FEC (<i>Forward Equivalence Class</i>)	30
3.2.3	NHLFE (<i>Next Hop Label Forwarding Entry</i>)	31

3.2.4	FTN (<i>FEC to NHLFE</i>)	31
3.2.5	LSR (<i>Label Switching Router</i>)	31
3.2.6	LER (<i>Label Edge Router</i>)	31
3.2.7	LSP (<i>Label Switched Path</i>)	32
3.3	Distribuição de Rótulos	32
3.3.1	Estratégias de distribuição de rótulos	34
3.3.2	Controle da distribuição de associações	34
3.3.3	Modo de retenção	34
3.3.4	Protocolo LDP - (<i>Label Distribution Protocol</i>)	35
3.3.5	Protocolo CR-LDP (<i>Constraint-based Routing LDP</i>)	36
3.3.6	Protocolo RSVP-TE (RSVP - <i>Traffic Engineering</i>)	36
3.4	Engenharia de Tráfego com MPLS (MPLS-TE)	38
3.4.1	Túneis LSP	39
3.4.2	Troncos de tráfego	41
3.4.3	Congestionamento	43
3.4.4	Confiabilidade	44
3.4.5	Roteamento baseado em Restrições (<i>CBR - Constraint-based Routing</i>)	45
3.4.6	MPLS - TE utilizando CBR	46
3.5	Re-roteamento rápido MPLS (MPLS <i>fast-reroute</i>)	47
3.5.1	Método L3- <i>Control Driven</i>	48
3.5.2	Método proposto por Haskin	49
3.6	Conclusões	50

4	DiffServ e MPLS	52
4.1	Introdução	52
4.2	Arquitetura Diffserv	54
4.3	A classificação e o condicionamento do tráfego	55
4.3.1	Classificador	56
4.3.2	Medidor	57
4.3.3	Escalonamento	57
4.3.4	Policimento	57
4.3.5	Condicionadores	58
4.4	Encaminhamento do tráfego	58
4.4.1	PHB EF (<i>Expedited Forwarding</i>)	60
4.4.2	PHB AF (<i>Assured Forwarding</i>)	60
4.5	Serviços e Contratos	61
4.5.1	SLA (<i>Service Level Agreement</i>) e SLS (<i>Service Level Specification</i>) .	62
4.6	A Integração MPLS e DiffServ (MPLS-DS)	63
4.6.1	MPLS-DS com QoS fim-a-fim	64
4.6.2	E-LSP (<i>EXP-Inferred-PSC LSP</i>)	66
4.6.3	L-LSP (<i>Label-Only-Inferred-PSC LSP</i>)	67
4.6.4	Modelo de operação	67
4.6.5	Determinação do PHB de entrada	68
4.6.6	Determinação do PHB de saída	69
4.6.7	Encaminhamento do rótulo no domínio	69

4.6.8	Codificação da informação DiffServ na camada de encapsulamento	70
4.7	Conclusões	70
5	Avaliação de Desempenho de uma Rede IP com DiffServ e Engenharia de Tráfego Baseada em MPLS	73
5.1	Introdução	73
5.2	Classificação das Aplicações	75
5.2.1	Aplicações em tempo real	76
5.2.2	Aplicações Elásticas	77
5.3	Suporte à QoS Fim-a-Fim em Redes MPLS - DiffServ	78
5.3.1	Trabalhos Relacionados	79
5.4	Ambiente de Rede e Parâmetros de Simulação	80
5.5	Apresentação e Análise de Resultados	82
5.5.1	Resultados sem a inclusão de mecanismos de QoS	83
5.5.2	Resultados obtidos quando se utiliza somente o mecanismo DiffServ	85
5.5.3	Resultados obtidos quando se utiliza somente o mecanismo MPLS .	88
5.5.4	Resultados obtidos quando se utiliza os mecanismos DiffServ e MPLS	92
5.5.5	Resultados obtidos quando se utiliza somente o mecanismo DiffServ e o Re-roteamento IP Convencional	98
5.5.6	Resultados obtidos quando se utiliza os mecanismos DiffServ e MPLS com o método Haskin de Re-roteamento	100
5.5.7	Resultados obtidos quando se utiliza os mecanismos DiffServ e MPLS com o método L3 <i>Control Driven</i> de re-roteamento	101

5.5.8 Conclusões	102
6 Conclusões Gerais	104
7 Referências bibliográficas	109

Lista de Figuras

2.1	Domínio DiffServ.	21
3.1	Rótulo MPLS.	29
3.2	Encaminhamento de pacotes sem TE.	38
3.3	Encaminhamento de pacotes com TE.	39
3.4	Relação entre fluxos, troncos, LSPs e enlaces.	41
4.1	Elementos DiffServ.	56
4.2	Campo ToS e <i>byte</i> DS.	59
4.3	Modelo Simplificado de um LSR - DiffServ.	65
4.4	Campos EXP e DSCP.	66
4.5	E-LSP suportando três PHBs.	68
5.1	Ambiente de Rede.	81
5.2	Vazão sem mecanismo de QoS.	84
5.3	Atraso fim-a-fim sem mecanismo de QoS.	84
5.4	<i>Jitter</i> sem mecanismo de QoS.	85
5.5	Vazão com mecanismo DiffServ.	86

5.6	Atraso fim-a-fim com mecanismo DiffServ.	87
5.7	<i>Jitter</i> com mecanismo DiffServ.	87
5.8	Vazão com mecanismo MPLS e um ER-LSP.	89
5.9	Atraso fim-a-fim com mecanismo MPLS e um ER-LSP.	89
5.10	<i>Jitter</i> com mecanismo MPLS e um ER-LSP.	90
5.11	Vazão com mecanismo MPLS e dois ER-LSPs.	91
5.12	Atraso fim-a-fim com mecanismo MPLS e dois ER-LSPs.	92
5.13	<i>Jitter</i> com mecanismo MPLS e dois ER-LSPs.	93
5.14	Vazão com os mecanismos DiffServ e MPLS e um ER-LSP.	94
5.15	Atraso fim-a-fim com os mecanismos DiffServ e MPLS e um ER-LSP. . . .	95
5.16	<i>Jitter</i> com os mecanismos DiffServ e MPLS e um ER-LSP.	95
5.17	Vazão com os mecanismos DiffServ e MPLS e dois ER-LSPs.	97
5.18	Atraso fim-a-fim com os mecanismos DiffServ e MPLS e dois ER-LSPs. . .	97
5.19	<i>Jitter</i> com os mecanismos DiffServ e MPLS e dois ER-LSPs.	98
5.20	Vazão com o mecanismo DiffServ e falha no enlace.	99
5.21	Vazão com DiffServ e MPLS e método Haskin de re-roteamento.	101
5.22	Vazão com DiffServ e MPLS e método L3 <i>Control Driven</i> de re-roteamento.	102

Lista de Tabelas

3.1	Comparação entre RSVP-TE e CR-LDP.	37
4.1	Mapeamento DSCP-EXP.	70
5.1	Requisitos típicos de QoS.	75
5.2	Classificação das aplicações.	82

LISTA DE ABREVIATURAS

AF	<i>Assured Forwarding</i>
ARIS	<i>Aggregated Route Based IP Switching</i>
ATM	<i>Asynchronous Transfer Mode</i>
BA	<i>Behavior Aggregate</i>
BGP	<i>Border Gateway Protocol</i>
BE	<i>Best Effort</i>
CLS	<i>Controlled Load Service</i>
CBQ	<i>Class Based Queuing</i>
CBR	<i>Constant Bit Rate</i>
CFQ	<i>Class-Based Fair Queuing</i>
CIR	<i>Committed Information Rate</i>
COS	<i>Class of Service</i>
CR-LDP	<i>Constrained-based Routing LDP</i>
DIFFSERV	<i>Differentiated Services</i>
DS	<i>Differentiated Services</i>
DSCP	<i>DiffServ Code Point</i>
DLCI	<i>Data Link Connection Identifier</i>
E-LSP	<i>EXP-Inferred-PSC LSP</i>
ECN	<i>Explicit Congestion Notification</i>
EF	<i>Expedited Forwarding</i>
ER-LSP	<i>Explicit Routing - Label Switching Path</i>

FEC	<i>Forwarding Equivalence Class</i>
FIFO	<i>First In First Out</i>
FR	<i>Frame Relay</i>
FTN	<i>FEC to NHLFE</i>
FTP	<i>File Transfer Protocol</i>
GMPLS	<i>Generalized Multiprotocol Label Switching</i>
GS	<i>Guaranteed Service</i>
IETF	<i>Internet Engineering Task Force</i>
IntServ	<i>Integrated Services</i>
IP	<i>Internet Protocol</i>
IPSEC	<i>Internet Protocol Security</i>
IPv4	<i>Internet Protocol version 4</i>
IPv6	<i>Internet Protocol version 6</i>
ISO	<i>International Organization for Standardization</i>
ISP	<i>Internet Service Provider</i>
LAN	<i>Local Area Network</i>
LDP	<i>Label Distribution Protocol</i>
L-LSP	<i>Label-Only-Inferred-PSC LSP</i>
LSP	<i>Label Switched Path</i>
LSR	<i>Label Switched Router</i>
LER	<i>Label Edge Router</i>
LIB	<i>Label Information Base</i>

MF	<i>Multifield</i>
MSS	<i>Maximum Segment Size</i>
MTU	<i>Maximum Transfer Unit</i>
MPLS	<i>Multiprotocol Label Switching</i>
NS	<i>Network Simulator</i>
NHLFE	<i>Next Hop Label Forwarding Entry</i>
OSPF	<i>Open Shortest Path First</i>
PHB	<i>Per-Hop Behavior</i>
PPP	<i>Point-to-Point Protocol</i>
PSC	<i>PHB Scheduling Class</i>
QoS	<i>Quality of Service</i>
QoSR	<i>QoS Routing</i>
RED	<i>Random Early Detection</i>
RIP	<i>Routing Information Protocol</i>
RR	<i>Round Robin</i>
RSVP	<i>Resource Reservation Protocol</i>
RSVP-TE	<i>Resource Reservation Protocol - Traffic Engineering</i>
SFQ	<i>Stochastic Fair Queuing</i>
SLA	<i>Service Level Agreement</i>
SONET	<i>Synchronous Optical Network</i>
TCP	<i>Transport Control Protocol</i>
TDP	<i>Tag Distribution Protocol</i>

TTL	<i>Time To Live</i>
TE	<i>Traffic Engineering</i>
ToS	<i>Type of Service</i>
UDP	<i>User Datagram Protocol</i>
VBR	<i>Variable Bit Rate</i>
VC	<i>Virtual Connection</i>
VCI	<i>Virtual Channel Identifier</i>
VPI	<i>Virtual Path Identifier</i>
VPN	<i>Virtual Private Network</i>
VoIP	<i>Voice over IP</i>
WAN	<i>Wide Area Network</i>
WFQ	<i>Weighted Fair Queueing</i>
WRED	<i>Weighted RED</i>
WRR	<i>Weighted Round Robin</i>

Capítulo 1

Introdução

Atualmente, a Internet opera apenas com o serviço de melhor esforço (*Best Effort - BE*), onde cada usuário da rede envia seus dados e compartilha a largura de banda disponível com todos os fluxos de dados dos demais usuários. Com o surgimento de novas aplicações multimídia e em tempo real, com requisitos diferentes das aplicações tradicionais e rígidos em relação aos parâmetros de Qualidade de Serviço (*Quality of Service - QoS*), sinaliza-se a necessidade de serviços adicionais a serem oferecidos pela Internet [1].

A garantia aos parâmetros de QoS é indispensável para viabilizar as novas aplicações multimídia na Internet, uma vez que estas aplicações sendo sensíveis à QoS passam a ter maior prioridade, enquanto usuários de aplicações tradicionais continuam utilizando o serviço de melhor esforço. Assim, em caso de congestionamento, os pacotes de melhor esforço serão descartados prioritariamente, sendo possível vender novos serviços de acordo com a largura de banda que se possa ou deseja contratar, fazendo com que se defina, assim,

uma nova forma de comércio na Internet.

Várias pesquisas estão sendo desenvolvidas para que seja possível implementar QoS na Internet. Como exemplo, tem-se a arquitetura de Serviços Intergrados (*Integrated Services - IntServ*) [2], a arquitetura de Serviços Diferenciados (*Differentiated Services - DiffServ*) [3], a Engenharia de Tráfego (*Traffic Engineering - TE*) [4] [6], o encaminhamento baseado em rótulos (*Multiprotocol Label Switching - MPLS*) [5] e o roteamento baseado em restrições (*Constraint-Based Routing*) [6].

A arquitetura de Serviços Integrados destaca-se por garantir a manutenção dos níveis de QoS fim a fim. Mas, necessita-se que cada fluxo de dados individual tenha uma reserva de recursos, o que gera um problema escalar nesta arquitetura. A manutenção de um estado por fluxo e o processamento a eles associado é inviável em roteadores com grande volume de tráfego em uma rede de escala global como a Internet. A impossibilidade de ser escalável limitou a adoção da proposta IntServ de forma intensiva na Internet. Atualmente, o uso do IntServ é recomendado somente para redes de pequeno porte [2].

Como solução ao problema de escalabilidade do IntServ surgiu a proposta dos Serviços Diferenciados, que objetiva prover serviços distintos de forma escalável sem a necessidade de manutenção de um estado por fluxo e de sinalização a cada salto. O tratamento oferecido por um nó compatível com esta proposta é efetuado em uma agregação de fluxos, não mais em fluxos individuais. Os fluxos individuais são classificados em agregação de fluxos pré-estabelecidos, que receberão serviços diferenciados entre elas. Esta classificação e marcação de pacotes ocorrem nas fronteiras de um domínio DiffServ. No interior do domínio DiffServ, as agregações de fluxos serão encaminhadas por cada nó segundo um

Comportamento por Salto (*Per-Hop Behavior - PHB*) selecionado através de um mapeamento entre o código que identifica a agregação e este comportamento. O serviço fim a fim é obtido pelo uso combinado dos PHBs em todos os nós do domínio. Atualmente dois PHBs estão padronizados pela comunidade de pesquisa em diferenciação de serviços: o PHB de Encaminhamento Expresso (*Expedited Forwarding - EF*) [7] e o PHB de Encaminhamento Assegurado (*Assured Forwarding - AF*) [8].

Como em todas as tecnologias emergentes, o esforço de pesquisa e desenvolvimento dos muitos fabricantes levou à criação de um grupo de trabalho para propor uma arquitetura de suporte à comutação de rótulos, em plataforma aberta e interoperável. O IETF tomou a iniciativa de criar um grupo de trabalho, denominado *Multiprotocol Label Switching - MPLS*. A tecnologia é baseada em rótulos que habilitam o encaminhamento de pacotes a partir da análise de rótulos de comprimento fixo e reduzido. Assim, os pacotes podem ser agrupados em classes de equivalência de encaminhamento (*Forward Equivalence Class - FEC*), num processo denominado roteamento explícito, efetuado pelos nós principais da infra-estrutura MPLS, denominados *Label Switching Routers* (LSR). Com isso, o encaminhamento dos pacotes é simplificado, se comparado com o roteamento convencional, onde os cabeçalhos dos pacotes são analisados pacote por pacote.

O roteamento explícito é uma técnica poderosa, podendo ser aplicada para vários propósitos. Para muitas aplicações, o roteamento implícito, baseado na análise dos data-gramas, pacote a pacote, gera uma sobrecarga muitas vezes inaceitável. O MPLS, permite que os pacotes sejam classificados, a partir de rótulos atribuídos na admissão dos nós MPLS, e encaminhados, dentro de uma mesma classe, num caminho virtual, sem a

necessidade de ser analisado nó a nó.

A engenharia de tráfego é o processo de seleção do caminho pelo qual o tráfego de dados deve ser encaminhado, a fim de propiciar o balanceamento de carga entre vários enlaces que interligam roteadores e switches na rede. A engenharia de tráfego é muito útil em redes complexas (como a Internet) onde existem diversos caminhos alternativos (paralelos) disponíveis no ambiente de rede.

Nas redes baseadas em datagramas a engenharia de tráfego é dificultada, devido ao caráter de não orientação à conexão. Dentro deste enfoque, a arquitetura MPLS possibilita que fluxos de dados oriundos de um particular nó de entrada direcionado para um particular nó de saída, possam ser identificados individualmente, como num circuito virtual. Além disso, dentro da estrutura de rótulos, com o roteamento explícito, pode ser habilitado o encaminhamento destes fluxos a partir de um caminho preferencial (na arquitetura MPLS é denominado *Label Switching Path - LSP*).

Com as FECs a possibilidade de associar estas classes de equivalência às classes de serviços, também melhora a utilização desta arquitetura como suporte aos Serviços Diferenciados (*DiffServ*).

Na proposta de serviços diferenciados, os pacotes são agrupados em diferentes classes de tráfego. Cada classe tem uma marca a ela associada, utilizando os bits DSCP (*DiffServ Code Point*). O DiffServ é bastante escalável, pois não necessita de muitas sinalizações, assegurando aos fluxos, largura de banda apropriada e garantias de atraso e *jitter* no núcleo da rede. O princípio desta arquitetura é concentrar na borda da rede a complexidade das operações envolvidas no tratamento do tráfego, de modo que o interior da rede permaneça

o mais simples possível. Assim, os roteadores de borda desta arquitetura são responsáveis por monitorar, classificar e marcar os pacotes a serem inseridos na rede. Cada roteador no núcleo da rede implementa técnicas de gerenciamento de filas e descarte de pacotes, baseados na marcação do pacote pelo seu comportamento por nó (*Per Hop Behavior - PHB*).

1.1 Descrição do Problema

Diante da complexidade da Internet, que é formada por vários Sistemas Autônomos (*AS - Autonomous Systems*), uma malha complexa de enlaces e roteadores é formada. Diante disso a arquitetura DiffServ sendo implementada em cada AS, provê atualmente a melhor forma de provisão de QoS fim a fim na Internet [9]. A arquitetura DiffServ utiliza algoritmos de roteamento IP convencionais, ou seja, o menor custo para rotear os pacotes de uma origem até um destino através destes ASs. Para uma melhor obtenção dos parâmetros de QoS exigidos por aplicações típicas, é necessário um balanceamento de carga nos roteadores, de forma que otimize as rotas entre um *host* de origem e um *host* de destino. Várias propostas para otimizar essas rotas vem sendo estudadas pela comunidade científica, destacando-se a engenharia de tráfego com o mecanismo MPLS.

A engenharia de tráfego MPLS é fundamental quando se deseja obter qualidade de serviço fim-a-fim em redes IP, permitindo uma utilização mais eficiente da rede e uma rápida correção em falhas de enlaces e nós. Esta tecnologia utiliza rótulos para encaminhar os pacotes através de seus domínios.

No roteador de entrada da rede os rótulos são associados aos pacotes baseados em Classes de Equivalência de Encaminhamento, e a complexidade fica a cargo da borda da rede, sendo que os roteadores do núcleo apenas realizam a troca dos rótulos para encaminhamento. Estas FECs podem ser associadas às classes de tráfego do DiffServ. O MPLS permite um roteamento explícito, baseado nos endereços de origem e destino, permitindo a introdução de novos serviços IP, como é o caso do DiffServ.

O principal objetivo do MPLS é agilizar o roteamento e o balanceamento de carga através da engenharia de tráfego. Como ele não oferece qualidade de serviço diretamente, necessita de outro mecanismo para prover QoS. Assim o DiffServ é o mecanismo de provisão de QoS mais indicado para ser utilizado juntamente com o MPLS por ser bastante escalável e possuir características em comum com o MPLS.

Neste trabalho, considera-se um ambiente de rede, onde os tráfegos multímidia oriundos de um ISP local são classificados na borda da rede backbone em sub-classes de tráfego definidas pelo DiffServ. Em seguida, estes tráfegos são mapeados em diferentes LSPs (*Label Switched Paths*) com diferentes características de vazão, atraso e jitter.

1.2 Objetivos Propostos

Os objetivos deste trabalho se resumem em:

- Relacionar as classes de tráfego do DiffServ em classes de tráfego do MPLS através do mapeamento entre os campos DSCP e EXP;
- Utilizar a Engenharia de Tráfego provida pelo MPLS para balancear a carga nos

roteadores, de forma que otimize as rotas entre um *host* de origem e um *host* de destino;

- Avaliar o impacto do tráfego de dados juntamente com tráfegos de Voz e Vídeo;
- Avaliar os tráfegos de Voz e Vídeo em termos de parâmetros de QoS, como: vazão, atraso fim-a-fim e *jitter*.

Desta forma, o objetivo principal deste trabalho é a visualização clara de que com a combinação de duas arquiteturas distintas (MPLS e DiffServ) obtém-se a QoS desejada para aplicações multimídia e em tempo real.

1.3 Estrutura da dissertação

A estratégia utilizada para atingir os objetivos acima está descrita em cada um dos capítulos do trabalho. Estruturalmente o texto está dividido da seguinte forma:

- Capítulo 1: É a introdução relativa a este trabalho;
- Capítulo 2: Apresenta inicialmente alguns conceitos sobre qualidade de serviço. Em seguida, apresenta-se uma visão geral das soluções que atualmente têm sido estudadas e propostas na literatura referentes à qualidade de serviço na Internet;
- Capítulo 3: faz-se uma descrição da arquitetura MPLS e a Engenharia de Tráfego, destacando os conceitos básicos e definições;
- Capítulo 4: É apresentada a arquitetura de Serviços Diferenciados (DiffServ) e a integração entre as arquiteturas MPLS e DiffServ. Inicialmente destaca-se neste

capítulo as características principais da arquitetura DiffServ, bem como as classes de serviços suportadas por essa arquitetura. Em seguida, aborda-se a interoperabilidade de duas arquiteturas para provêr QoS fim-a-fim, destacando-se pesquisas recentes nessa área e os problemas de compatibilidade e possíveis soluções;

- Capítulo 5: A avaliação de desempenho sobre a obtenção de qualidade de serviço fim a fim para aplicações multimídia em redes IP, com DiffServ e Engenharia de Tráfego baseada em MPLS;
- Capítulo 6: Apresentação das conclusões finais relevantes do trabalho, bem como de sugestões para trabalhos futuros que podem ser realizados a partir do mesmo.

Capítulo 2

Qualidade de Serviço nas Redes IP

2.1 Introdução

A Internet era restrita aos centros de pesquisas e universidades, para que os pesquisadores pudessem compartilhar e divulgar seus trabalhos. Hoje o uso comercial da Internet em grande escala é uma realidade e, consequentemente, problemas surgiram.

Atualmente, a Internet opera com o serviço de melhor esforço. Os usuários da rede enviam seus dados e a largura de banda é compartilhada com todos os fluxos de dados dos outros usuários. Quando há congestionamento, os pacotes são descartados, sem distinção, nos roteadores. Não há garantia de que o serviço será realizado com sucesso.

Com o advento de novas aplicações multimídia, com diferentes requisitos das aplicações tradicionais em relação aos parâmetros de QoS, surge a necessidade de serviços adicionais a serem oferecidos pela Internet [1]. Muitas aplicações como voz sobre IP, vídeo-conferência, ensino a distância e aplicações em tempo real necessitam de tais garantias.

2.2 O protocolo IP e a Internet

O crescimento acentuado na utilização da Internet tem aumentado o tráfego, assim como o número de usuários e aplicações. Entretanto, o aumento crescente da largura de banda para o transporte de dados não é, em muitos casos, suficiente para acomodar este aumento de demandas. O tráfego na Internet e nas Intranets corporativas não só aumentou, como também mudou suas características. Aplicações novas têm exigências de novos serviços e como resultado a Internet precisa mudar também.

O protocolo IP possibilitou uma rede de comunicação global com uma variedade infinita de sistemas e meios de transmissão. Troca de e-mails e navegação na WEB fazem parte do dia a dia das pessoas a título de trabalho, estudo e lazer. Outras redes como a de telefonia, de rádio e de televisão também estão convergindo para IP, aproveitando-se de sua grande abrangência e flexibilidade. Com estas novas redes aparecem novas aplicações e novos usuários [10]

2.2.1 Exigência das aplicações

As redes existem para apoiar as aplicações. Assim, diferentes aplicações possuem exigências operacionais diferentes que exigem serviços de rede diferentes. O aumento de tráfego na rede requer aumento da capacidade em termos de largura de banda. Aplicações como telefonia IP têm outras exigências e o simples aumento de largura de banda pode não ser a única solução ou pode ser somente parte da solução.

Têm-se dois tipos de classes de tráfego:

- Constante, que é a entrega previsível a uma taxa de bits relativamente constante. Fluxos de áudio e vídeo geralmente são considerados CBR porque eles têm limites superiores quantificáveis, embora suas taxas flutuem freqüentemente.
- Rajada, entrega dados a uma taxa variável. Aplicações como transferência de arquivos que movimentam grandes volumes de dados e podem aumentar e diminuir sua taxa freqüentemente, mas sem nenhum limite superior.

2.3 Qualidade de Serviço

De acordo com a ISO a Qualidade de Serviço (QoS) é definida como sendo o efeito do desempenho de um serviço, o qual determina o grau de satisfação de um usuário do serviço.

Para obter QoS, disponibilidade de largura de banda é o ponto fundamental. Mecanismos de QoS gerenciam o uso da largura de banda de acordo com as exigências da aplicação e de acordo com os parâmetros de configuração e administração da rede e, assim, não podem prover garantias nos casos onde compartilhamento estiver envolvido. Conseqüentemente, QoS com um nível de serviço garantido requer alocação de recursos a fluxos de dados individuais.

Considerando que largura de banda é um recurso finito, a prioridade dos projetistas de redes com QoS tem sido assegurar que o tráfego de melhor esforço não pereça após as reservas serem efetivadas. Habilitar QoS para aplicações de alta prioridade não deve impossibilitar o uso das aplicações de baixa prioridade.

Essencialmente existem dois mecanismos básicos disponíveis para prover QoS em redes IP:

- Reserva de Recursos: os recursos da rede são divididos de acordo com os requisitos de QoS da aplicação, e sujeitos à política de administração de largura de banda. O RSVP (*Resource ReSerVation Protocol*), por exemplo, fornece os mecanismos para implementação de serviços integrados (IntServ) [2] baseado na reserva de recursos.
- Priorização: o tráfego da rede é classificado e os recursos de rede são divididos de acordo com critérios de políticas de administração de largura de banda. Para habilitar QoS os mecanismos de classificação dão tratamento preferencial a aplicações identificadas como tendo requisitos mais exigentes. A arquitetura de Serviços Diferenciados (DiffServ) [3] provê esse mecanismo.

Estes dois grupos de protocolos de QoS e algoritmos são complementares, ou seja, eles são projetados para uso em conjunto, de forma a acomodar diferentes exigências operacionais em diferentes contextos de redes.

2.3.1 Serviço de Melhor Esforço

Este serviço é conhecido como sem QoS, provê simplesmente a conectividade básica sem garantias. A Internet de hoje é um bom exemplo de serviço de melhor esforço. Apesar de não oferecer garantias, é adequado para uma grande gama de aplicações de rede, tais como, correio eletrônico e transferência de arquivos, pois estas se adaptam facilmente às condições da rede.

2.3.2 Protocolos de Sinalização

A finalidade de um protocolo de sinalização no que diz respeito à qualidade de serviço em redes IP é informar ou solicitar aos elementos da rede (roteadores), a respeito da sua necessidade de qualidade de serviço (QoS).

Além disso, os protocolos de sinalização permitem também que os equipamentos de rede, por exemplo, roteadores, possam trocar informações visando à garantia da qualidade de serviço aceita pela rede. Exemplos de protocolos de sinalização é o RSVP (*Resource Reservation Protocol*) [11] e o LDP (*Label Distribution Protocol*) [12].

2.3.3 Prioridades

Os algoritmos de prioridade são um outro mecanismo utilizado pelos equipamentos de rede para a garantia da qualidade de serviço. Assim, a prioridade pode ser entendida como um mecanismo que provê diferentes tempos de espera para o processamento da informação.

Estes algoritmos são tipicamente implementados em roteadores, mas algumas tecnologias de rede da camada 2 também suportam a utilização destes mecanismos.

Um exemplo é o Priority Queuing, que é utilizado por alguns fornecedores para priorização de pacotes IP nas filas de saída de roteadores.

2.3.4 Escalonamento

No contexto da qualidade de serviço, o mecanismo de escalonamento presente nos roteadores procura garantir que fluxos diferentes de pacotes obtenham os recursos que lhes

foram alocados. No caso, a banda disponível é distribuída de forma justa entre os fluxos ativos existentes. Alguns exemplos de mecanismos de escalonamento são: WRR (*Weighted Round Robin*), GPS (*Generalized Processor Sharing*), CBQ (*Class Based Queuing*), WFQ (*Weighted Fair Queuing*).

2.3.5 Controle de Filas

O controle de filas diz respeito aos mecanismos de descarte de pacotes. A política de descarte de pacotes é necessária na ocorrência de um congestionamento e visa igualmente a garantia de equidade quanto ao compartilhamento da banda. Alguns exemplos de controle de filas são: SFQ (*Stochastic Fair Queuing*), CFQ (*Class-Based Fair Queuing*), WFQ (*Weighted Fair Queuing*).

2.3.6 Congestionamento

Os mecanismos de controle de congestionamento são também importantes para a implantação da qualidade de serviço numa rede IP. A idéia básica destes mecanismos é a inibição dos fluxos de pacotes durante o período de congestionamento de forma que os geradores de fluxos de pacotes IP reduzam a sua carga sobre a rede. Com menos pacotes sendo entregues à rede tem-se uma tendência de redução no nível de congestionamento. Alguns exemplos de algoritmos de congestionamento de filas são: RED (*Random Early Detection*), WRED (*Weighted Random Early Detection*), ECN (*Explicit Congestion Notification*).

2.4 Necessidade de Qualidade de Serviço

A Internet hoje é um ambiente comercial e apresenta problemas fundamentais. A cada ano, os provedores de acesso à Internet (ISPs) aumentam a capacidade de suas redes [1], mas enfrentam limitações em oferecer uma maior variedade de diferenciação nos serviços.

Uma diferenciação possível em termos de preço e qualidade está na largura de banda do enlace de acesso do usuário até o provedor. Um exemplo simples seria as aplicações de vídeo possíveis na Internet, a qualidade das imagens impede a sua utilização para a maior parte dos usuários. A causa disso é o modelo de serviço único (melhor esforço), que trata todos os pacotes com igualdade e não oferece garantias de QoS.

Nas horas de pico, a demanda dos usuários costuma ser maior do que a largura de banda disponível, o que leva a um baixo desempenho. A simplicidade do protocolo IP é a causa do seu grande sucesso e espantosa escalabilidade, que deixa toda a complexidade para os sistemas finais. Com o crescimento da Internet nos últimos anos, os problemas do IP estão cada vez mais visíveis [20]. A ocorrência de congestionamentos nos roteadores acarreta atrasos e perdas de pacotes.

O modelo de melhor esforço não consegue sempre oferecer um serviço razoável, que o torna inadequado para várias aplicações [20]. Mesmo em redes com pouca carga, atrasos na entrega dos pacotes podem apresentar grandes variações, afetando negativamente aplicações em tempo real. Para oferecer garantias de serviço é necessário adicionar algum tipo de complexidade à rede, para que ela possa diferenciar tráfego e possibilitar níveis diferentes de serviços para usuários e aplicações distintos. Redes IP necessitam de mecanismos de gerenciamento ativo da largura de banda, ou em outras palavras, necessitam de

Qualidade de Serviço (QoS).

2.4.1 QoS Fim-a-Fim

A qualidade de serviço fim-a-fim, consiste na habilidade da rede em prover o serviço requerido por um tráfego específico de uma extremidade a outra da rede.

Existem soluções que operam somente em um domínio específico, enquanto que os clientes requerem uma solução de uma extremidade a outra, independente dos ASs intermediários.

Inicialmente, para a obtenção de qualidade de serviço fim-a-fim, muitos ISPs tratam um acordo com os clientes sobre os parâmetros de QoS e consequentemente o SLA (*Service Level Agreement*) é o fator de diferenciação entre os provedores.

2.5 Parâmetros de QoS

De acordo com as aplicações, a solicitação de QoS é realizada através de um Acordo de Nível de Serviço. O SLA define o conjunto de critérios mensuráveis a serem utilizados para validar o recebimento do serviço contratado. Esses critérios definem os parâmetros de qualidade de serviço, tais como: Largura de banda, atraso fim-a-fim, *jitter* (variação do atraso) e perdas.

2.5.1 Largura de Banda

A largura de banda é definida como sendo a taxa de transmissão de dados máxima que pode ser alcançada entre dois pontos, sendo normalmente uma característica do enlace físico. Cada tipo de aplicação exige uma quantidade de largura de banda para que tenha um mínimo de qualidade na transmissão. Como a largura de banda está diretamente ligada ao enlace físico, é necessária uma atualização na infra-estrutura para que aplicações exigentes de grande quantidade de largura de banda não sofram impactos negativos. Muitas vezes somente esse investimento na infra-estrutura não é suficiente [54].

2.5.2 Vazão

O termo vazão (*throughput*) é a quantidade de pacotes entregues livres de erros no receptor [54].

2.5.3 Atraso fim-a-fim

É o tempo entre o envio de uma mensagem por um nó e a recepção desta mensagem pelo nó destino. Este atraso ocorre no caminho de transmissão ou em um dispositivo no caminho de transmissão. Em um roteador, o atraso é o montante de tempo entre a recepção do pacote e a sua transmissão [54].

2.5.4 *Jitter* (Variação do Atraso)

O *jitter* é um parâmetro importante para a qualidade de serviço. No caso, o *jitter* é importante para as aplicações executando em rede cuja operação adequada depende

de alguma forma da garantia de que as informações (pacotes) devem ser processadas em períodos de tempo bem definidos. Este é o caso, por exemplo, de aplicações de voz sobre IP (VoIP), aplicações de tempo real, etc.

Do ponto de vista de uma rede de computador, o *jitter* pode ser entendido como a variação no tempo e na seqüência de entrega das informações devido à variação na latência (atrasos) da rede.

O *jitter* introduz distorção no processamento da informação na recepção e deve ter mecanismos específicos de compensação e controle que dependem da aplicação em questão. Genericamente, uma das soluções mais comuns para o problema consiste na utilização de buffers (Técnica de "*buffering*").

2.5.5 Perdas

As perdas de pacotes em redes IP ocorrem principalmente em função de fatores tais como: Descarte de pacotes nos roteadores e perda de pacotes devido à erros ocorridos na camada 2 (PPP - *Point-to-Point Protocol*, *Ethernet*, *frame relay*, ATM.) durante o transporte dos mesmos.

De maneira geral, as perdas de pacotes em redes IP são um problema sério para determinadas aplicações como, por exemplo, a voz sobre IP. Neste caso específico, a perda de pacotes com trechos de voz digitalizada implica numa perda de qualidade eventualmente não aceitável para a aplicação. O que fazer em caso de perdas de pacotes é uma questão específica de cada aplicação em particular.

Do ponto de vista da qualidade de serviço da rede (QoS) a preocupação é normalmente

no sentido de especificar e garantir limites razoáveis (Taxas de Perdas) que permitam uma operação adequada da aplicação.

2.6 Algumas Soluções de QoS

A *Internet Engineering Task Force* (IETF) vem desenvolvendo uma série de propostas que visam enfrentar a questão de QoS na Internet. Destaca-se entre elas a arquitetura de Serviços Integrados [2], a arquitetura de Serviços Diferenciados [3], o MPLS (*Multiprotocol Label Switching*) [5] e a Engenharia de Tráfego [6].

2.6.1 Serviços Integrados (IntServ)

A especificação do modelo de Serviços Integrados, ou simplesmente Intserv [2], descreve as principais funções de provisão de QoS que devem estar presentes em um ambiente de fornecimento de serviços. O modelo apresenta como uma das suas características mais particulares a alta granularidade na reserva de recursos, realizada por fluxos individuais. Uma ênfase especial é reservada no modelo à especificação de serviços, que deve sempre ocorrer por intermédio de classes ou categorias padronizadas. As categorias de serviço Intserv resumem-se à garantida [14] e à de carga controlada [15].

A categoria de serviço QoS garantida, ou simplesmente serviço garantido, como o próprio nome indica, fornece aos seus usuários uma garantia de retardo máximo fim-a-fim, sem perda de pacotes, sendo indicada às aplicações em tempo real não elásticas. Esse serviço representa o outro extremo em relação ao controle de retardo dos pacotes em

comparação com o serviço de melhor esforço.

Os recursos necessários ao fornecimento do serviço QoS garantida devem ser reservados exclusivamente, o que torna esse o serviço de maior custo quando comparado com os demais.

Uma generalização da categoria de serviços garantidos, não plenamente suportados pelo modelo intserv, é a dos serviços probabilísticos, onde a garantia é fornecida em termos estatísticos. Como vantagem, a categoria de serviços probabilísticos, também chamada de serviços previsíveis, permite o compartilhamento de um percentual dos recursos, o que diminui os custos de implementação. Por outro lado, as aplicações que fazem uso desse serviço terão que ser mais tolerantes a variações ocasionais na qualidade de serviço fornecida pela infra-estrutura de comunicação.

As dificuldades de utilização do serviço de melhor esforço no suporte às aplicações multimídia reside na degradação da QoS fornecida pelo provedor em situações de congestionamento. Em condições normais, esse serviço, mesmo sem fornecer garantias de QoS, obtém-se resultados satisfatórios com a grande maioria das aplicações multimídia que foram desenvolvidas para utilizar a Internet atualmente.

A partir dessas constatações, foi proposta a categoria de serviço de carga controlada, ou simplesmente serviço de rede de carga controlada, que fornece um retardo baixo, próximo ao oferecido por um serviço de melhor esforço em uma infraestrutura de rede controlada.

A definição do que vem a ser um retardo baixo vai depender da forma com que os recursos necessários ao fornecimento do referido serviço foram reservados por cada provedor. Cabe a este controlar o número de usuários atendidos simultaneamente de forma

a manter o nível de utilização da rede em um patamar relativamente baixo. O serviço de carga controlada é indicado às aplicações em tempo-real tolerantes, em especial às tolerante-adaptativas. O serviço de rede de carga controlada também é considerado um serviço probabilístico, onde o percentual de garantia é determinado pelo provedor, ou seja, sem controle por parte da aplicação.

2.6.2 Serviços Diferenciados (DiffServ)

A arquitetura DiffServ [3] baseia-se em reservar recursos para um conjunto de fluxos e não para um único fluxo, como no IntServ.

A arquitetura DiffServ busca oferecer um tratamento diferenciado aos diversos fluxos na Internet, através de classes de serviços, de acordo com os parâmetros de QoS (vazão, atraso fim-a-fim, variação de atraso (*jitter*) e perda de pacotes). Este tratamento diferenciado é oferecido no interior de um domínio de Diferenciação de Serviços, conforme a Figura 2.1.

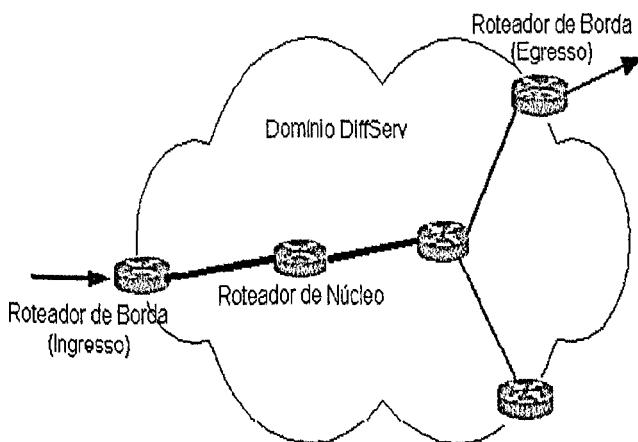


Figura 2.1: Domínio DiffServ.

O domínio DiffServ é composto pelos roteadores de borda e pelos roteadores de núcleo. Os roteadores de borda são responsáveis pela classificação do fluxo e pela sua adequação às condições pré-estabelecidas entre o cliente e o provedor de serviços. Dependendo do sentido do tráfego, os roteadores de borda podem ser denominados de roteadores de ingresso ou egresso.

Os roteadores de núcleo do domínio DS encaminham os fluxos de acordo com a classe de serviço estabelecida. Assim, os datagramas IP são marcados, pelo usuário ou pelo provedor de acesso, com uma determinada classe de QoS. Baseados nessas classes, os roteadores reservam recursos para lidar com os pacotes de cada classe.

Os recursos alocados para uma determinada classe é o mesmo para todos os pacotes de usuários de uma mesma classe. Desta forma, os recursos que possibilitam o envio de pacotes com uma determinada QoS são compartilhados por todos os pacotes que estão marcados como sendo da mesma classe de QoS.

No DiffServ cada pacote é diferenciado através da alteração do estado dos bits do campo DS byte do cabeçalho do datagrama IP. O octeto DS é uma redefinição do octeto TOS (*Type Of Service*) no IPv4 e do octeto *Traffic Class* do IPv6.

Dos 8 bits, 6 são usados para definir o tratamento que será dado em cada pacote por cada roteador. Os outros 2 bits restantes não são usados atualmente.

Para um usuário receber serviços diferenciados de seu ISP, ele deve ter um SLA, ou seja, um acordo de nível de serviço com seu ISP. Um SLA basicamente especifica as classes de serviço suportadas e a quantidade de tráfego permitida em cada classe.

2.6.3 Multiprotocol Label Switching (MPLS)

O modelo tradicional de roteamento IP, baseado na determinação do próximo roteador para onde cada pacote deve ser enviado a partir do endereço de destino, mostrou-se inadequado nos grandes provedores de rede da Internet, onde o maior volume de fluxos transportados exige mecanismos de encaminhamento mais eficientes. Como uma alternativa a esse modelo, o roteamento explícito sugere a determinação à priori da seqüência completa dos roteadores que compõem uma rota. Calculada pelo roteador localizado na borda de entrada do provedor, a referida seqüência, chamada de rota explícita, é inserida no cabeçalho dos pacotes transportados pelo provedor. Dessa forma, a determinação do caminho a ser seguido por cada pacote é realizada pela simples leitura da rota explícita. Porém, a inclusão de uma seqüência de tamanho variável em cada pacote transportado representa um custo adicional de processamento e largura de banda. Para minimizá-lo, substituiu-se à referida seqüência por um rótulo pequeno e de tamanho fixo, no mecanismo de encaminhamento conhecido como comutação por rótulos, empregado, há bastante tempo, nas redes frame relay e ATM.

A tecnologia MPLS representa um esforço de padronização, organizado pelo IETF, das diversas técnicas de comutação por rótulos em redes IP, implementadas nos últimos anos através de soluções proprietárias de diversos fabricantes como a IP Switching (Ipsilon), a ARIS (IBM) e a Tag Switching (Cisco), para citarmos algumas. Seu funcionamento é baseado na identificação dos pacotes que são transportados por uma mesma rota ou, em outras palavras, que pertençam à mesma classe de encaminhamento ou FEC (*Forward Equivalence Class*). A identificação da FEC ao qual um pacote pertence é realizado pelo

roteador de borda de entrada do domínio MPLS, responsável por acrescentar o rótulo ao mesmo. Os roteadores do domínio, chamados de LSRs (*Label Switching Routers*), na nomenclatura MPLS, encaminham os pacotes a partir da leitura e análise do rótulo identificado. Cabe aos LSRs de borda de saída do domínio MPLS a retirada do referido rótulo e a entrega dos pacotes originais ao destino.

É importante ressaltar que o rótulo MPLS pode assumir as mais variadas formas, dependendo do mecanismo de encaminhamento empregado na sua implementação, podendo ser representado por um par VPI / VCI ATM, um DLCI Frame Relay ou mesmo um novo cabeçalho (*shim-header*, como é chamado) acrescentado aos pacotes IP, caso sejam utilizadas tecnologias de nível 2 como a Ethernet ou a PPP (*Point-to-Point Protocol*), por exemplo. A implementação de LSRs a partir de comutadores ATM, realizada pela atualização do plano de controle dos referidos comutadores, tem como objetivo principal preservar o grande investimento inicial feito nessa tecnologia. Sendo essa uma solução temporária, a tendência natural, nos próximos anos, é a substituição gradativa desses equipamentos mistos por LSRs MPLS.

Independente do seu real formato, um rótulo MPLS sempre tem significado local em cada roteador, devendo a associação dos mesmos a FECs ser acordada entre LSRs vizinhos, o que pode ser feito estáticamente ou dinamicamente. Na associação estática, os mapeamentos são configurados manualmente pelo administrador do domínio, em contraste com a associação dinâmica, onde um protocolo de sinalização é empregado com o mesmo propósito. O LDP (*Label Distribution Protocol*) [12] é o protocolo recomendado pelo IETF na troca de associações de rótulos a FECs em provedores MPLS. Nos últimos anos, no

entanto, protocolos originalmente concebidos para outros propósitos têm sido estendidos de forma a permitir a troca de associação de rótulos MPLS, como o BGP [16] e o RSVP [17]. Quando as associações de rótulos a FECs possibilitam a comutação de pacotes entre dois LSRs, pode-se dizer que uma ou mais rotas, chamadas de LSPs (*Label Switching Paths*), encontram-se configuradas entre os mesmos.

Além de permitir um encaminhamento mais eficiente, baseado na comutação por rótulos, o MPLS apresenta outros atrativos que têm justificado o crescente interesse por essa tecnologia. O roteamento baseado em restrições (*constraint-based routing*), por exemplo, normalmente implementado através do mecanismo de roteamento explícito, é uma das facilidades disponibilizada pelo MPLS. Adicionalmente, rotas de contingência (backup LSPs) podem ser configuradas previamente em um domínio MPLS, fornecendo um mecanismo de reroteamento rápido (*fast re-route*) [18] e uma maior confiabilidade às aplicações de missão-crítica, cada vez mais comuns na Internet. O balanceamento de carga, que permite a distribuição de fluxos de tráfego por diferentes rotas de forma a serem evitados pontos de congestionamento, é um dos mecanismos de engenharia de tráfego que também podem ser implementados pela utilização da tecnologia MPLS. Finalmente, a possibilidade de configuração de LSPs sujeitos a requisitos de QoS tornou possível o emprego do MPLS no suporte às novas aplicações multimídia, com a vantagem adicional da facilidade de integração dessa tecnologia com o IP e os modelos Intserv e Diffserv.

2.6.4 Conclusões

A Internet é uma rede em constante evolução, e QoS surgiu com o intuito de suprir as novas necessidades de comunicação da mesma e resolver alguns problemas que impedem o desenvolvimento de algumas aplicações.

Foram descritas as principais propostas de modelos de serviços com qualidade na Internet. A primeira proposta abordada, o modelo intserv, permite um controle mais fino dos recursos do provedor de serviços, a um custo maior em termos de processamento e de memória necessários à manutenção de estados por fluxo. O modelo diffserv tornou possível a gerência de recursos com uma granularidade menor, em termos de agregado de fluxos, permitindo uma maior escalabilidade. Os modelos intserv e diffserv representam arquiteturas complementares. Uma configuração de serviços bastante discutida procura agregar as vantagens de cada um destes modelos, através da aplicação do intserv nos ISPs e do diffserv nos *backbones*.

A tecnologia MPLS surge para o suporte a engenharia de tráfego ao se utilizar recursos de forma eficiente, minimizando-se a ocorrência de gargalos e congestionamentos, amplia-se a quantidade de usuários atendidos, bem como a qualidade do serviço prestado. Durante o levantamento das propostas de diferenciação de serviços, muitos conceitos comuns foram levantados, como as formas de solicitação de serviços e gerenciamento de recursos, incluindo o controle de admissão, a reserva de recursos, e as funções de classificação, policiamento, escalonamento, dentre outras. O estudo destes conceitos é importante na medida que contribui para o entendimento dos modelos propostos, além de facilitar a integração, imprescindível diante da heterogeneidade de soluções presentes na Internet.

Capítulo 3

MPLS e Engenharia de Tráfego

3.1 Introdução

Com o crescimento da Internet, novas aplicações (principalmente do tipo multimídia) foram sendo desenvolvidas para executar sobre a Internet. Exemplos desse tipo de aplicação são aplicações interativas que utilizam vídeo, como filmes ou mesmo *trailers* de filmes; áudio, como transmissão de rádio, etc. Outras aplicações em tempo real, como jogos, também estão sendo executadas através da rede.

No roteamento tradicional em redes IP, que a maioria dos backbones da Internet utiliza, os pacotes seguem o caminho de menor custo até o destino. Algumas rotas podem então ficar sobrecarregadas enquanto outras são subutilizadas. Outro problema é que as buscas nas tabelas de rotas são demoradas pois as entradas não têm tamanho fixo: procura-se o maior prefixo de endereço que coincide com o endereço de destino do pacote (este método é conhecido como *Longest Match*). Além disso, o processo de roteamento,

incluindo as buscas nas tabelas de rotas, é realizado em cada roteador para cada pacote que chega. A simples aquisição de hardware de roteamento mais poderoso não corrige esses problemas.

Multi Protocol Label Switching [5] é um protocolo de comutação de rótulos que trabalha entre as camadas 2 e 3. É um modelo proposto pelo IETF, projetado somente para rodar nos roteadores (formando uma sub-rede transparente para o resto da rede). Sua proposta é melhorar o tráfego nas redes, principalmente na Internet, utilizando uma solução elegante, relativamente simples e que possa também aproveitar e integrar-se com tecnologias já existentes como por exemplo ATM, Frame Relay, Ethernet, OSPF, BGP ou RSVP, aproveitando as melhores características de cada uma delas.

O MPLS provê meios para separar o roteamento da transmissão dos dados em redes IP. Os roteadores que ficam na borda (*Label Edge Routers - LER*) da sub-rede MPLS, recebem os pacotes de redes convencionais e adicionam-lhes rótulos (ou então retiram, se os pacotes estiverem saindo do domínio MPLS). Os rótulos indicarão o caminho a ser seguido, dispensando o roteamento para cada pacote nos roteadores internos à rede (*Label Switching Routers - LSR*), pois este caminho é pré-estabelecido. Estes roteadores internos realizam a "troca de rótulos" (*Label Switching*), ou seja, verificam o rótulo do pacote no momento da chegada e o substituem por outro, encaminhando-o para seu destino. Essa substituição é realizada pela busca exata em uma tabela de rótulos, que são pequenos e de tamanho fixo. Os caminhos (*Label Switched Paths - LSPs*) são estabelecidos apenas uma vez, a não ser que ocorram imprevistos ou por alguma necessidade do próprio protocolo de distribuição de rótulos (*Label Distribution Protocol - LDP*). O estabelecimento se dá pela

troca de informações entre os LSRs e LERs e são realizados antes do início do tráfego de pacotes. Esses pacotes, ainda, são agrupados em classes de encaminhamento equivalentes (*Forwarding Equivalence Classes - FECs*), que devem receber o mesmo tratamento.

3.2 Arquitetura MPLS (*Multi Protocol Label Switching*)

A seguir serão apresentados os principais conceitos e definições dos elementos que compõem a tecnologia MPLS.

3.2.1 Rótulo MPLS

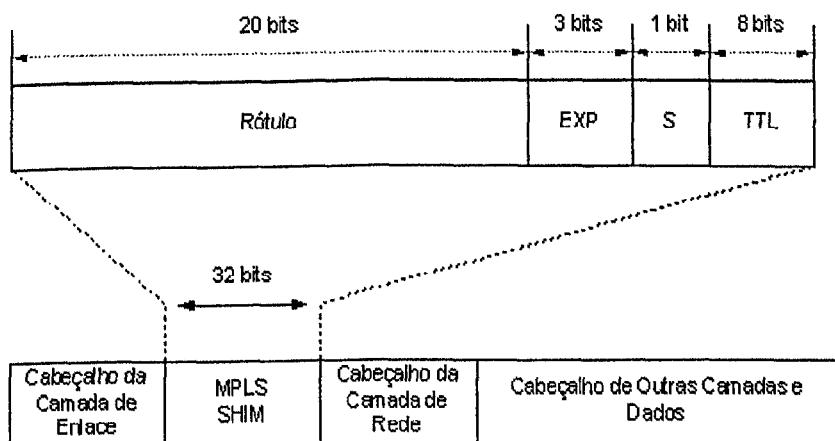


Figura 3.1: Rótulo MPLS.

O rótulo MPLS é um identificador fixo de 32 bits que pode ser inserido no pacote entre os cabeçalhos dos níveis 2 e 3 ou pode ser encapsulado no nível 2 como nos campos

VPI e/ou VCI nas redes ATM, ou no campo DLCI das redes Frame Relay. A Figura 3.1 ilustra o formato do rótulo MPLS.

O campo rótulo tem 20 bits e transporta o valor real do rótulo MPLS. O EXP ou campo experimental é composto de 3 bits e tem um uso significativo no desempenho das aplicações, principalmente em momentos de congestionamento, pois os fluxos de mais alta prioridade recebem melhor tratamento em relação aos de baixa prioridade. O campo S (*Stack*) é de 1 bit cujo objetivo é indicar se o rótulo é o último de uma pilha de rótulos (*label stack*). Por fim o TTL (*Time to Live*) tem 8 bits e fornece uma funcionalidade convencional IP-TTL.

3.2.2 FEC (*Forward Equivalence Class*)

A FEC é um grupo de pacotes que é encaminhado seguindo um mesmo critério. O acréscimo de um rótulo a um pacote significa atribuir este pacote a uma determinada FEC. Cada FEC é especificada como um ou mais elementos de FEC [5]. Há atualmente dois tipos de elementos de FEC definidos para o MPLS: prefixos de endereços e endereços completos (prefixos podem ser também completos e haverá casos em que terão significado diferente de endereços completos). Este conceito é importante, pois como as FECs agrupam pacotes que se adequam a determinado critério, é possível criar FECs para satisfazer requisitos de QoS ou de engenharia de tráfego. No MPLS, as FECs não serão mapeadas diretamente com as interfaces de saída do roteador.

3.2.3 NHLFE (*Next Hop Label Forwarding Entry*)

A NHLFE [5] contém todas as informações que devem ser aplicadas à pilha de rótulos de um pacote. Além do endereço do próximo nó vizinho a quem deve ser encaminhado o pacote, ela contém também operações como: trocar o rótulo do topo da pilha por outro novo, colocar outros rótulos na pilha, remover a pilha de rótulos, como codificar a pilha de rótulos e qualquer informação que sirva para encaminhar corretamente o pacote.

3.2.4 FTN (*FEC to NHLFE*)

A FTN [5] é implementada somente nos roteadores da borda, sendo portanto utilizado apenas para pacotes não-rotulados. Faz o mapeamento de cada FEC em um conjunto de NHLFEs. No caso de duas ou mais NHLFEs serem mapeadas, apenas uma deve ser escolhida.

3.2.5 LSR (*Label Swiching Router*)

Roteador que fica no núcleo da rede. É responsável por fazer a troca, retirada ou inserção de rótulos na pilha mediante o mapeamento na NHLFE do rótulo do topo da pilha. Atua, junto aos LERs, na troca de mensagens para requisição ou envio de rótulos para manutenção de LSPs.

3.2.6 LER (*Label Edge Router*)

É o roteador que fica na borda da rede, sendo responsável por inserir ou remover rótulos dos pacotes, na entrada ou saída da rede. LERs realizam o FTN no momento

da chegada de um pacote ao domínio MPLS. Também podem se conectar com redes de diferentes tipos, já que fazem a fronteira entre o domínio MPLS e as demais. LER é uma definição à parte do padrão MPLS, criado para facilitar a visão do domínio. Eles são, na verdade, LSRs que tem a capacidade de fazer fronteira com outras redes.

3.2.7 LSP (*Label Switched Path*)

O LSP é um caminho unidirecional estabelecido antes do envio de pacotes ao longo dos roteadores, que será seguido por um pacote segundo a análise do rótulo do topo da pilha. Para percorrer o sentido inverso é necessário um novo LSP. A própria estrutura de pilha dos rótulos permite que se tenha vários níveis de LSPs na rede, que são chamados de túneis. Para cada LSP existe um roteador de início e de fim, geralmente LERs mas LSRs também podem ser usados, inclusive para o estabelecimento de túneis.

3.3 Distribuição de Rótulos

Os rótulos são distribuídos entre os LSRs quando do estabelecimento de LSPs e ainda um rótulo tem sempre uma FEC associada. A distribuição de rótulos é chamada de Distribuição a Jusante (*Downstream*) que especifica dado um sentido de um fluxo, qualquer associação FEC-rótulo deve ser criada a jusante (*downstream*) e distribuída a montante (*upstream*). Assim as associações criadas por LSRs são distribuídas somente para seus pares a montante.

Cada LSR mantém uma Base de Informações de Rótulos, ou LIB (*Label Information*

Base). Essa LIB contém pelo menos cinco colunas: a FEC, o rótulo de entrada, a interface de entrada, o rótulo de saída e a interface de saída.

A diferença no processamento de rótulos entre os LSRs vai depender de sua condição de ingresso, egresso e núcleo. Os LSRs de ingresso são responsáveis por rotular pacotes quando estes casam com uma determinada FEC. Esta FEC determina uma entrada na LIB que estipula o rótulo que o pacote irá receber, bem como a interface de saída. Nos LSRs de núcleo, o datagrama será encaminhado exclusivamente com base no rótulo. Ao receber um datagrama rotulado em uma determinada interface, o par rótulo-interface determina numa entrada na LIB. O pacote então tem seu rótulo comutado e encaminhado à interface de saída. No LSR de egresso, o rótulo é removido e o pacote passa a ser roteado de forma convencional (*hop-by-hop*) pela camada de rede.

Basicamente tem-se duas maneiras para a distribuição de rótulos em uma rede MPLS. Criar um novo protocolo de distribuição de rótulos ou adicionar essa capacidade a um protocolo já existente.

A arquitetura MPLS não impõe uma estratégia específica para a distribuição de rótulos, ela prevê a existência de diferentes estratégias e protocolos para esta finalidade [19]. Uma solução possível é dotar o protocolo RSVP (*Resource Reservation Protocol*) para a distribuição de rótulos [11]. O RSVP tem sua origem na arquitetura IntServ. A segunda solução é encontrada no protocolo LDP (*Label Distribution Protocol*) [12]. O LDP foi desenvolvido exclusivamente para a distribuição de rótulos em redes MPLS. A diferença fundamental entre o RSVP e LDP é que o RSVP mantém sua característica *soft state*, enquanto o LDP emprega *hard state* [20].

Extensões a esses dois protocolos citados acima foram desenvolvidas principalmente para suportar a engenharia de tráfego e o roteamento baseado em restrições. O CR-LDP (*Constrained-based Routing LDP*) e RSVP-TE (*RSVP Traffic Engineering*).

3.3.1 Estratégias de distribuição de rótulos

Na arquitetura MPLS tem-se duas estratégias de distribuição de associações rótulo FEC: Sob Demanda (*Downstream On-demand*) onde um LSR somente associa e distribui um rótulo a uma FEC quando explicitamente requisitado por um LSR para a montante, e Sem Solicitação (*Unsolicited Downstream*) onde um LSR associa um rótulo a uma FEC e distribui a um subconjunto de seus LSRs pares a montante sem que estes o tenha solicitado [19].

3.3.2 Controle da distribuição de associações

A distribuição de associações se dá via controle ordenado ou independente. No controle independente, o LSR pode distribuir associações no momento que desejar. No controle ordenado, o LSR distribui uma associação para o LSR a montante apenas quando já possui uma associação de um rótulo para a mesma FEC suprida pelo LSR a jusante, ou quando for o LSR de egresso para a FEC.

3.3.3 Modo de retenção

O modo de retenção das associações pode ser conservativo ou liberal, sendo aplicado somente quando a distribuição de rótulos for Sem Solicitação. No modo conservativo, o

LSR mantém apenas as associações rótulo-FEC recebidas de um LSR a jusante quando este LSR é o próximo salto para a FEC. Se uma associação não satisfazer essa condição, é descartada. No modo liberal, a associação é mantida para que se por acaso o LSR a jusante torna o próximo salto.

3.3.4 Protocolo LDP - (*Label Distribution Protocol*)

O LDP é um protocolo desenvolvido para a distribuição de rótulos entre LSRs [12]. A troca de associações de rótulos a FECs é feita através do estabelecimento de uma sessão entre LSRs vizinhos. O protocolo LDP é bidirecional, sendo assim, uma única sessão é estabelecida para a troca de associações entre os LSR pares.

Cada LSR deve iniciar um processo de determinação dos seus LSR vizinhos, que é realizado através de troca de mensagens de descoberta (*discovery messages*). Depois, através de mensagens de sessão (*session messages*) é estabelecida uma sessão LDP entre os LSRs vizinhos. Uma vez estabelecida uma sessão LDP, associações bidirecionais podem trocar informações através das mensagens de anúncio (*advertisement messages*). O protocolo, através das mensagens de notificação (*notification message*), pode trocar mensagens específicas, informando condições de erro e status da sessão [12].

Periodicamente os LSRs enviam aos seus vizinhos mensagens LDP *hello* de forma a anunciar a sua presença na rede. Ao receber uma mensagem LDP *hello*, o LSR inicia o estabelecimento de uma conexão TCP com o LSR que originou a mensagem. Após o estabelecimento da conexão, parâmetros são trocados por intermédio de mensagens *LDP Initialization*, que inicia uma sessão LDP entre os pares LSRs. Assim, estes LSRs

devem concordar sobre o método de distribuição de rótulos a ser utilizado, o controle de distribuição, o modo de retenção, entre outros parâmetros. Após o estabelecimento de uma sessão LDP, os LSRs podem trocar associações de rótulos a FECs por intermédio das mensagens *Label Request* e *Label Mapping*.

3.3.5 Protocolo CR-LDP (*Constraint-based Routing LDP*)

O CR-LDP adiciona a mensagem *Label Request* a seqüência de LSRs correspondente a rota calculada na origem, permitindo assim o roteamento explícito e baseado em restrições [21]. Dependendo da mensagem de solicitação, o CR-LDP permite a configuração de rotas do tipo estritas (*strict*) ou não estritas (*loose*). A mensagem Label Request contém uma descrição das características do tráfego que será transportado pela rota, permitindo assim a alocação de recursos durante a configuração de um LSP. A especificação dos parâmetros é em termos de taxa de pico (*PDR - Peak Data Rate*), taxa média (*CDR - Committed Data Rate*) e variações da taxa média aceitáveis (*frequency*).

3.3.6 Protocolo RSVP-TE (RSVP - *Traffic Engineering*)

O RSVP-TE [17] é uma extensão ao RSVP para permitir a distribuição de rótulos entre LSRs. O RSVP-TE implementa uma distribuição de rótulos do tipo *downstream-on-demand*, sendo as solicitações realizadas através do objeto *label-request* acrescentado às mensagens RSVP de caminho.

As associações de rótulos a FECs são informadas através do objeto *label*, encapsulado nas mensagens RSVP de reserva. Mensagens de caminho podem conter os objetos *explicit-*

route, que permite ao transmissor informar a seqüência completa de roteadores da rota a ser criada. O objeto *record-route* torna possível ao transmissor obter a seqüência escolhida pelos mecanismos de roteamento.

A Tabela 3.1 compara os protocolos RSVP-TE e CR-LDP [22].

Característica	RSVP-TE	CR-LDP
Aplicação	redes <i>backbone</i>	redes <i>backbone</i>
Tipos de roteamento	<i>strict, loose, pinning</i>	<i>strict, loose, pinned</i>
Escalabilidade	sim	sim
Estado do LSP	<i>soft state</i>	<i>hard state</i>
Refresh no estado do LSP	periodicamente	não é necessário
Re-roteamento de caminhos	sim	sim
Notificação de falhas	não confiável	confiável
Mecanismo de transporte	UDP	TCP
Mensagens exigidas para estabelecer e manter um LSP	<i>path, resv, resvconf</i>	<i>request, mapping</i>

Tabela 3.1: Comparação entre RSVP-TE e CR-LDP.

Para implementação do roteamento explícito e baseado em restrições é necessário o envio pelo remetente da seqüência completa dos roteadores, que é uma característica importante para a engenharia de tráfego.

3.4 Engenharia de Tráfego com MPLS (MPLS-TE)

A engenharia de tráfego ou *traffic engineering* (TE) pode ser definida como sendo a tarefa de realizar o mapeamento de fluxos de tráfego em uma infra-estrutura física de transporte de modo a atender critérios definidos pela operação da rede [23].

A engenharia de tráfego destaca-se pela possibilidade de balancear o tráfego de modo a ocupar os vários enlaces e elementos de rede (comutadores e roteadores), de forma otimizada, buscando utilizar os recursos da rede eficientemente, de forma que não ocorra a super ou subutilização destes componentes, ilustrados nas Figuras 3.2 e 3.3.

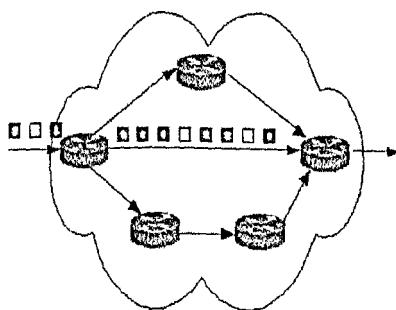


Figura 3.2: Encaminhamento de pacotes sem TE.

A engenharia de tráfego [6], é responsável pela otimização de desempenho no nível operacional e seu maior objetivo é melhorar a eficiência e a confiabilidade das operações de rede, e ao mesmo tempo, otimiza a utilização dos recursos da rede e desempenho de tráfego.

A engenharia de tráfego lida com aspectos de avaliação e otimização de performance de uma rede através do uso de mecanismos científicos de medição, caracterização, modelagem

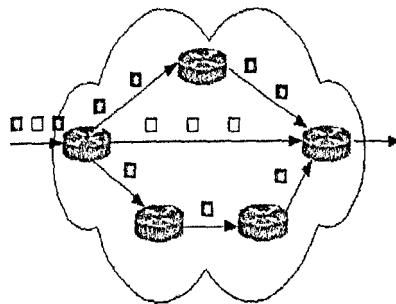


Figura 3.3: Encaminhamento de pacotes com TE.

e controle do tráfego, para desta forma, obter um uso eficiente dos recursos aliado a uma maior confiabilidade [6]. Os objetivos de performance associados à engenharia de tráfego podem ser classificados em:

- orientado a recursos: procuram otimizar a utilização dos recursos do provedor de serviços;
- orientado ao tráfego: visam melhorar a qualidade de serviço fornecida aos fluxos de tráfego.

3.4.1 Túneis LSP

Em [6] são definidos os requisitos para a realização de engenharia de tráfego sobre MPLS e identificam os principais componentes e as respectivas funcionalidades. A questão essencial envolve a criação de túneis LSPs através da engenharia de tráfego (LSP-TE *tunnels*) na rede para o encaminhamento do tráfego IP.

A rota a ser seguida por um túnel é determinada de forma estática ou através de

roteamento baseado em restrições realizado pelos roteadores de borda.

Para fazer com que o processo de Engenharia de Tráfego seja automático usa-se o ”*QoS Routing*” ou Roteamento baseado em QoS, ou mais usualmente, o roteamento baseado em restrições (CBR).

Um aspecto muito importante da engenharia de tráfego em MPLS é a diferença entre troncos de tráfegos, fluxos de tráfego e LSPs. Um LSP pode ser definido como um caminho virtual do nó de ingresso até o nó de egresso do domínio MPLS seguido por pacotes com o mesmo rótulo sendo unidirecional.

Um tronco de tráfego é uma agregação de fluxos de tráfego de mesma classe, a qual é colocada dentro de um LSP. Por essa razão, todos os pacotes sob um tronco de tráfego tem o mesmo rótulo e os mesmos 3 bits do campo de CoS (*Class of Service*) no cabeçalho MPLS. Estes troncos podem ser estabelecidos estaticamente ou dinamicamente (sob demanda) entre quaisquer dois nós em um domínio MPLS.

A engenharia de tráfego MPLS interessa em mapear os troncos de tráfego dentro de enlaces físicos através dos LSPs.

Uma rede habilitada com a tecnologia MPLS empresta, por si mesma, operações de engenharia de tráfego porque:

- os rótulos não estão restritos ao encaminhamento IP convencional ditado pelos protocolos de roteamento baseados em IP convencional;
- os troncos de tráfego podem ser mapeados dentro de LSPs;
- os atributos podem ser associados com os troncos de tráfegos;

- o encaminhamento IP convencional permite somente agregação de endereço, enquanto o MPLS permite agregação ou desagregação.

3.4.2 Troncos de tráfego

O conceito de troncos de tráfego é bastante utilizado em MPLS. Eles são similares aos circuitos virtuais das redes ATM e Frame Relay. A Figura 3.4 ilustra a relação entre fluxos, troncos, LSPs e enlaces.

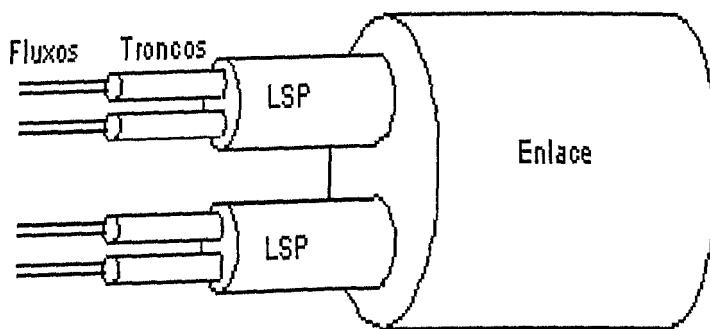


Figura 3.4: Relação entre fluxos, troncos, LSPs e enlaces.

Considerando como um grafo MPLS a malha constituída de um conjunto de roteadores de comutação por rótulos (LSRs), que formam os nodos do grafo e um conjunto de LSPs que fornecem a conectividade lógica ponto a ponto entre os LSRs, o problema fundamental da engenharia de tráfego MPLS é mapear troncos de tráfego dentro da topologia física da rede através de LSPs.

Em muitas situações, faz sentido distribuir o tráfego através de troncos de tráfego paralelos, ou, em algumas situações, através de caminhos em LSRs na rede física. No

primeiro caso, múltiplos troncos são estabelecidos entre os LSRs, permitindo cada tronco transportar uma porção da carga do tráfego total.

A segunda operação, que é distribuir a carga de tráfego sob diversos troncos e diferentes nós na rede, é um problema mais complexo.

A engenharia de tráfego MPLS estabelece as seguintes características para um tronco de tráfego [24]:

- De acordo com a idéia de FEC, um tronco de tráfego é uma agregação de fluxos de tráfego pertencente a uma mesma classe. Por exemplo, pode ser desejável colocar diferentes classes de tráfego dentro de uma FEC se a granularidade de tráfego não for necessária;
- Um tronco de tráfego é capaz de "encapsular" uma FEC entre qualquer LSR de ingresso e um LSR de saída;
- Troncos de tráfego são roteáveis através do rótulo da FEC;
- Um tronco de tráfego pode ser movido de um caminho para outro, o que significa que ele é distinto do LSP através do qual ele trafega.

Um tronco de tráfego é unidirecional, mas na prática, dois troncos com sentidos opostos podem ser associados, enquanto eles são estabelecidos e desativados em conjunto. Esta associação é denominada de BTT (*Bidirectional Traffic Trunk*). O BTT simétrico é aquele onde os troncos de tráfegos estão no mesmo caminho e BTTs assimétricos são roteados através de diferentes caminhos.

Para se atingir uma engenharia de tráfego sobre MPLS em redes IP que seja bastante eficiente deve-se ter atributos associados aos troncos de tráfego, que restringem o uso dos recursos existentes aos troncos de tráfego e ao roteamento.

Estes atributos podem ser modificados através de ações do operador da rede ou através de agentes automatizados que direcionam a rede a um estado desejado.

Um tronco de tráfego possui parâmetros que lhe são fornecidos identificando seus atributos. Por sua vez, estes atributos influenciam suas características de comportamento, isto é, como o tráfego será tratado pela rede. Os valores dos atributos são fornecidos pelo administrador da rede, que examina, automaticamente, os critérios da FEC (endereço, número da porta, identificação do protocolo) e estabelece os parâmetros.

3.4.3 Congestionamento

O congestionamento é um dos principais problemas enfrentados pelos provedores de serviço na Internet atualmente. Os mecanismos de roteamento tradicionais, em uso pela maioria dos provedores, baseiam o encaminhamento dos fluxos de tráfego no princípio do menor caminho ou SPF (*Shortest Path First*). Assim a forma de encaminhamento, embora seja eficiente em termos de economia no uso de recursos, contribui para o aumento do congestionamento nas redes [6].

Os menores caminhos, calculados por fontes de dados distintas, tendem a se sobrepor em determinados enlaces, em especial naqueles localizados no backbone da rede, criando gargalos de tráfego nestes pontos. Ao mesmo tempo, caminhos mais longos tendem a ser sub-utilizados contribuindo para o não balanceamento da rede.

A engenharia de tráfego procura contornar o problema do congestionamento através da medição contínua do tráfego em vários pontos da rede. A partir das informações obtidas, fluxos de pacotes podem ser redirecionados para caminhos menos congestionados. A escala de tempo utilizada nestas medições pode variar bastante [4]. Normalmente, medições de médio e longo prazo permitem ao administrador da rede antecipar-se a situações de congestionamento, tornando possível a adoção de medidas preventivas, como o aumento da capacidade instalada, por exemplo. Muitas vezes na matriz de tráfego ou nos requisitos de QoS dos usuários podem não ser facilmente prevista, obrigando o provedor a uma ação em uma escala de tempo menor. Informações de uso dos recursos devem ser distribuídas continuamente entre os elementos da rede assim que seja possível adaptação instantânea no estado geral do provedor. Estas informações serão reunidas em uma base de conhecimento a ser utilizada pelos mecanismos de engenharia de tráfego na tomada de decisões.

3.4.4 Confiabilidade

Com o crescimento da Internet aplicações que exigem confiabilidade na transmissão estão sendo desenvolvidas e colocadas em operação. O serviço oferecido pelos provedores a estas aplicações, pela própria natureza das mesmas, deve ser ininterrupto. Por outro lado a velocidade dos enlaces utilizados na Internet tem crescido continuamente, em especial naqueles disponibilizados pelos provedores de backbone. O uso de enlaces de maior velocidade possibilita o transporte de uma maior quantidade de fluxos. Mas, falhas nestes enlaces terão maior repercussão devido ao maior número de fluxos comprometidos.

Técnicas de roteamento dinâmico foram introduzidas objetivando o aumento da con-

fiabilidade na presença de falhas na rede. O roteamento dinâmico envolve a detecção de falhas e o redirecionamento do tráfego para rotas alternativas. Estes mecanismos são normalmente lentos, podendo consumir minutos desde a detecção da falha à propagação de informações de estado e recálculo das rotas, inaceitável pelas aplicações em tempo real. Uma técnica que procura diminuir o tempo de recuperação de uma rede na presença de falhas é o reroteamento rápido (*fast-reroute*) [18].

3.4.5 Roteamento baseado em Restrições (*CBR - Constraint-based Routing*)

Novos modelos em desenvolvimentos na área de qualidade de serviço, permitem aos usuários especificar os requisitos de QoS que devem ser aplicados aos seus fluxos de dados. É de responsabilidade do provedor identificar e reservar os recursos necessários ao transporte destes fluxos, assim como garantir a qualidade de serviço solicitada. A identificação destes recursos caberá muitas vezes aos mecanismos de roteamento, uma vez que são os responsáveis pela determinação do caminho por onde os fluxos serão transportados de forma a serem entregues no destino.

Os mecanismos de roteamento em uso são baseados em técnicas que procuram minimizar o custo total de encaminhamento através da escolha das rotas de menor caminho. Muitas vezes estas rotas de menor caminho podem não permitir o compartilhamento de outros fluxos, podendo ser comprometida a qualidade de serviço dos fluxos previamente aceitos. Desta forma, o provedor deve fornecer outras rotas mesmo a um custo maior, para que possam atender a qualidade de serviço solicitada.

Roteamento com QoS (*QoS routing*) [25] é o mecanismo de roteamento onde o caminho a ser seguido por um fluxo é determinado a partir de algum conhecimento sobre a disponibilidade de recursos na rede, bem como os requisitos de QoS da aplicação.

A partir do conceito de roteamento com QoS, deriva-se outro mais abrangente, o roteamento baseado em restrições (*CBR - Constraint-based Routing*) [6]. O roteamento baseado em restrições leva em consideração não somente os requisitos de QoS, mas também outras restrições à rota a ser escolhida, como a interface por onde o fluxo foi recebido, a prioridade relativa associada ao fluxo, as classes de roteadores que compõem o caminho, além de outras considerações administrativas.

O problema do cálculo de rotas a partir de restrições é considerado intratável [6]. Mas soluções sub-ótimas podem ser obtidas por meio de heurísticas [26]. Por ser de processamento intenso, este cálculo é normalmente realizado uma única vez, pelo roteador de borda que recebe o tráfego do usuário. Este roteador deve informar a rota calculada aos demais, o que pode ser feito através do mecanismo de roteamento explícito, ou seja, o envio juntamente com os dados da seqüência de roteadores por onde o tráfego deve ser encaminhado.

3.4.6 MPLS - TE utilizando CBR

Utilizando o MPLS torna relativamente simples implementar o roteamento baseado em restrições ou CBR. Os troncos de tráfego podem ser mapeados em LSPs. O projeto e a arquitetura da rede ficam mais simples e com menor custo.

Outra vantagem é que o MPLS permite tanto a agregação quanto a desagregação de

tráfego, facilitando a engenharia de tráfego.

Os protocolos de roteamento IGP (*Internet Gateway Protocols*) utilizam o algoritmo SPF (*Shortest Path First*), buscando a seleção da rota de menor custo, o que conserva os recursos da rede, porém as rotas de diferentes origens podem se sobrepor em alguns enlaces, causando congestionamento nos mesmos.

O uso do CBR melhora a engenharia de tráfego pois permite a seleção de rotas distintas obtidas pelo SPF baseadas nas restrições definidas, além de otimizar os recursos e o desempenho da rede em um só algoritmo. Para que seja possível implementar a engenharia de tráfego na rede MPLS, é necessário:

- um conjunto de atributos associados aos troncos de tráfego que especifiquem o seu comportamento, e de atributos ligados aos LSPs por onde estes troncos vão trafegar;
- um ambiente para a implementação de CBR que selecione os caminhos para os troncos de tráfego.

3.5 Re-roteamento rápido MPLS (*MPLS fast-reroute*)

O re-roteamento rápido permite um caminho danificado seja temporariamente substituído por outro secundário [18]. A reserva de recursos para uso específico na presença de falhas também é outro mecanismo empregado pelos algoritmos de recuperação. A engenharia de tráfego procura utilizar uma combinação desta e de outras técnicas objetivando o aumento da confiabilidade dos serviços oferecidos pelo provedor.

Em [6] é descrito o atributo de resiliência como o comportamento de um tronco de

tráfego em caso de falha. Um atributo de resiliência básico indica o procedimento de recuperação a ser aplicado aos troncos de tráfego cujos caminhos sofreram falha. O que permite essa recuperação é o re-roteamento de LSP.

O re-roteamento calcula um novo caminho para um LSP depois que seu caminho existente é interrompido. Entretanto, durante o tempo exigido para realizar esse re-roteamento básico, pode haver perda de tráfego significativa.

Quando um enlace ou nó falha, essa falha é sinalizada e um novo LSP é estabelecido para o tráfego. Embora alguns segundos de perda geralmente sejam aceitáveis para o tráfego de dados, as aplicações de tempo real, como voz e vídeo podem não ser compatíveis.

A proteção, no contexto da restauração rápida, é ter procedimentos preparados que, quando aplicados a recursos selecionados, garante a perda mínima de tráfego na falha. A proteção pode ser concentrada na proteção de recursos lógicos (LSP) a partir de falhas físicas (enlaces ou nós) [27].

A seguir apresenta-se alguns métodos de re-roteamento rápido MPLS.

3.5.1 Método L3-*Control Driven*

Este método utiliza o Protocolo de Distribuição de Rótulos (LDP) através de uma mensagem de controle para fazer o roteamento na rede.

Com a utilização do L3 *Control Driven*, cada LSR, incluindo os roteadores de borda e possíveis *hosts*, podem trocar mensagens de controle com seus vizinhos para compartilhar informações entre o rótulo configurado e o caminho especificado pelo rótulo. Inicialmente,

o roteador de borda transmite a mensagem de controle MPLS ao seu vizinho para estabelecer o LSP. As mensagens de controle levam a informação que especifica o caminho (prefixo de endereço destino L3) de criação do LSP e também o rótulo a ser utilizado [28].

Mudanças dos caminhos para os LSPs primários em resposta às mensagens (L3) de trocas de rotas são iniciadas pelo LSR que detecta a troca de rota para o estabelecimento inicial da nova rota.

O LSR que detecta a troca de rota, invalida o mapeamento entre o rótulo e a rota para seu vizinho *downstream*, trocando mensagens de controle com ele. O LSP antigo será liberado por *timeout*, porque nenhum tráfego de dados foi emitido por este caminho. A criação de um novo caminho pelo LSR que detectou a troca de rota será transportada de uma maneira distribuída, similar ao procedimento de estabelecimento inicial de um LSP [28].

3.5.2 Método proposto por Haskin

O método proposto por Haskin consiste em pré-configurar os ER-LSPs antes mesmo das fontes iniciarem o envio de tráfegos. É configurado um LSP alternativo para dar suporte a um re-roteamento rápido de pacotes de dados, quando da ocorrência de uma falha no LSP primário em redes MPLS [29]. Este método traz alguns benefícios: (a) A complexidade da computação de um novo caminho é reduzida. Apenas um único caminho adicional entre os roteadores de origem e destino deve ser calculado; (b) A sinalização do estabelecimento do LSP é minimizada. Com uma pequena extensão para LDP, um único roteador na entrada do caminho protegido pode iniciar a alocação de

rótulos para os LSPs primário e alternativo; (c) A presença de tráfego no LSP alternativo, transitando em direção inversa ao LSP primário, pode ser usada como indicador de falha ou congestionamento de um enlace *downstream* ao longo do LSP primário. Logo ao verificar o fluxo de tráfego reverso ao caminho primário, o roteador pára de enviar tráfego *downstream* no LSP primário e inicia imediatamente o re-roteamento do tráfego de dados para o LSP alternativo.

3.6 Conclusões

O MPLS se destaca como uma arquitetura emergente, que representa o esforço de padronização do modelo de encaminhamento de pacotes baseados em rótulos e que já existe comercialmente, produzido por uma série de fabricantes. Este esforço permitirá a interoperabilidade e compatibilidade necessárias para compor os *backbones* atuais e da Internet2.

O MPLS torna possível a implementação eficiente do mecanismo de roteamento explícito, que é empregado na configuração de rotas baseadas em restrições. Adicionalmente, o uso de LSPs possibilita, a um domínio MPLS, implementar, virtualmente, qualquer topologia lógica. Também são possíveis alterações dinâmicas nestas topologias, através da criação e eliminação de LSPs em tempo de operação, viabilizando a implementação de mecanismos que venham a melhorar a distribuição dos fluxos de tráfego pela rede, minimizando situações de congestionamento e, consequentemente, fornecendo um serviço de melhor qualidade ao usuário final.

Um dos principais desafios da engenharia de tráfego, consiste no mapeamento otimizado dos fluxos de tráfego dos usuários aos recursos disponíveis no provedor. Em um domínio MPLS, este mapeamento é realizado através da inserção dos referidos fluxos nos LSPs definidos no provedor. A associação entre fluxos de tráfego a LSPs é realizada pelos LSRs de borda através da adição de rótulos aos pacotes que compõem os referidos fluxos.

O protocolo CR-LDP (*Constraint Routed Label Distribution Protocol*) fornece uma infraestrutura escalável para satisfazer as exigências de engenharia de tráfego em um ambiente de rede de comunicação.

A manutenção de LSPs de contingência (backup LSPs) contribuem para o aumento da confiabilidade fornecida pelo domínio MPLS. A pré-configuração destes LSPs torna possível a implementação de mecanismos de recuperação bastante eficientes.

Com a possibilidade de estabelecer caminhos explícitos, o MPLS torna-se uma promissora ferramenta para implementar a engenharia de tráfego. A engenharia de tráfego nas redes MPLS melhora, efetivamente, o desempenho e escalabilidade dos ISPs e das redes *backbone*, além de facilitar o suporte de novos serviços, tais como as aplicações em tempo real.

Capítulo 4

DiffServ e MPLS

4.1 Introdução

A arquitetura de Serviços Diferenciados [3] posiciona-se entre os extremos do melhor esforço (BE - sem priorização) e dos Serviços Integrados [2], com reserva de recursos e grande sobrecarga de sinalização. Na arquitetura DiffServ não existe alocação explícita de recursos e não é feita sinalização, tendo em vista que a prioridade do pacote é transmitida no cabeçalho IP, permitindo desta forma maior escalabilidade e baixa sobrecarga de sinalização.

O Diffserv define o conceito de domínio DS, que é um conjunto contíguo de nós DS que aplicam um conjunto comum de políticas sobre o tráfego que atravessa o domínio. Um domínio DS tem nós de borda e nós de interior [3]. Os nós DS de borda são responsáveis pela classificação e condicionamento do tráfego que entra no domínio DS. Para cada fluxo de tráfego entrando no domínio pelos nós de borda, a política de QoS define qual terá um

serviço diferenciado, como este deverá ser marcado nos nós de borda e como será tratado pelos nós interiores. Estes, por sua vez, examinam a marcação dos pacotes e atuam de acordo com as políticas definidas ou seu perfil de tráfego.

A arquitetura DiffServ ainda opera com os protocolos de roteamento IP convencionais. Para a obtenção de qualidade de serviço fim-a-fim são necessários protocolos de roteamento que suportem engenharia de tráfego para um melhor aproveitamento da malha de enlaces. A arquitetura mais adequada atualmente para prover engenharia de tráfego é baseada na tecnologia MPLS.

A engenharia de tráfego MPLS é fundamental quando se deseja obter qualidade de serviço fim-a-fim em redes IP, permitindo uma utilização mais eficiente da rede e uma rápida correção em falhas de enlaces e nós. Esta tecnologia utiliza rótulos para encaminhar os pacotes através de seus domínios.

No roteador de entrada da rede os rótulos são associados aos pacotes baseados em Classes de Equivalência de Encaminhamento (*Forwarding Equivalence Classes - FEC*), e a complexidade fica a cargo da borda da rede, sendo que os roteadores do núcleo apenas realizam a troca dos rótulos para encaminhamento. Estas FECs podem ser associadas às classes de tráfego do DiffServ. O MPLS permite um roteamento explícito, baseado nos endereços de origem e destino, permitindo a introdução de novos serviços IP, como é o caso do DiffServ.

Dante disso surge a necessidade da integração de duas arquiteturas para resolver o problema citado anteriormente. A arquitetura DiffServ fica responsável pela diferenciação dos serviços e a arquitetura MPLS para prover engenharia de tráfego, assim, justifica-se

a integração MPLS - DiffServ.

Neste capítulo apresenta-se os conceitos da arquitetura DiffServ e a integração com a arquitetura MPLS.

4.2 Arquitetura Diffserv

A arquitetura de Serviços Diferenciados é baseada em um modelo, onde o tráfego que entra na rede é classificado e possivelmente condicionado na entrada da rede e associado a diferentes agregações de fluxos. Cada agregação é identificada por um código marcado no cabeçalho dos pacotes, sendo que, no núcleo da rede, os pacotes são encaminhados de acordo com o código estabelecido [3].

Um domínio DS é formado por um conjunto de nós que implementam a diferenciação de serviços e que utilizam uma mesma política de provisão de recursos. Os limites do domínio DS são constituídos por roteadores de borda e por roteadores de núcleo. Os roteadores de borda são responsáveis pela classificação dos pacotes em agregações de fluxos e pela sua adequação às condições pré-estabelecidas entre o cliente e o provedor de serviços (*Internet Service Provider - ISP*). Os roteadores no interior do domínio DS, denominados roteadores de núcleo, determinam o encaminhamento dos pacotes de acordo com a classificação estabelecida. Os roteadores de borda recebem ainda a denominação de roteadores de ingresso ou egresso, dependendo do sentido do tráfego tratado.

A identificação das agregações de fluxos no interior de um domínio DS é feita através da marcação de um novo campo denominado DS (*Differentiated Services*) no cabeçalho

de cada pacote IP. No IPv4 há um mapeamento do campo ToS (*Type of Service*) e no IPv6 o TC (*Traffic Class*) [30]. Seis bits do campo DS formam o subcampo DSCP (*Differentiated Services Code Point*), que identifica a agregação de fluxos. Os dois outros bits estão reservados para uso futuro. Em cada roteador compatível com a proposta de diferenciação de serviços, o código (*codepoint*) contido no subcampo DSCP é mapeado em um comportamento por nó (*Per-Hop Behavior - PHB*) que define o tratamento a ser recebido pelo pacote naquele roteador para o seu encaminhamento na rede. Existem duas propostas de PHB para a implementação de Serviços Diferenciados: o Encaminhamento Expresso (*Expedited Forwarding - EF*) [7] e o Encaminhamento Assegurado (*Assured Forwarding - AF*) [8].

4.3 A classificação e o condicionamento do tráfego

A arquitetura DiffServ define dois importantes componentes nos nós DS, os componentes de classificação e de condicionamento de tráfego. Estas funções são mais complexas em nós DS de borda que em nós de interior, mas ambos os nós têm um classificador. A Figura 4.1 ilustra os elementos DiffServ.

Existem dois tipos de classificadores: um que classifica o fluxo baseado apenas na classificação DS e outro que verifica múltiplos campos no cabeçalho IP. Estes classificadores são conhecidos como classificador de comportamento agregado (*BA - Behaviour Aggregate*) e classificador multi-campo (*MF - multifield classifier*), respectivamente. A marca de classificação DS é conhecida como DS *codepoint* ou DSCP.

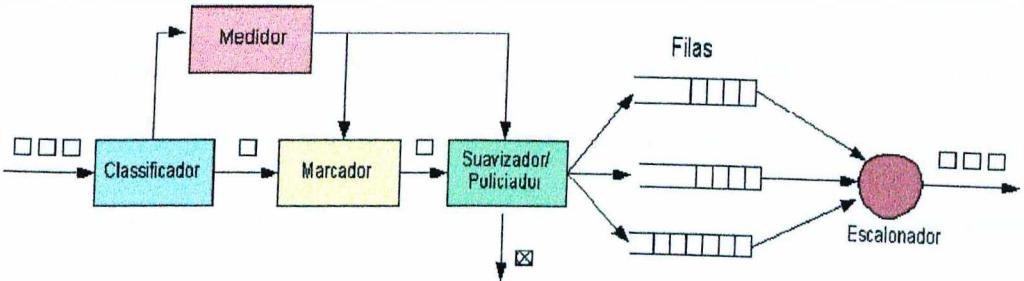


Figura 4.1: Elementos DiffServ.

4.3.1 Classificador

Em nós DiffServ, o classificador é o componente que divide o fluxo de entrada em um conjunto de fluxos de saída por meio de filtros de tráfego baseados no conteúdo do cabeçalho do pacote e/ou em diferentes atributos do pacote que podem ser implicitamente derivados [3]. Existem dois tipos de classificadores.

O classificador BA (ou de agregação por comportamento) é uma agregação de fluxos de pacotes, em grandes conjuntos, que recebem o mesmo comportamento. Logo que adentram ao domínio DiffServ, os fluxos são classificados em diferentes BAs. O BA classifica os pacotes com base somente no conteúdo do campo DSCP, ele ocorre quando o domínio antecessor é compatível com o DiffServ e os pacotes já vêm marcados. Quando o domínio antecessor não está habilitado para enviar os pacotes com o campo DSCP previamente marcado, o classificador avaliará outros campos dos pacotes.

Quando isso ocorre, ele é chamado de classificador MF (*Multi-Field*). Em ambos os casos, o resultado da classificação é o enquadramento do pacote em um BA válido no

domínio e o seu encaminhamento para um processamento posterior. Por exemplo, se o classificador detectar que o pacote está usando o serviço de melhor esforço, ele provavelmente será encaminhado normalmente sem nenhum processamento adicional. Caso o pacote pertença a um BA para o qual foi definido um perfil de tráfego, ele é, geralmente, encaminhado para a fase de medição.

4.3.2 Medidor

De acordo com a chegada do pacote o medidor verifica-se o pacote está de acordo com um perfil de tráfego pré-definido. Várias ações podem ser executadas, através dos elementos de ações. Três tipos de ações, e a combinação delas, podem ser executadas: marcação, formatação e descarte.

4.3.3 Escalonamento

Escalonamento é o modo como os pacotes enfileirados são selecionados para a transmissão pelo enlace.

4.3.4 Policiamento

A ação de regulação regula a taxa com a qual se permite que um fluxo injete pacotes na rede.

4.3.5 Condicionadores

Os condicionadores de tráfego são empregados em um determinado estágio do caminho dos dados para forçar uma determinada política. Eles podem ser implementados através da combinação de um ou mais componentes de Diffserv definidos anteriormente (classificadores, medidores, elementos de ação e filas) ou, alternativamente, através da combinação de condicionadores de tráfego existentes. Marcação é um exemplo de condicionamento de tráfego.

4.4 Encaminhamento do tráfego

Nos nós de borda, o fluxo de tráfego é classificado e marcado. Os campos DSCP (*Differentiated Services Code Point*) [3] são mapeados para os PHBs (*Per Hop Behaviors*) [7] [8] definidos na arquitetura DiffServ. Os PHBs definem o comportamento de encaminhamento de um pacote em um nó DiffServ. PHBs são identificados através de um *label* de 6 bits, do campo TOS (*Type Of Service*) do cabeçalho do pacote IPv4 e o campo *Class* do cabeçalho do pacote IPv6 agora chamados de *Differentiated Services Code Point* (DSCP) conforme mostra a Figura 4.2 que é colocado no campo Diffserv do cabeçalho do pacote IP.

Este campo é tratado como um índice em uma tabela usada para selecionar o mecanismo de manipulação de pacotes implementado em cada dispositivo. Este campo é definido como um campo não estruturado para facilitar a definição de futuros PHBs. Outros comportamentos podem especificar que, para determinados pacotes, serão dadas certas prioridades relativas a outros, em termos de vazão média (*throughput*) ou de pre-

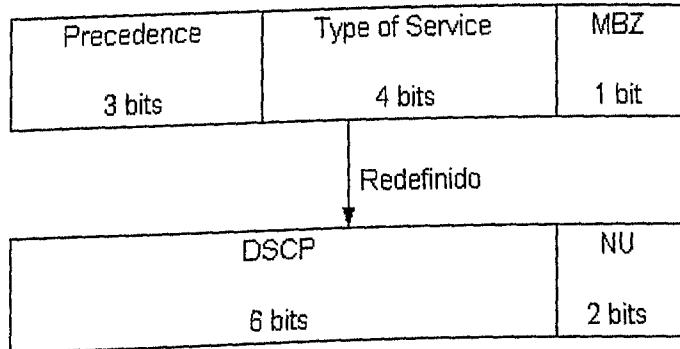


Figura 4.2: Campo ToS e byte DS.

ferência para descarte, mas sem ênfase em atrasos.

Os PHBs são implementados utilizando mecanismos de enfileiramento. PHBs são comportamentos individuais aplicados em cada roteador, por isso isoladamente não garantem QoS fim-a-fim. Entretanto, a interligação de roteadores com os mesmos PHBs e a limitação da taxa em que os pacotes são enviados para um PHB, possibilita o uso de PHBs para construir QoS de fim-a-fim. Por exemplo, a concatenação de EF PHBs ao longo de uma rota preestabelecida, com um cuidadoso controle de admissão, pode prover um serviço similar ao de uma linha dedicada, que é satisfatório para voz interativa. Uma outra concatenação de PHBs pode ser utilizada para transmissão de vídeo armazenado, e assim por diante.

O PHB *default* serve para o tráfego BE (*best effort*) ou melhor esforço. Ele assegura compatibilidade com o encaminhamento melhor esforço padrão em todos os roteadores [7] [8].

4.4.1 PHB EF (*Expedited Forwarding*)

O PHB EF (*Expedited Forwarding*), também referido como serviço premium ou de canal dedicado, pode ser usado para tráfego com requisitos de baixa perda, baixo atraso, baixo *jitter* (variação de atraso) e garantia de largura de banda. Estes requisitos são alcançados assegurando-se que os agregados de tráfego encontrem nenhum ou pouco enfileiramento. As implementações devem prover meios de limitar o dano que o tráfego EF pode infligir sobre o outros tipos de tráfegos.

4.4.2 PHB AF (*Assured Forwarding*)

O PHB AF tem por objetivo fornecer entrega de pacotes IP, com largura de banda assegurada, em quatro classes de transmissão, mas não oferece garantias quanto ao atraso. Cada classe tem três precedências de descartes (*Drop Precedence*), as quais são utilizadas para determinar a importância do pacote. Assim, um nó congestionado dá preferência para serem descartados, entre os pacotes de uma mesma classe, aqueles com maiores valores de precedência de descarte.

Os três primeiros bits do DSCP identificam a classe de transmissão, 001, 010, 011 e 100 e os três últimos bits definem a precedência de descarte. 010 para a precedência mais baixa (ou seja, ultimo a ser descartado), 100 para precedência média e 110 para a mais alta precedência de descarte (ou seja, primeiro a ser descartado).

A marcação da precedência de descarte para tráfegos AF pode seguir dois enfoques: marcador com taxa simples [31] e marcador com taxa dupla [32]. Ambos usam um regulador do tipo duplo balde furado (*Dual Leaky Bucket*), sendo que no marcador simples, os

dois baldes são preenchidos na mesma taxa e no marcador com taxa dupla, os baldes são preenchidos com duas taxas diferentes. Ambos os enfoques medem o tráfego e marcam o pacote, dependendo se o fluxo excede a taxa acordada ou contratada CIR (*Committed Information Rate*). Um pacote é marcado como verde, amarelo ou vermelho dependendo em quanto o CIR foi excedido. Um pacote verde tem a menor prioridade de descarte, enquanto que um pacote vermelho, a mais alta. O principal objetivo é fornecer a marcação para algum mecanismo de descarte baseado na precedência de descartes, tal como o mecanismo RED (*random early detect*).

4.5 Serviços e Contratos

Na arquitetura DiffServ o serviço está associado à QoS do ponto de vista da aplicação, permitindo especificar as necessidades desta em termos de largura de banda, atraso, *jitter* e taxa de perdas, e pode ser quantitativo ou qualitativo. No primeiro caso o serviço é especificado através de um conjunto de métricas e valores de referência correspondentes, enquanto que no segundo caso somente uma definição de alto nível é fornecida. Em ambos os casos a implementação de QoS está baseada em uma determinada combinação dos componentes DiffServ. Isto faz a arquitetura DiffServ muito flexível.

Quanto aos serviços oferecidos por um domínio aderente à DS, deve-se notar: serviços DS são todos para tráfego unidirecional apenas e serviços DS são para tráfegos agregados, não fluxos individuais. Em [3] um serviço DS é definido como o tratamento global de um subconjunto do tráfego do cliente dentro de um domínio aderente a DS ou fim-a-fim.

Os clientes podem marcar os campos DS dos pacotes para indicar o serviço desejado, ou estes campos podem ser marcados pelo roteador de borda que liga o cliente à rede, baseado na classificação MF. No ponto de ingresso da rede, os pacotes são classificados, policiados e, possivelmente, atrasados para torná-los aderentes a algum perfil de tráfego pré-definido. As regras de classificação, policiamento e moldagem de tráfego usadas nos roteadores de ingresso da rede são derivadas a partir de um Acordo de Nível de Serviço (*SLA - Service Level Agreement*). O montante de espaço de bufferização necessário para estas operações também é derivado dos SLAs.

4.5.1 SLA (*Service Level Agreement*) e SLS (*Service Level Specification*)

O acordo do nível de serviço (SLA) é uma negociação formal entre duas partes, chamado de garantia do nível de serviço. É um contrato que existe entre o provedor de serviços e o usuário, desenvolvido para criar um entendimento comum sobre os serviços, prioridades, responsabilidades, etc. [33]

O acordo relativos aos parâmetros de tráfego e QoS entre duas partes é especificado em uma sub parte do SLA, chamado SLS ou especificação do nível de serviço. O mesmo especifica como o tráfego do cliente será tratado pelo provedor da rede [34]. Da perspectiva do provedor, métricas individuais têm sido identificadas e reforçadas na rede para prover ao usuários o acordo SLA. Assim, mecanismos de monitoramento podem ser ativados para o controle de rede. Esses parâmetros e tarefas são chamados SLO (*Service Level Objective*) [34].

4.6 A Integração MPLS e DiffServ (MPLS-DS)

MPLS e DiffServ operam em diferentes camadas na pilha de protocolos. DiffServ opera na camada de rede, enquanto o MPLS opera entre a camada de enlace e a camada de rede. Eles não trabalham juntos naturalmente, porque o MPLS foi desenvolvido sem considerações de QoS, mas os dois são técnicas complementares que podem ser implementados em uma Rede IP para soluções de QoS fim a fim [35].

DiffServ é usado por alguns provedores de serviços para melhorar a escalabilidade da rede, suportando múltiplas classes de serviço. Em algumas redes DiffServ, onde procura-se otimizar os recursos na transmissão de dados, o mecanismo DiffServ pode ser complementado pelo mecanismo de engenharia de tráfego MPLS, o qual opera através de agregados todas as classes DiffServ. Neste caso, DiffServ e MPLS-TE (DS-TE) provê os respectivos benefícios [36].

Quando usados em conjunto, o DiffServ provê mecanismo padrão de QoS e MPLS provê técnicas de roteamento aumentando a otimização dos recursos da rede e provendo engenharia de tráfego. Um domínio MPLS usa protocolos de sinalização para estabelecer um LSP para encaminhamento dos dados através de um caminho. O LSR de ingresso ou LER rotula os pacotes, e os LSRs ao longo do LSP encaminham os pacotes para o próximo hop. No DiffServ, o roteador de ingresso classifica os pacotes e então os marca com o DSCP correspondente. Os roteadores intermediários usam o PHB para determinar o tratamento de escalonamento e probabilidade de descarte para cada pacote [37].

4.6.1 MPLS-DS com QoS fim-a-fim

Diante da complexidade da Internet, que é formada por vários Sistemas Autônomos (*Autonomous Systems - ASs*), uma malha complexa de enlaces e roteadores é formada. Diante disso a arquitetura DiffServ sendo implementada em cada AS, provê atualmente a melhor forma de provisão de QoS fim a fim na Internet [9]. A arquitetura DiffServ utiliza algoritmos de roteamento IP convencionais, ou seja, o menor caminho para rotear os pacotes de uma origem até um destino através destes AS. Para uma melhor obtenção dos parâmetros de QoS exigidos por aplicações típicas, é necessário um balanceamento de carga nos roteadores, de forma que otimize as rotas entre um *host* de origem e um *host* de destino. A engenharia de tráfego com o mecanismo MPLS é a melhor proposta para optimizar essas rotas [4] [6] [38].

O mapeamento de fluxos de tráfego sobre uma topologia de rede física existente é chamado de engenharia de tráfego [39]. Para suportar uma taxa de crescimento extremamente rápida e manter uma infra-estrutura confiável para aplicações críticas, uma organização tem que balancear a carga de tráfego nos vários enlaces, roteadores e switches na rede assim que um deles estiver sub-utilizado ou congestionado.

A engenharia de tráfego provida pelo MPLS resolve os problemas acima pela seleção de um caminho menos congestionado que o menor caminho, conduzindo para uma utilização ótima e confiável da infra-estrutura da rede.

Assim, os agregados de fluxos que pertencem a uma mesma classe de serviço, são associados a LSPs tem-se um ambiente de serviços diferenciados sobre o MPLS.

Em um cenário DiffServ - MPLS, os LSRs devem ser capazes de identificar as classes

de serviços diferenciados nos pacotes de forma a ser possível o fornecimento de QoS contratada pelo usuário.

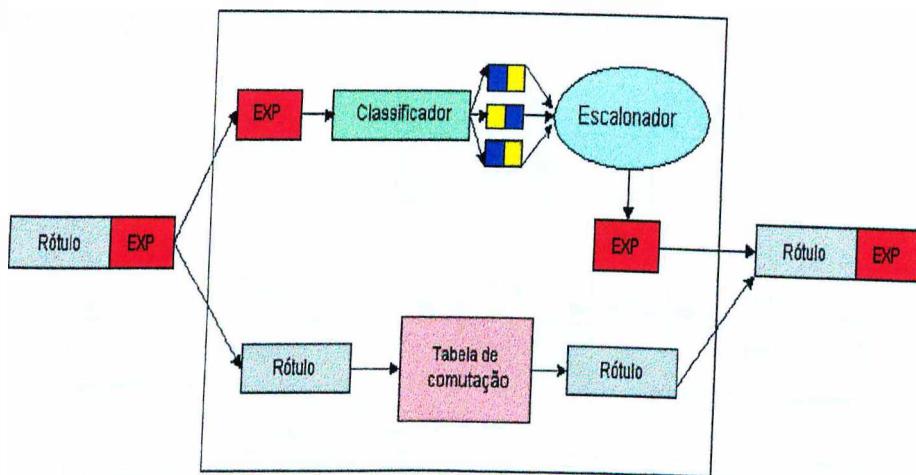


Figura 4.3: Modelo Simplificado de um LSR - DiffServ.

De acordo com a Figura 4.3, o LSR com capacidades DiffServ inicialmente separa o rótulo de entrada e o campo EXP. O rótulo de entrada é utilizado na determinação do novo rótulo e da interface de saída. O campo EXP por sua vez, é interpretado pelo módulo de classificação, que identificará a classe de serviço associada ao fluxo. Cabe ao escalonador implementar o PHB associado à classe, determinando o momento em que os pacotes que compõem o fluxo ser encaminhado. Desta forma, o LSR determina onde enviar cada pacote a partir do rótulo de entrada e quando efetivar o envio com base no valor do campo EXP.

Diante disso existem dois problemas básicos para o MPLS suportar DiffServ ilustrados na Figura 4.4. Primeiro, o DSCP (*DiffServ Code Point*) encontra-se no cabeçalho IP, e os LSRs (*Label Switched Route*) examinam somente o cabeçalho do rótulo. Segundo, o

DSCP é composto de 6 bits e o campo EXP do rótulo MPLS tem somente 3 bits. Existem então duas soluções definidas em [40] para resolver esses problemas: *EXP-Inferred-PSC LSP* (E-LSP) e *Label-Only-Inferred-PSC LSP* (L-LSP).

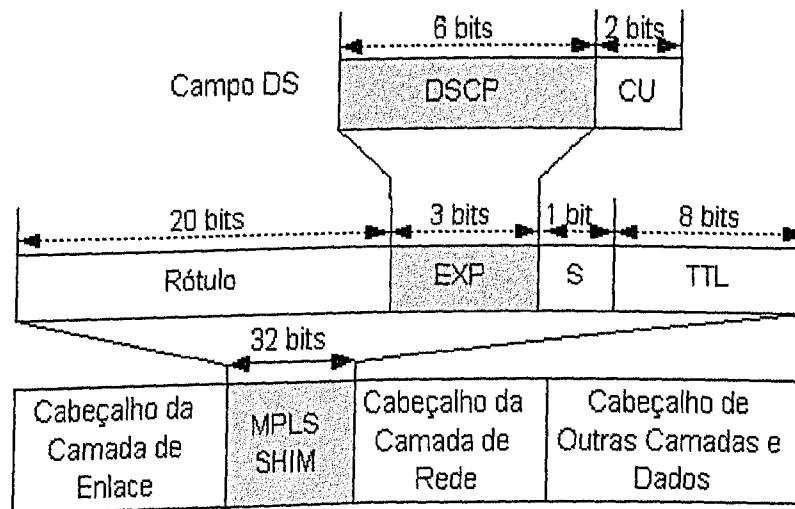


Figura 4.4: Campos EXP e DSCP.

4.6.2 E-LSP (*EXP-Inferred-PSC LSP*)

O E-LSP determina o PHB de um pacote unicamente do campo EXP, e assim pode suportar até 8 PHBs por E-LSP. O campo de EXP transmite o enfileiramento, escalonamento e a precedência de descarte do pacote para o LSR. O PHB pode ser usado para sinalizar explicitamente os PHBs suportados durante o estabelecimento do LSP, mas pode ser pré-configurado [40].

4.6.3 L-LSP (*Label-Only-Inferred-PSC LSP*)

Os pacotes de um micro fluxo devem manter a mesma ordem do LSR de ingresso ao LSR de egresso, assim eles pertencem à mesma PSC [41], que é um grupo de PHBs tais que a ordem dos pacotes no grupo deve ser preservada, e são colocados em uma fila comum. O conjunto de BAs cuja ordem deve ser mantida durante a transmissão constitui um OA (*Ordered Aggregate*)[41].

O L-LSP determina o PHB de um pacote para ambos os campos label e EXP. O campo label determina o PSC (filas e escalonamento) enquanto o campo EXP determina o PHB (precedência de descarte). Um número arbitrariamente grande de PHBs pode ser suportado. O objeto de DiffServ definido na extensão RSVP ou o DiffServ TLV definido na extensão LDP pode ser usado para suportar o grupo programado de PHBs, que é usado para sinalizar o PSC durante o estabelecimento de um L-LSP [40].

4.6.4 Modelo de operação

Para um E-LSP, cada LSR ao longo do caminho usa o rótulo para determinar onde o pacote deve ser encaminhado e o campo EXP determina o PHB aplicado ao pacote durante o processo de encaminhamento. A Figura 4.5 mostra como um E-LSP suporta três diferentes PHBs transportados em um agregado de fluxos do LER de entrada para o LER de saída.

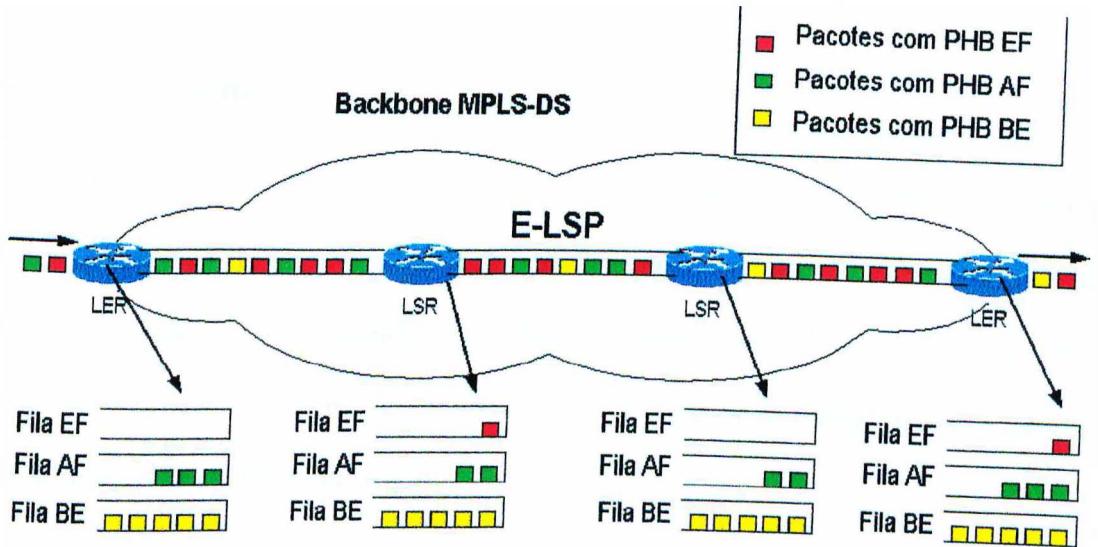


Figura 4.5: E-LSP suportando três PHBs.

4.6.5 Determinação do PHB de entrada

Para E-LSP, o mapeamento EXP-to-PHB pode ser pré-configurado ou explicitamente sinalizado durante o estabelecimento do E-LSP. O LSR determina o PHB a ser aplicado ao pacote que entra na rede, examinando o campo EXP no mapeamento EXP-to-PHB [40].

Para L-LSP, o mapeamento EXP-to-PHB é uma função do PSC transportado no L-LSP, e é ajustado durante o estabelecimento do L-LSP. Consequentemente, o PSC (*PHB Scheduling Class*) (isto é, filas e escalonamento) é conhecido ao LSP baseado no campo label. O LSR então determina a precedência de descarte do pacote, assim o PHB é aplicado ao pacote que chega examinando o campo EXP no mapeamento EXP-to-PHB.

[40].

4.6.6 Determinação do PHB de saída

Um LSR com capacidades DiffServ pode executar marcação, policiamento e formar os fluxos de tráfego que entram na rede, mudando potencialmente os PHBs associados aos pacotes que não estão adequados com o fluxo de tráfego [40]. Assim o PHB que entra e sai da rede pode ser diferente. Se nenhum condicionador de tráfego for executado, os PHBs que entram e saem da rede serão idênticos.

4.6.7 Encaminhamento do rótulo no domínio

Cada LSR com capacidades DiffServ deve conhecer o contexto DS para um rótulo, o qual é armazenado na NHLFEs para cada rótulo que sai da rede [40]. Um contexto DS consiste nos seguintes itens:

- Tipo de LSP
- PHBs suportado
- Mapeamento EXP-to-PHB para um rótulo que entra na rede
- Mapeamento PHB-to EXP para um rótulo que sai da rede

Esta informação é armazenada na ILM e FTN durante a atribuição do rótulo e é usada para encaminhar pacotes ao próximo salto. Um rótulo pode ser mapeado a várias NHLFEs na ILM ou FTN para permitir vários próximos saltos em caso de balanceamento de carga e re-roteamento rápido.

PHB	DSCP	Campo EXP	Classe de Serviço MPLS
EF	101110	110	Ouro
AF11	001010	101	Prata
AF31	011010	011	Bronze
BE	000000	000	Melhor Esforço

Tabela 4.1: Mapeamento DSCP-EXP.

A Tabela 4.1 mostra como o campo EXP do rótulo MPLS pode ser utilizado por pacotes transitando pela rede MPLS para obter o serviço DiffServ equivalente.

4.6.8 Codificação da informação DiffServ na camada de encapsulamento

Para E-LSP, o mapeamento PHB-to-EXP pode ser pré-configurado ou sinalizado explicitamente durante o estabelecimento do E-LSP. O LSR determina o valor do campo EXP do pacote que sai da rede examinando o PHB no mapeamento PHB-to-EXP [40].

Para L-LSP, o mapeamento PHB-to-EXP é uma função do PSC transmitido no L-LSP, e é ajustada durante o estabelecimento do L-LSP. O LSR determina o valor do campo EXP do pacote que sai examinando o PHB do mapeamento PHB-to-EXP [40].

4.7 Conclusões

A arquitetura DiffServ provê mecanismos que conduzem ao oferecimento de um nível diferenciado de serviços na Internet. Na arquitetura DiffServ são definidos níveis de

prioridade, visando oferecer garantias por fluxos agregados.

A implementação proposta pela arquitetura DiffServ é denominada Domínio DS. Este domínio é constituído de roteadores de borda e roteadores de núcleo. O roteadores de borda são responsáveis pelas atividades referentes à classificação dos pacotes em agregações de fluxos e ao condicionamento de tráfego, o qual verifica a adequação dos pacotes ao perfil de tráfego contratado. Este perfil é estabelecido entre o cliente e o provedor de serviços (ISP) e descrito em um Acordo de Nível de Serviço (SLA). Os roteadores de núcleo determinam o encaminhamento dos pacotes de acordo com a classificação estabelecida.

A engenharia de tráfego MPLS, desonta como uma arquitetura complementar ao provimento de QoS, uma vez que apresenta recursos de orientação à conexão na camada de rede, além da característica de roteamento explícito, onde são inseridos rótulos aos pacotes na admissão dos domínios MPLS, exigindo somente a leitura destes campos de tamanho fixo e reduzido pelos roteadores interiores aos domínios, agilizando o processo de encaminhamento dos pacotes. A possibilidade de ligar os rótulos a classes de equivalência de encaminhamento (FECs), se apresenta como um grande precursor da arquitetura DiffServ, que atuando em conjunto com o MPLS, permite no roteador de borda de um domínio DiffServ - MPLS, o mapeamento das classes de serviço do DiffServ em FECs do MPLS e os fluxos de dados ao longo do domínio são tratados a partir da leitura do rótulo pelos roteadores interiores ao domínio.

O principal objetivo do MPLS é agilizar o roteamento e o balanceamento de carga através da engenharia de tráfego. Como ele não oferece qualidade de serviço diretamente,

necessita de outro mecanismo para prover QoS. Assim, o DiffServ é o mecanismo de provisão de QoS mais indicado para ser utilizado juntamente com o MPLS por ser bastante escalável e possuir características em comum com o MPLS.

Capítulo 5

Avaliação de Desempenho de uma Rede IP com DiffServ e Engenharia de Tráfego Baseada em MPLS

5.1 Introdução

Os problemas de congestionamento na Internet e a necessidade de provisão de Qualidade de Serviço para aplicações multimídia e em tempo real é uma realidade. Mecanismos para o oferecimento de QoS, tais como DiffServ, que é o mais indicado para a Internet, e ainda o mecanismo de roteamento rápido MPLS são propostas bastante interessantes. Isoladamente, cada um destes mecanismos oferece benefícios particulares à Internet. O DiffServ classifica os fluxos de acordo com a QoS exigida por cada um deles e oferece, no mínimo, a QoS que foi contratada pelo cliente para cada fluxo. O MPLS classifica

os fluxos em FECs, conseguindo transportá-los por caminhos que não sofrem grandes congestionamentos, resultando em um roteamento mais rápido que o roteamento IP convencional.

O principal objetivo do MPLS é agilizar o roteamento e o balanceamento de carga através da engenharia de tráfego. Como ele não oferece qualidade de serviço diretamente, necessita de outro mecanismo para prover QoS. Assim, o DiffServ é o mecanismo de provisão de QoS mais indicado para ser utilizado juntamente com o MPLS por ser bastante escalável e possuir características em comum com o MPLS.

Neste trabalho, a proposta de estudo consiste em considerar um ambiente de rede, onde os tráfegos multímidia oriundos de um ISP local são classificados na borda da rede *backbone* em sub-classes de tráfego definidas pelo DiffServ. Em seguida, estes tráfegos são mapeados em diferentes LSPs com diferentes características de vazão, atraso e jitter.

Desta forma, através de modelagem e simulação, é feito um estudo comparativo de desempenho em termos de Qualidade de Serviço fim-a-fim, em diferentes condições, no ambiente de rede apresentado. Consideram-se como condições o uso de uma rede *backbone* IP sem nenhum mecanismo adicional de oferecimento de QoS, ou seja, a situação atual da Internet; em seguida, as implementações isoladas do DiffServ e do MPLS, e posteriormente, a implementação integrada de ambas as tecnologias nessa rede *backbone*, a fim de avaliar a utilização de diferentes tecnologias integradas. Para esta última condição considera-se dois métodos de re-roteamento MPLS: o método proposto por Haskin, e o *L3 Control Driven* (*L3 Dirigido ao Controle*), a fim de avaliar possíveis falhas que possam ocorrer em enlaces da rede *backbone*.

Aplicação	Vazão	Atraso	Jitter
Áudio	192 a 1411 Kbps	- 24 ms	- 100 ms
Voz	4 a 64 Kbps	100 a 250 ms	- 400 ms
Vídeo MPEG	1 a 6 Mbps	100 a 500 ms	- 100 ms
Vídeo-conferência	112 Kbps	100 a 500 ms	- 400 ms
WWW, FTP, Telnet	—	—	—

Tabela 5.1: Requisitos típicos de QoS.

5.2 Classificação das Aplicações

A tabela 5.1 apresenta os requisitos típicos de aplicações em rede. Os requisitos de QoS de uma aplicação especificam um conjunto de características quantitativas ou qualitativas de processamento e de comunicação suportadas por um serviço e que permite a provisão da funcionalidade desejada por usuários do ambiente. Estas características são chamadas de parâmetros de especificação da QoS. Normalmente, serviços de transporte de dados, possuem associados os seguintes parâmetros quantitativos de QoS: disponibilidade do serviço, retardo e variação do retardo na entrega de pacotes, vazão e taxa de perda de pacotes, dentre outros [10]. Uma característica qualitativa de um serviço define uma relação comparativa com outro. Um exemplo de parâmetro qualitativo poderia ser a entrega de pacotes com o menor retardo possível. Neste caso, o parâmetro de QoS especifica uma característica de retardo de um determinado serviço em relação aos demais.

A QoS solicitada ao ISP depende dos requisitos específicos das aplicações. De acordo com este critério, as aplicações distribuídas podem ser classificadas em aplicações em

tempo real e aplicações elásticas [42].

5.2.1 Aplicações em tempo real

As aplicações em tempo real caracterizam-se por uma forte dependência em relação aos instantes de entrega dos pacotes de informação pela rede. Em geral, envolvem o transporte de mídias contínuas, como a reprodução de um áudio ou vídeo remotamente. Após ser transmitido, um pacote acumula retardos ao longo dos enlaces e dos elementos de comunicação intermediários, normalmente roteadores, antes de ser reproduzido no destino. O retardo fim-a-fim total sofrido por um pacote corresponde ao retardo de propagação nos enlaces, o retardo de transmissão e tempo de fila de cada pacote nos roteadores antes do seu efetivo encaminhamento. Esta parcela de retardo, chamado de retardo de enfileiramento, vai depender das condições de congestionamento em cada roteador. Em situações extremas, pacotes podem ser perdidos ou entregues no destino após o instante correto de reprodução. Ambos os casos introduzem distorções na reprodução, que são, até um certo grau, aceitáveis devido às limitações da percepção humana e à natureza tolerante da maioria das aplicações. Nestes casos, diz-se que a aplicação em tempo real é tolerante à perda de pacotes. Uma aplicação tolerante pode ainda alterar o ponto de reprodução dos pacotes a partir dos retardos reais experimentados pelos mesmos na rede, sendo chamadas de adaptativas. Como exemplo de técnica de adaptação, cita-se, na reprodução de um vídeo, o descarte de quadros, caso o retardo dos pacotes recebidos seja menor do que o esperado e a capacidade de armazenamento dos mesmos tenha chegado ao seu limite, ou a reprodução seguida de um mesmo quadro, caso o referido retardo seja maior do

que o esperado. Obviamente, a capacidade de adaptação de uma aplicação é limitada pelo nível de distorção aceito pelo usuário. Além do mais, nem todas as aplicações em tempo real são tolerantes. Um vídeo contendo informações médicas, por exemplo, não deve permitir qualquer distorção na reprodução sob o risco de ocultar informações consideradas vitais. As aplicações em tempo real que não permitem adaptações são chamadas de não tolerantes.

A distinção entre aplicações em tempo real tolerantes, adaptativas ou não, e não tolerantes é importante na medida em que identifica a necessidade de diferentes níveis de QoS a serem fornecidos pelo provedor de serviços. As aplicações não tolerantes exigirão um serviço que forneça garantias em relação aos requisitos de QoS solicitados. Por outro lado, as aplicações tolerantes serão muito mais flexíveis no que diz respeito à QoS solicitada e a efetivamente fornecida pela rede, principalmente as tolerantes adaptativas. Esses diferentes níveis de QoS influenciarão diretamente no grau de compartilhamento dos recursos e, consequentemente, na quantidade de usuários que poderão ser atendidos simultaneamente pela rede. Os parâmetros de QoS mais importantes no suporte às aplicações em tempo real são o retardo máximo fim-a-fim e a sua variação na entrega dos pacotes. Para as aplicações não tolerantes, acrescenta-se a taxa de perda de pacotes e a disponibilidade do serviço [42].

5.2.2 Aplicações Elásticas

As aplicações elásticas, por sua vez, não dependem tanto dos instantes de entrega dos pacotes, admitindo maiores variações no desempenho da rede. Por outro lado, perdas de

pacotes não são, em geral, admitidas. Para estas aplicações, os parâmetros de QoS mais importantes são a vazão média e a taxa de perda de pacotes. Exemplos de aplicações elásticas: correio eletrônico, transferência de arquivos (FTP), *login* remoto (Telnet), dentre outras.

De acordo com os diferentes requisitos das aplicações, tem-se uma idéia da dificuldade em se prover, em uma única infraestrutura de rede, serviços capazes de atender às mais diversas aplicações. Esta dificuldade torna-se maior quando se exige que o uso dos recursos desta rede seja eficiente, permitindo o maior grau de compartilhamento possível [42].

5.3 Suporte à QoS Fim-a-Fim em Redes MPLS - Diff-

Serv

Atualmente as redes IP trabalham com o serviço de melhor esforço. Neste esquema, todos os pacotes dos usuários competem igualmente pelos recursos da rede e nenhuma garantia é dada sobre quando ou se os pacotes chegarão ao destino. Entretanto, com o crescimento acentuado da Internet nos últimos anos e o surgimento de novas classes de aplicações distribuídas, tais como multimídia e em tempo real baseadas em voz e vídeo, impõem rígidos limites para os parâmetros de qualidade de serviço fim a fim, como atraso, variação do atraso (*jitter*) e vazão.

Diante da complexidade da Internet, que é formada por vários Sistemas Autônomos (*Autonomous Systems - ASs*), uma malha complexa de enlaces e roteadores é formada. Assim, a arquitetura DiffServ sendo implementada em cada AS, provê atualmente a me-

lhor forma de obtenção de QoS fim a fim na Internet [16]. O DiffServ utiliza algoritmos de roteamento IP convencionais, ou seja, o menor caminho para rotear os pacotes de uma origem até um destino através destes ASs. Para a obtenção dos limites para os parâmetros de QoS exigidos por aplicações típicas, é necessário balancear a carga nos roteadores, de forma que otimize as rotas entre um *host* de origem e um *host* de destino. A proposta para otimizar essas rotas é a engenharia de tráfego com o mecanismo MPLS.

Desta forma, em um *backbone* MPLS/DiffServ, os tráfegos são classificados na borda da rede em sub-classes de tráfegos definidas pelo DiffServ, que são mapeados em diferentes LSPs com diferentes características de vazão, atraso e *jitter*.

5.3.1 Trabalhos Relacionados

Várias pesquisas propõem esquemas de obtenção de QoS em redes MPLS-DiffServ. Em [51], mostra-se como o MPLS combinado com roteamento baseado em restrições e serviços diferenciados forma um modelo de Internet simples e eficiente capaz de prover diferentes níveis de QoS às aplicações. O artigo [49] combinou a tecnologia DiffServ com engenharia de tráfego sobre MPLS para oferecer um mecanismo adaptativo que seja capaz de rotear o tráfego IP de alta prioridade, em múltiplos caminhos paralelos com restrições de atraso. Em [52], propõe-se uma arquitetura heterogênea para obtenção de qualidade de serviço baseado em MPLS. Já em [48] estuda-se os conceitos e apresenta-se resultados de simulação para redes MPLS/DiffServ. Em [50], propõe-se um esquema de engenharia de tráfego por classes em E-LSPs. O artigo [35] discute-se a praticabilidade de suportar políticas de QoS fim-a-fim em redes DiffServ-aware-MPLS. Em [33], propõe-se

uma melhoria na integração de mecanismos para garantia de QoS fim-a-fim, introduzindo mecanismo de interoperabilidade entre um sistema de gerenciamento baseado em políticas pertencentes ao ISP e agentes móveis pertencentes aos usuários para facilitar a negociação entre diferentes domínios. O artigo [38] discute os aspectos da engenharia de tráfego com MPLS na Internet.

5.4 Ambiente de Rede e Parâmetros de Simulação

Este estudo é baseado em modelagem e simulação e utilizou o Network Simulator - NS em sua versão ns-2 [44], [45]. A adição dos módulos MNS-2.0 [46] e DiffServ patch [47] à versão básica do ns-2, possibilitou a utilização de diferentes mecanismos para o suporte à simulação de serviços diferenciados e MPLS [48], [49], [50].

O ambiente de rede adotado neste estudo é apresentado na Figura 5.1. Um modelo simplificado de um determinado número de ISPs e *backbone* da Internet, representam o ambiente de rede adotado neste estudo.

Considerou-se a existência de quatro fontes de tráfegos em uma rede de um provedor local (ISP 1): voz, vídeo e duas fontes FTP. O ISP 1 está conectado diretamente a uma rede *backbone* (ISP Regional) através dos roteadores de borda R1 e LER 1, que implementa o mecanismo DiffServ para diferenciação dos serviços e MPLS para prover engenharia de tráfego. Na outra extremidade, considerou-se os *hosts* conectados diretamente a uma outra rede de um provedor local (ISP 2) através dos roteadores de borda LER 2 e R2. No ISP 2 considerou-se que dois *hosts* recebem os tráfegos oriundos do ISP 1. Os enlaces

entre o roteador dos ISPs e a rede *backbone* operam a 10 Mbps com latência de 10 ms [58].

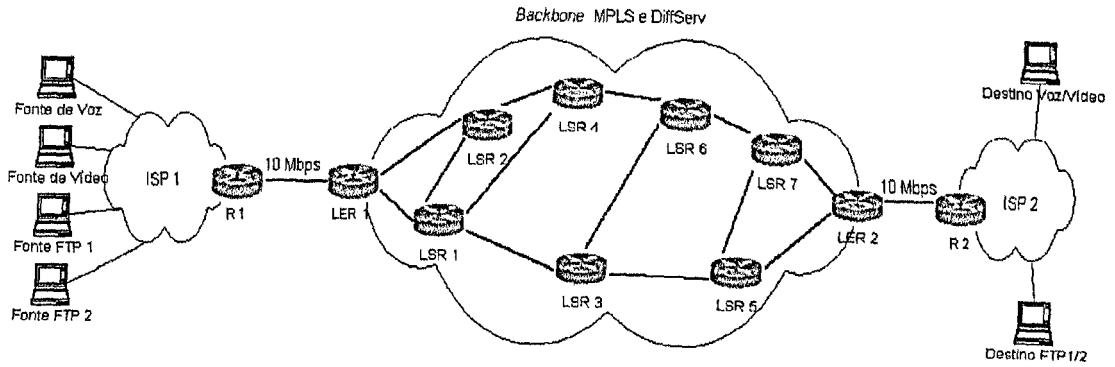


Figura 5.1: Ambiente de Rede.

A largura de banda dos enlaces da rede *backbone* é de 1 Mbps com uma latência de 10 ms. O tráfego de voz é modelado por uma fonte CBR (*Constant Bit Rate*) a uma taxa de 64 kbps, sendo transportado pelo UDP (*User Datagram Protocol*) e classificado como ouro. O tráfego de vídeo, representado por sistemas de videoconferência, é modelado por uma fonte CBR a uma taxa de 115 kbps, sendo transportado pelo UDP e classificado como prata. O tráfego de dados é transportado pelo TCP (*Transmission Control Protocol*), que é gerado por aplicações FTP ativas durante toda a simulação. Uma fonte denominada FTP2, que pode, por exemplo, ser usada para transações bancárias é classificada como bronze. E uma fonte FTP1, que pode ser usada para transferência comum de arquivos foi classificada como melhor esforço [58]. A Tabela 5.2 mostra a classificação das aplicações.

Aplicação	PHB	Classe de Serviço MPLS
Voz	EF	Ouro
Vídeo	AF11	Prata
FTP1	AF31	Bronze
FTP2	BE	Melhor Esforço

Tabela 5.2: Classificação das aplicações.

5.5 Apresentação e Análise de Resultados

Nesta sessão apresenta-se uma análise dos resultados obtidos através de experimentos baseados em simulação. Avalia-se quatro ambientes, o primeiro é avaliado sem nenhum mecanismo de QoS, o segundo com somente o DiffServ na rede, o terceiro com somente o MPLS na rede e por final a combinação MPLS e DiffServ. Para todas as simulações foram avaliados a vazão, o atraso fim-a-fim e a variação do atraso (*jitter*). A taxa de perdas não foi avaliada porque o objetivo principal do trabalho visa aplicações multimídias, e neste caso, a taxa de perdas torna-se irrelevante devido às características destas aplicações que aceitam pequenas perdas de pacotes.

Para avaliar o mecanismo de re-roteamento MPLS, introduz-se uma falha no enlace entre o LSR 3 e o LSR 5. Apresenta-se uma comparação entre o método *L3 Control Driven* e o método proposto por Haskin. Estes mecanismos foram simulados no ambiente de rede onde se tem a combinação MPLS e DiffServ no *backbone*. Avalia-se, ainda, um ambiente de rede com somente o mecanismo DiffServ na rede, onde tem-se o roteamento IP convencional.

O parâmetro vazão é utilizado para avaliar o impacto da falha de um enlace, utilizando o mecanismo de re-roteamento MPLS.

5.5.1 Resultados sem a inclusão de mecanismos de QoS

Na fase inicial dos experimentos baseados em simulação, o ambiente de rede foi configurado sem nenhum mecanismo de QoS, ou seja, utilizando o serviço de melhor esforço suportado atualmente pela Internet e, consequentemente, todos os tráfegos receberam um mesmo tratamento.

As Figuras 5.2 e 5.3 mostram os resultados das simulações feitas para os tráfegos de voz, vídeo, FTP1 e FTP2 sem a implementação dos mecanismos MPLS ou DiffServ.

A Figura 5.2 mostra a vazão alcançada pelos tráfegos. De acordo com os resultados obtidos, observa-se que as fontes FTP 1 e FTP 2 consomem a largura de banda tanto quanto possível, sendo assim, não há justiça no compartilhamento da banda e uma má utilização do enlace [58].

A Figura 5.3 mostra os atrasos fim-a-fim, ou seja, o intervalo de tempo que o pacote leva para transitar em uma rede desde sua origem até seu destino, sofridos pelos tráfegos de voz, vídeo, FTP1 e FTP2. Conforme pode ser observado, os resultados obtidos sem o uso de mecanismos de QoS mostram valores que superam os limites superiores aceitáveis pelas aplicações multimídia [53] [58].

A Figura 5.4 mostra os níveis de *jitter* ou variação do atraso sofridos pelos pacotes dos tráfegos. Neste caso, tem-se também valores acima dos limites exigidos pelos tráfegos de voz e vídeo [53].

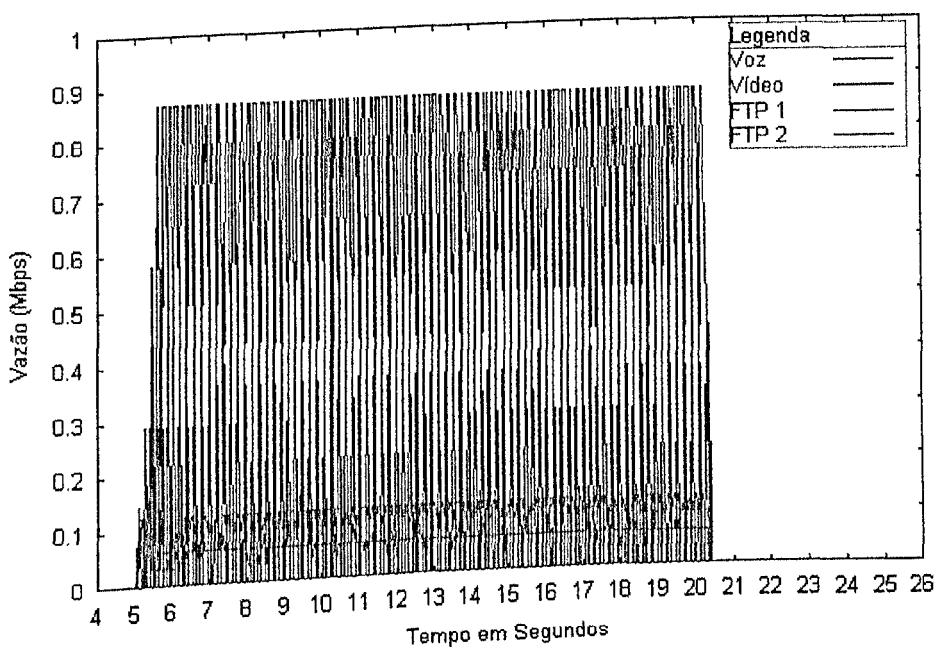


Figura 5.2: Vazão sem mecanismo de QoS.

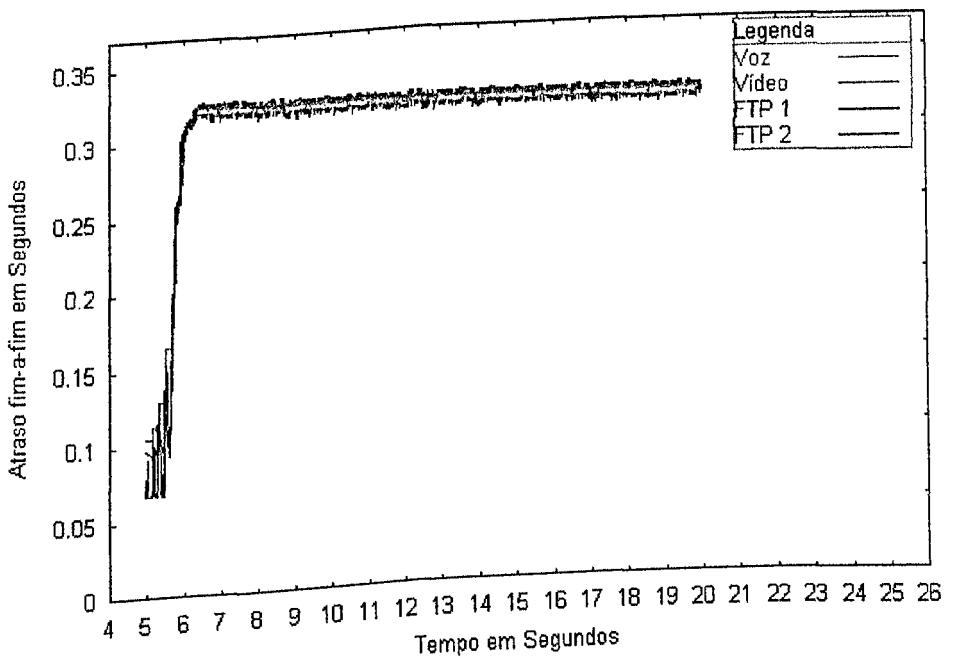


Figura 5.3: Atraso fim-a-fim sem mecanismo de QoS.

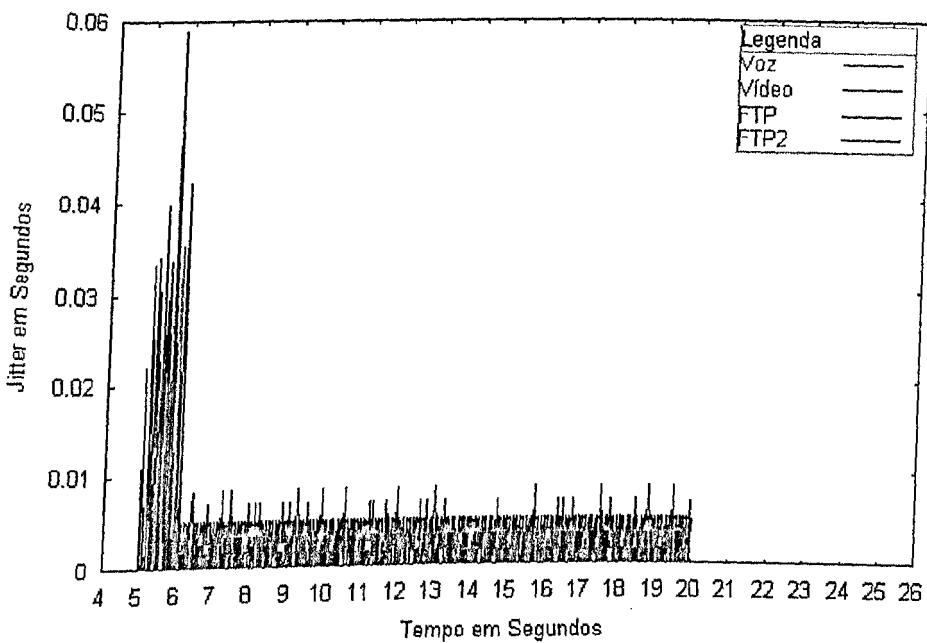


Figura 5.4: *Jitter* sem mecanismo de QoS.

5.5.2 Resultados obtidos quando se utiliza somente o mecanismo DiffServ

Na fase seguinte dos experimentos, a rede *backbone* foi configurada para implementar somente o mecanismo DiffServ para prover QoS fim-a-fim e os ISPs com o serviço de melhor esforço. Os pacotes referentes ao tráfego de voz foram marcados segundo a classe EF, e os pacotes de vídeo segundo a classe AF11. Os pacotes referentes ao tráfego de FTP2 foram marcados segundo a classe AF31 e os pacotes do tráfego FTP1 foram marcados segundo o modelo de serviço melhor esforço.

A Figura 5.5 apresenta a vazão alcançada pelos vários tráfegos. Como pode ser observado, somente com o mecanismo DiffServ há uma justiça no compartilhamento da banda,

mas ainda há uma má utilização do enlace [58].

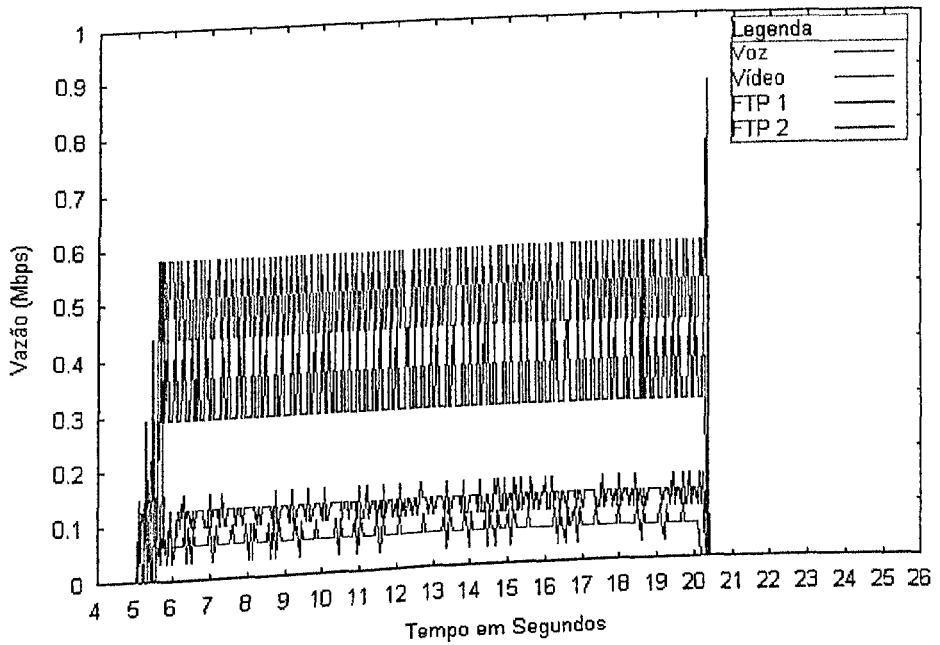


Figura 5.5: Vazão com mecanismo DiffServ.

A Figura 5.6 ilustra o atraso fim-a-fim a que os tráfegos ficaram sujeitos. Percebe-se uma melhora significativa para os tráfegos com requisitos de QoS comparando-se com o ambiente sem nenhum mecanismo de QoS. O atraso fim-a-fim para o tráfego de voz ficou próximo de 100 ms. Já o tráfego de vídeo ficou próximo de 250 ms, o que ainda não satisfaz os requisitos de QoS [58].

A Figura 5.7 mostra os níveis de jitter sofridos pelos pacotes dos tráfegos, e tem-se ainda valores acima dos exigidos pelos tráfegos de voz e vídeo.

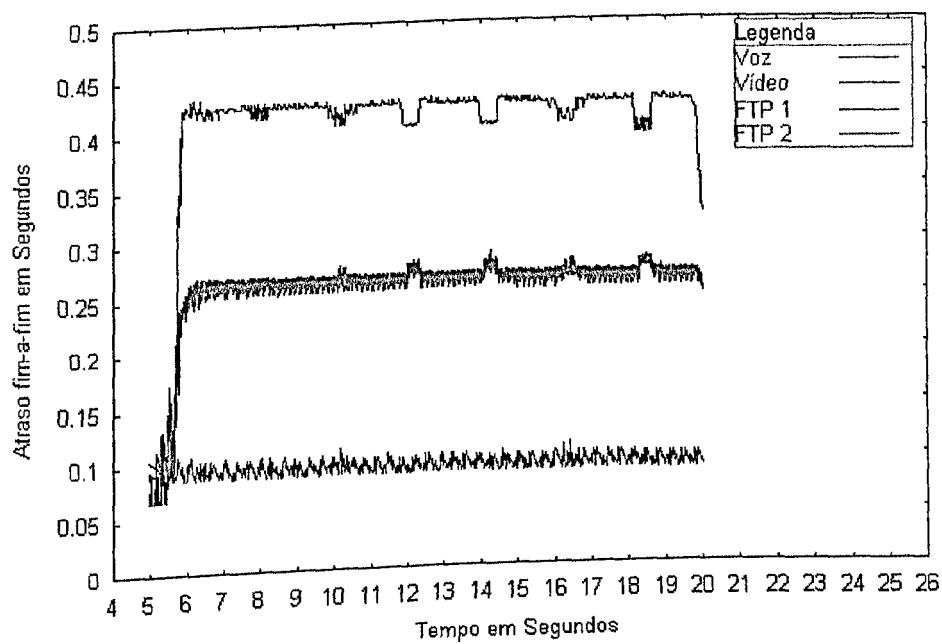


Figura 5.6: Atraso fim-a-fim com mecanismo DiffServ.

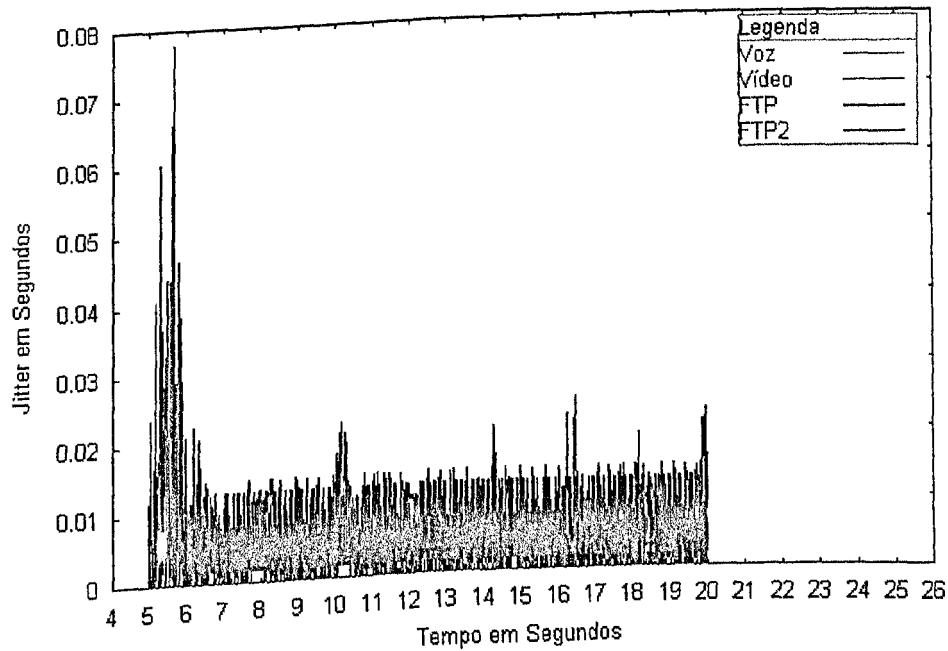


Figura 5.7: Jitter com mecanismo DiffServ.

5.5.3 Resultados obtidos quando se utiliza somente o mecanismo MPLS

Nesta fase dos experimentos, a rede *backbone* foi configurada para implementar somente o mecanismo MPLS e os ISPs seguiram o modelo melhor esforço.

O mecanismo MPLS provê um rápido roteamento e a engenharia de tráfego para balancear a carga da rede.

Foram simulados dois casos: no primeiro somente um ER-LSP foi estabelecido para todos os tráfegos e um segundo caso onde se tem dois ER-LSPs.

A seguir apresentam-se os dois casos citados anteriormente.

Resultados obtidos com um ER-LSP

Estabeleceu-se 1 ER-LSP para os tráfegos de voz, vídeo, FTP1 e FTP2. O caminho estabelecido para esse único ER-LSP é através dos roteadores R1-LER1-LSR1-LSR3-LSR5-LER2-R2 apresentados na Figura 5.1.

A Figura 5.8 apresenta a vazão alcançada pelos tráfegos. Como pode ser observado, somente com o mecanismo MPLS não há uma justiça no compartilhamento da banda, e uma má utilização do enlace, equiparando-se com o resultado sem mecanismo de QoS, isto se deve ao fato de que o MPLS implementado isoladamente não provê qualidade de serviço.

A Figura 5.9 ilustra o atraso fim-a-fim a que os tráfegos ficaram sujeitos. Percebe-se que os resultados equipararam com o ambiente sem mecanismo de QoS. Novamente isto se deve ao fato de que o MPLS implementado isoladamente não provê qualidade de serviço.

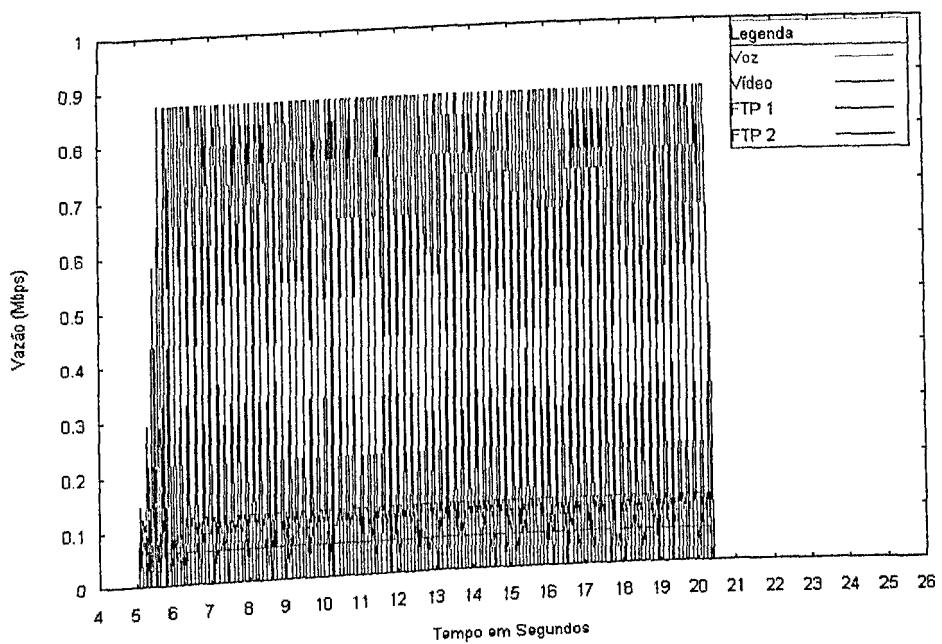


Figura 5.8: Vazão com mecanismo MPLS e um ER-LSP.

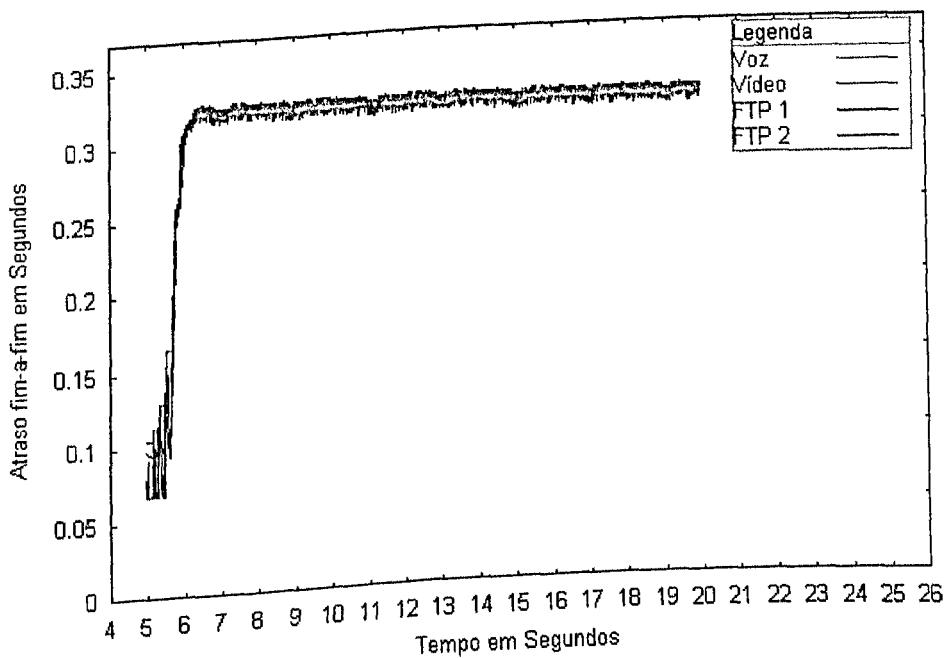


Figura 5.9: Atraso fim-a-fim com mecanismo MPLS e um ER-LSP.

A Figura 5.10 mostra os níveis de *jitter* sofridos pelos pacotes dos tráfegos. Tem-se ainda valores acima dos exigidos pelos tráfegos de voz e vídeo. Verifica-se que os resultados são aproximadamente iguais aos obtidos no ambiente de rede sem mecanismo de QoS implementado.

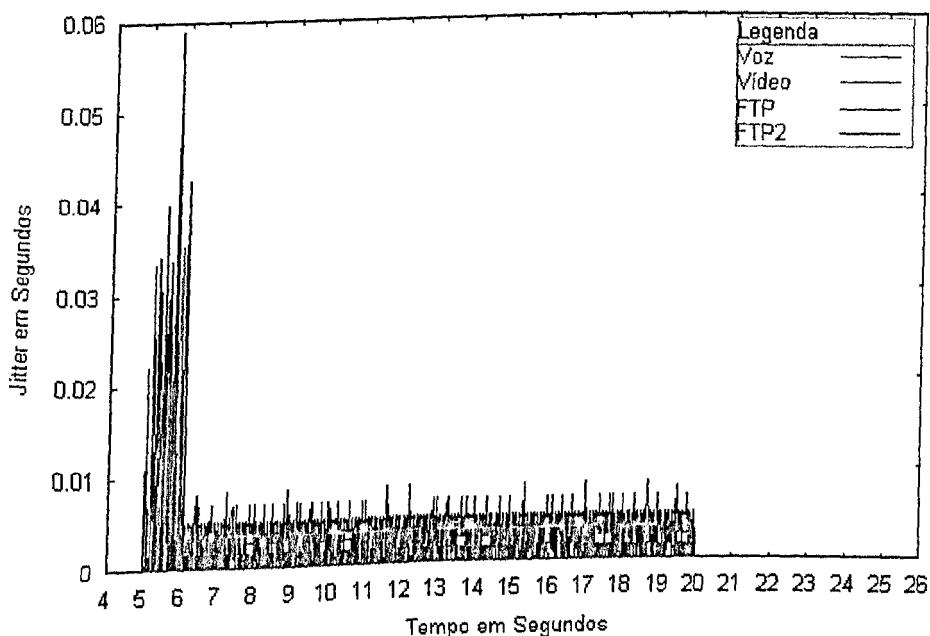


Figura 5.10: *Jitter* com mecanismo MPLS e um ER-LSP.

Resultados obtidos com dois ER-LSPs

Nesta fase dos experimentos estabeleceu-se 2 ER-LSPs. Um ER-LSP foi estabelecido para os tráfegos de voz e FTP1. O caminho estabelecido para este ER-LSP é através dos roteadores R1-LER1-LSR1-LSR3-LSR5-LER2-R2 apresentados na Figura 5.1.

O segundo ER-LSP foi estabelecido para os tráfegos de vídeo e FTP2. O caminho adotado para esse segundo ER-LSP é através do roteadores R1-LER1-LSR2-LSR4-LSR6-

LSR7-LER2-R2 apresentados na Figura 5.1.

A Figura 5.11 apresenta a vazão alcançada pelos tráfegos. Como pode ser observado, somente com o mecanismo MPLS com 2 ER-LSP já se percebe a influência da engenharia de tráfego provida pelo MPLS, assim há uma justiça no compartilhamento da banda e um balanceamento na carga da rede.

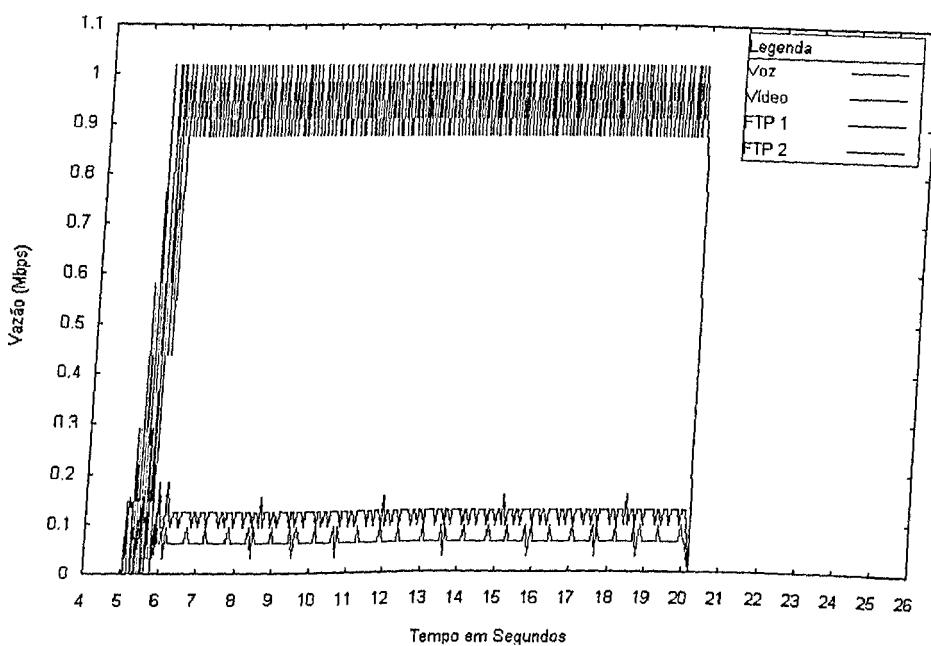


Figura 5.11: Vazão com mecanismo MPLS e dois ER-LSPs.

A Figura 5.12 ilustra o atraso fim-a-fim a que os tráfegos ficaram sujeitos. Percebe-se uma melhora significativa comparando-se com os ambientes sem mecanismo de QoS e com somente o mecanismo DiffServ. Mas com somente o mecanismo MPLS e 2 ER-LSPs já são satisfeitos os requisitos de QoS, devido à engenharia de tráfego MPLS que faz o balanceamento da carga da rede.

A Figura 5.13 mostra os níveis de *jitter* sofridos pelos pacotes dos tráfegos. De acordo

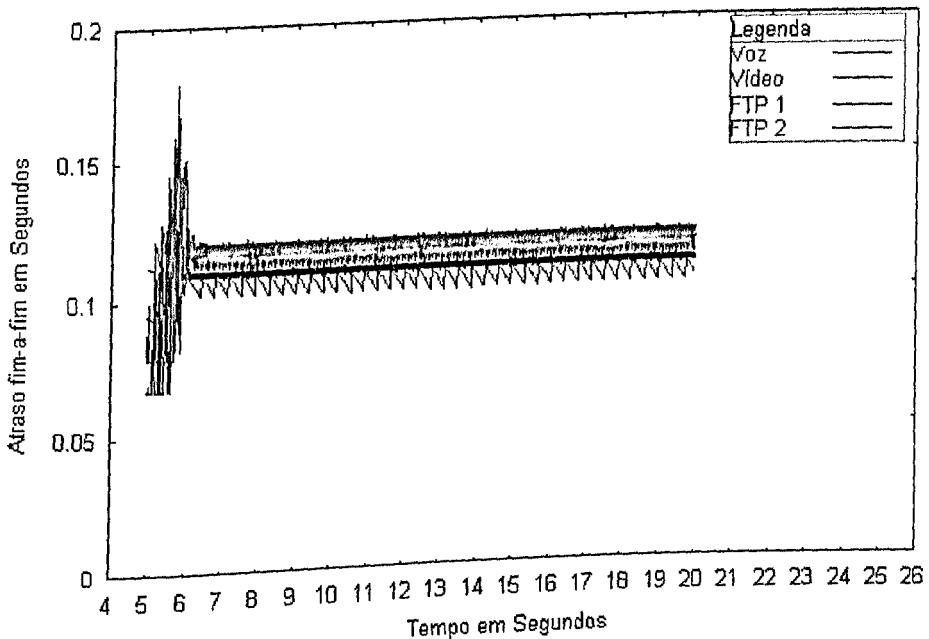


Figura 5.12: Atraso fim-a-fim com mecanismo MPLS e dois ER-LSPs.

com o resultado obtido os níveis de jitter já satisfaz os requisitos de QoS das aplicações, devido a engenharia de tráfego MPLS.

5.5.4 Resultados obtidos quando se utiliza os mecanismos Diff-Serv e MPLS

Na fase final dos experimentos, a rede backbone foi configurada para implementar o mecanismo DiffServ para a diferenciação dos serviços e a engenharia de tráfego provida pelo MPLS para obter a qualidade de serviço requerida pelas aplicações multimídia. Foram simulados ainda dois casos: no primeiro caso tem-se somente um ER-LSP estabelecido pelo mecanismo MPLS, e no segundo caso tem-se dois ER-LSPs. A seguir

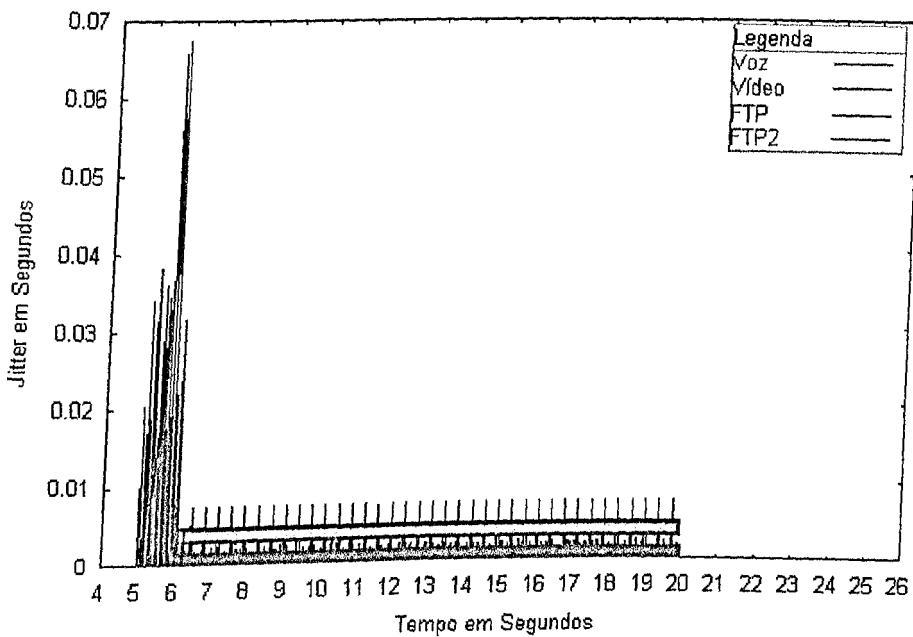


Figura 5.13: *Jitter* com mecanismo MPLS e dois ER-LSPs.

apresenta-se os resultados nos casos citados.

Resultados obtidos com um ER-LSP

Estabeleceu-se 1 ER-LSP para os tráfegos de voz, vídeo, FTP1 e FTP2, os quais foram classificados como ouro, prata, melhor esforço e bronze, respectivamente, pelo DiffServ, de acordo com a tabela 5.2. O caminho estabelecido para esse único ER-LSP é através dos roteadores R1-LER1-LSR1-LSR3-LSR5-LER2-R2 apresentados na Figura 5.1.

A Figura 5.14 apresenta a vazão alcançada pelos referidos tráfegos. Como pode ser observado, somente com um ER-LSP não utilizando a engenharia de tráfego propriamente dita, há justiça no compartilhamento da banda, porém há baixo nível de utilização do enlace, equiparando-se com o ambiente de rede onde tem-se somente o mecanismo DiffServ

implementado.

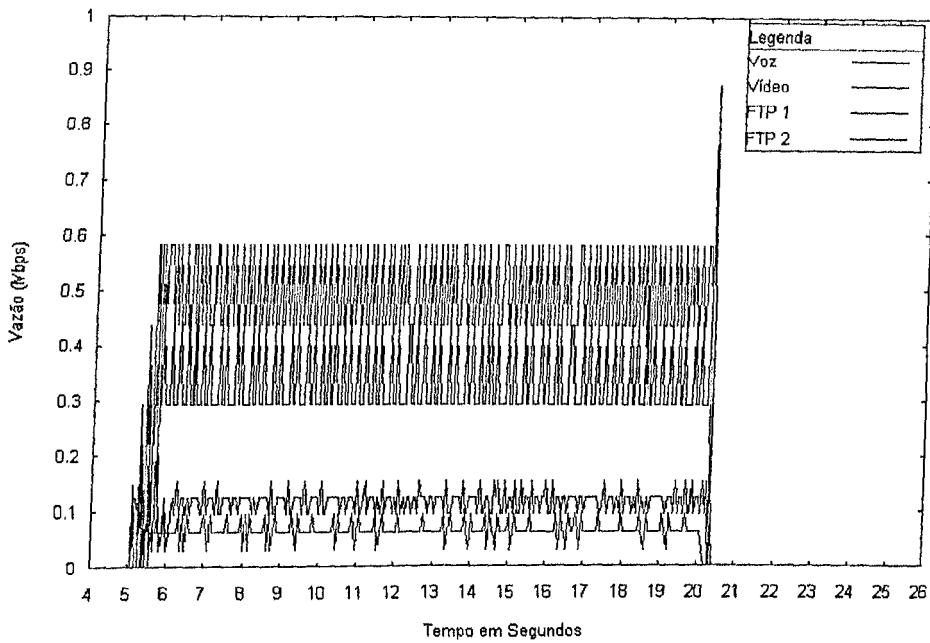


Figura 5.14: Vazão com os mecanismos DiffServ e MPLS e um ER-LSP.

A Figura 5.15 ilustra o atraso fim-a-fim a que os tráfegos ficaram sujeitos. Percebe-se uma melhora significativa comparando-se com o ambiente sem mecanismo de QoS. Contudo, com 1 ER-LSP as classes de tráfego definidas pelo DiffServ sofreram diferenças nos níveis de atraso fim-a-fim de acordo com seus requisitos. Sendo assim, já satisfaz os requisitos de QoS necessários para o tráfego de voz.

A Figura 5.16 mostra os níveis de *jitter* dos tráfegos, e de acordo com os resultados, tem-se valores que satisfazem os limites exigidos pelos tráfegos de voz e vídeo.

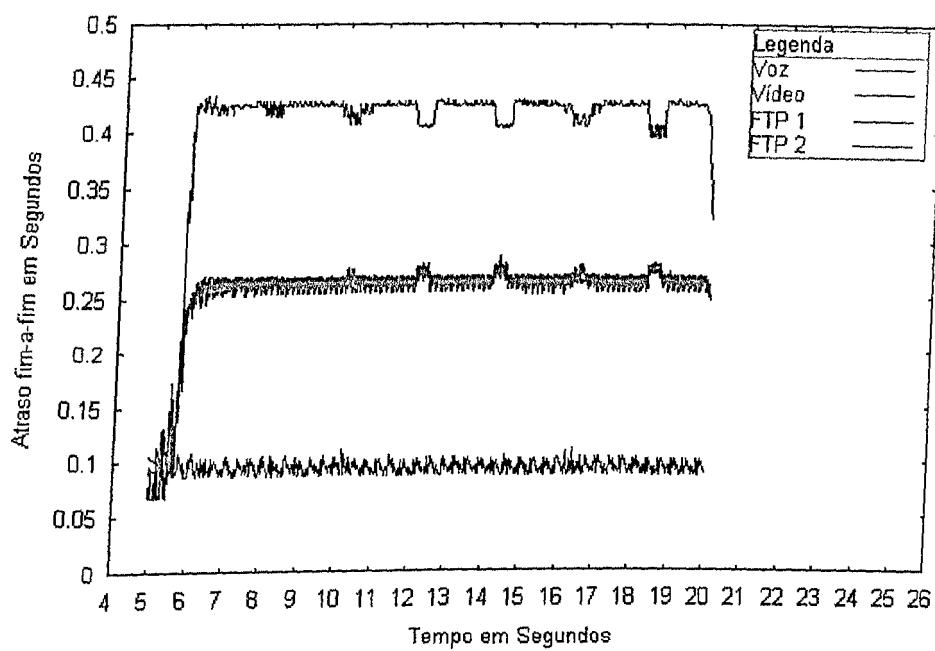


Figura 5.15: Atraso fim-a-fim com os mecanismos DiffServ e MPLS e um ER-LSP.

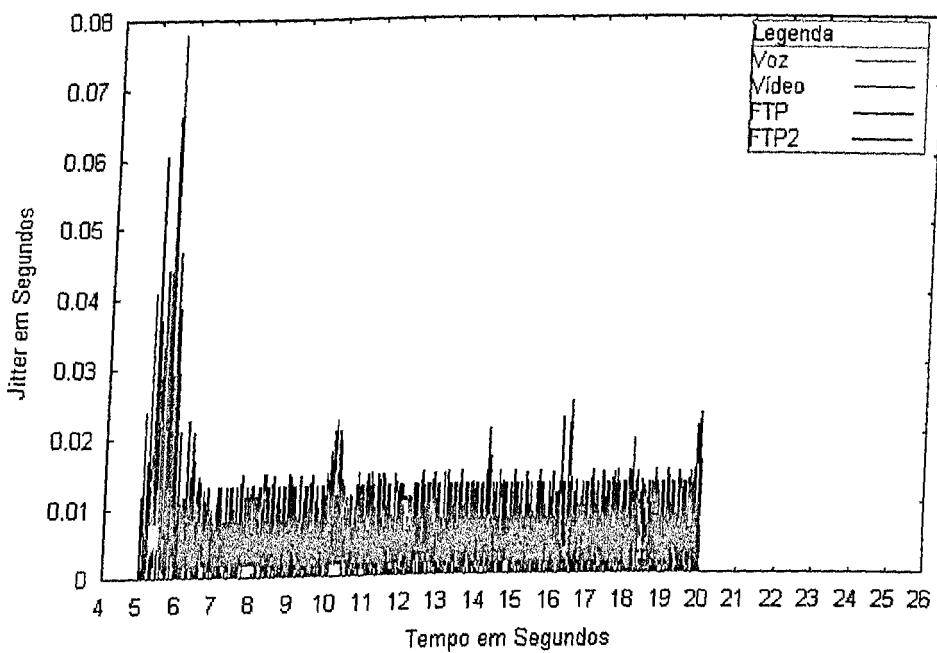


Figura 5.16: Jitter com os mecanismos DiffServ e MPLS e um ER-LSP.

Resultados obtidos com dois ER-LSPs

Nesta fase dos experimentos estabeleceu-se 2 ER-LSPs. Um ER-LSP foi estabelecido para os tráfegos de voz e FTP1, sendo classificados como ouro e melhor esforço pelo DiffServ, de acordo com a tabela 5.2. O caminho estabelecido para esse ER-LSP é através dos roteadores R1-LER1-LSR1-LSR3-LSR5-LER2-R2 apresentados na Figura 5.1.

O segundo ER-LSP foi estabelecido para os tráfegos de vídeo e FTP2, sendo classificados como prata e bronze pelo mecanismo DiffServ, de acordo com a tabela 5.2. O caminho adotado para esse segundo ER-LSP é através do roteadores R1-LER1-LSR2-LSR4-LSR6-LSR7-LER2-R2 apresentados na Figura 5.1.

A Figura 5.17 apresenta a vazão alcançada pelos tráfegos. Como pode ser observado, com o mecanismo MPLS provendo a engenharia de tráfego e a diferenciação dos serviços provida pelo mecanismo DiffServ, há uma justiça no compartilhamento da banda e um balanceamento na carga da rede. Garantindo assim os requisitos de QoS impostos pelas aplicações multimídia e em tempo real [53] [58].

A Figura 5.18 ilustra o atraso fim-a-fim a que os tráfegos ficaram sujeitos. Percebe-se que com a combinação das duas arquiteturas propostas permite obter os níveis de QoS exigidos pelas aplicações. E, neste caso, o atraso fim-a-fim do tráfego de voz ficou ainda menor comparando-se com a rede onde tem-se somente o mecanismo MPLS e com a engenharia de tráfego, garantindo assim limites exigidos para conversação [53] [58].

A Figura 5.19 mostra os níveis de jitter sofridos pelos pacotes dos tráfegos, e já tem valores que satisfazem os limites exigidos pelos tráfegos das aplicações multimídia e em tempo real [53] [58].

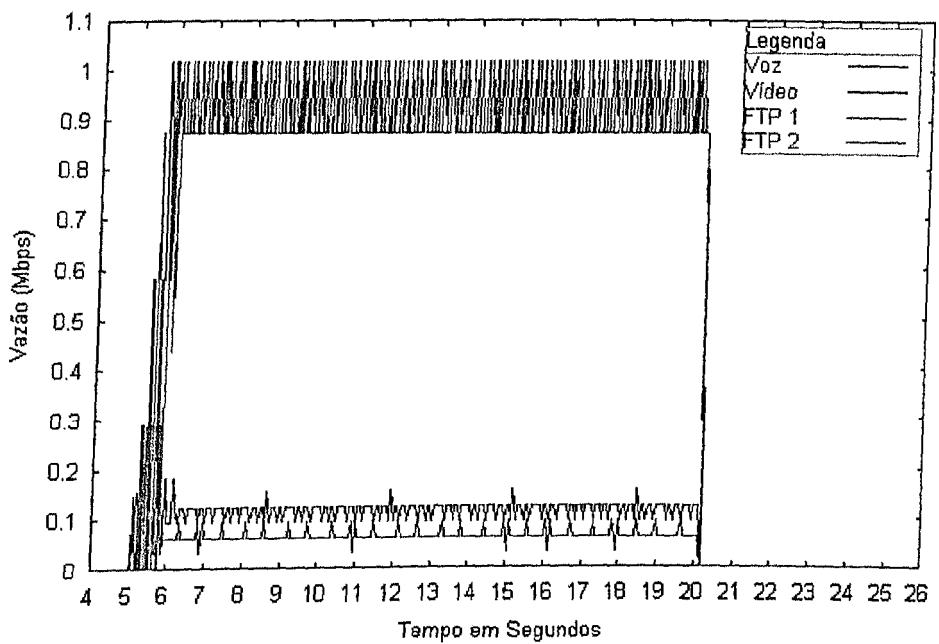


Figura 5.17: Vazão com os mecanismos DiffServ e MPLS e dois ER-LSPs.

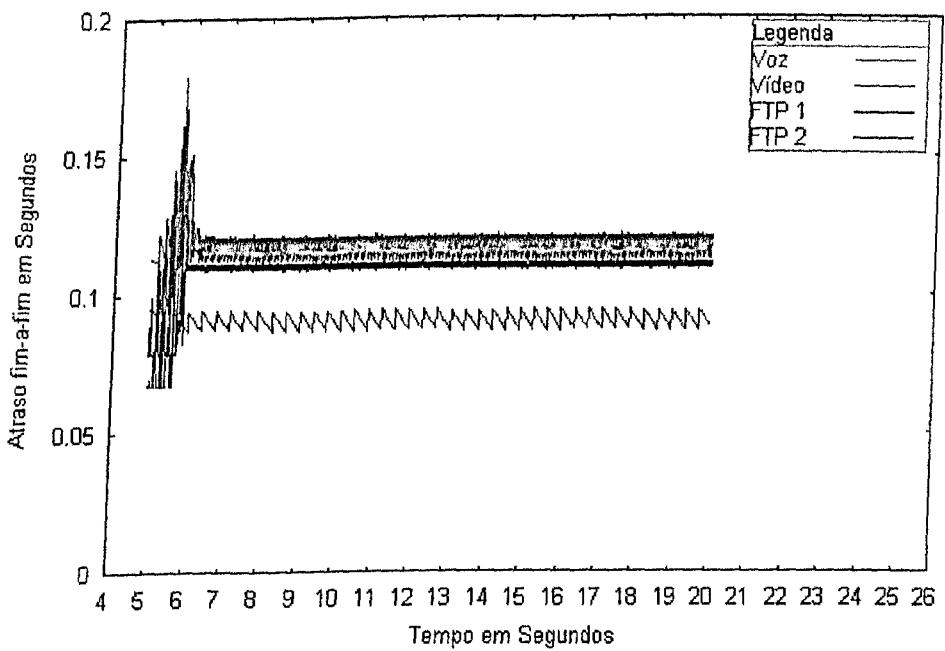


Figura 5.18: Atraso fim-a-fim com os mecanismos DiffServ e MPLS e dois ER-LSPs.

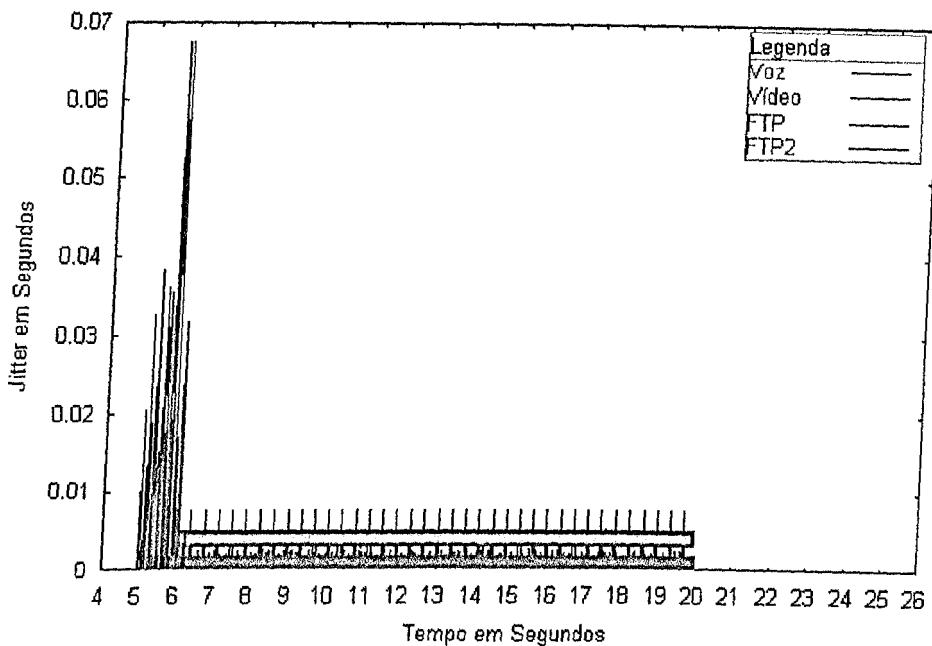


Figura 5.19: *Jitter* com os mecanismos DiffServ e MPLS e dois ER-LSPs.

5.5.5 Resultados obtidos quando se utiliza somente o mecanismo DiffServ e o Re-roteamento IP Convencional

Na fase seguinte dos experimentos, a rede *backbone* foi configurada para implementar somente o mecanismo DiffServ para prover QoS fim-a-fim e os ISPs com o serviço de melhor esforço. Os pacotes referentes ao tráfego de voz foram marcados segundo a classe EF, e os pacotes de vídeo segundo a classe AF11. Os pacotes referentes ao tráfego de FTP2 foram marcados segundo a classe AF31 e os pacotes do tráfego FTP1 foram marcados segundo o modelo de serviço melhor esforço.

Neste caso foi introduzido uma falha no enlace entre os roteadores LSR 3 e LSR 5 aos 10 segundos de simulação. No caso de falha é necessário um re-roteamento dos tráfegos e,

neste caso, o re-roteamento IP convencional. O parâmetro vazão foi utilizado para avaliar o re-roteamento dos tráfegos.

A Figura 5.20 apresenta a vazão alcançada pelos tráfegos. Como pode ser observado, com o mecanismo de re-roteamento IP convencional os tráfegos voltaram a ocupar a largura de banda após a correção da falha no enlace aos 15 segundos de simulação, demorando, portanto, cerca de 5 segundos para que os tráfegos alcançassem novamente seus destinos.

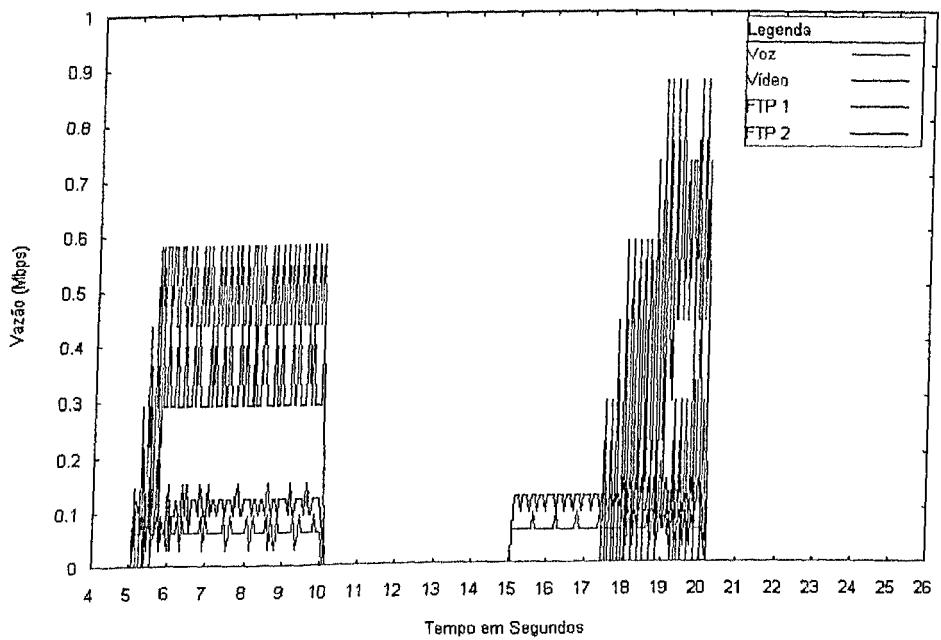


Figura 5.20: Vazão com o mecanismo DiffServ e falha no enlace.

5.5.6 Resultados obtidos quando se utiliza os mecanismos Diff-Serv e MPLS com o método Haskin de Re-roteamento

Nestes experimentos baseados em simulações, a rede backbone foi configurada para implementar o mecanismo DiffServ juntamente com o mecanismo MPLS e, os ISPs o serviço de melhor esforço. Os tráfegos de voz, vídeo, FTP1 e FTP2 foram classificados pelo mecanismo DiffServ como ouro, prata, melhor esforço e bronze, respectivamente, de acordo com a tabela 5.2. Neste caso foi introduzido uma falha no enlace entre os roteadores LSR 3 e LSR 5 aos 10 segundos de simulação.

Neste caso, adotou-se o método proposto por Haskin [29] a ser utilizado pelo MPLS para fazer o re-roteamento dos pacotes no momento em que for introduzida a falha na rede. Este método consiste em pré configurar os ER-LSPs antes mesmo das fontes iniciarem o envio de tráfegos.

A Figura 5.21 apresenta a vazão alcançada pelos tráfegos, percebe-se o benefício que a engenharia de tráfego baseada no MPLS com múltiplos LSPs, traz à rede DiffServ com o mecanismo de re-roteamento proposto por Haskin, onde, rapidamente os tráfegos voltam a atividade.

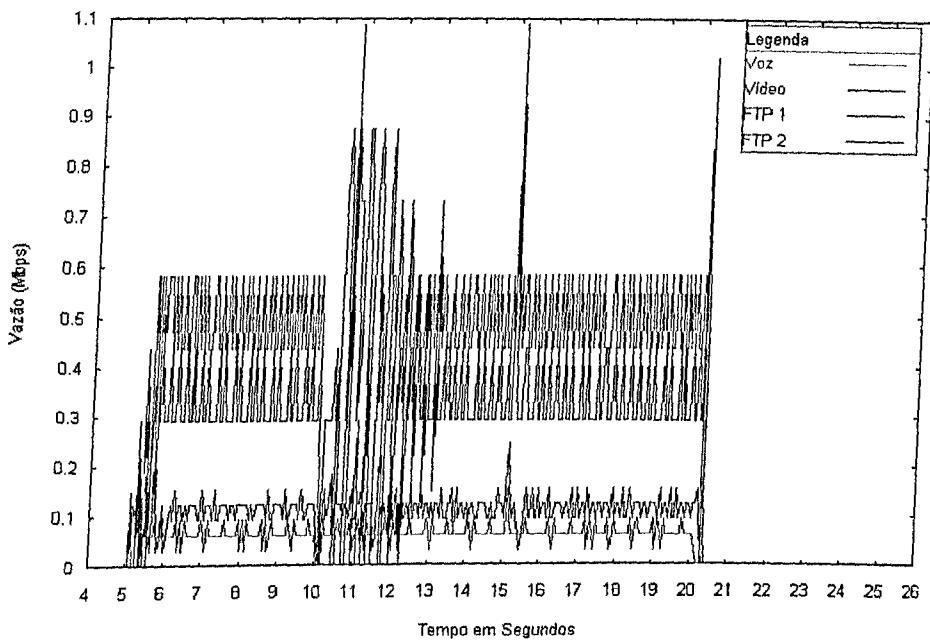


Figura 5.21: Vazão com DiffServ e MPLS e método Haskin de re-roteamento.

5.5.7 Resultados obtidos quando se utiliza os mecanismos Diff-Serv e MPLS com o método L3 *Control Driven* de re-roteamento

Nesta fase dos experimentos baseados em simulações, a rede backbone foi configurada para implementar mecanismo DiffServ juntamente com o mecanismo MPLS e os ISPs com o serviço de melhor esforço. Os tráfegos de voz, vídeo, FTP1 e FTP2 foram classificados pelo mecanismo DiffServ como ouro, prata, melhor esforço e bronze, respectivamente, de acordo com a tabela 5.2. Neste caso foi introduzido uma falha no enlace entre os roteadores LSR 3 e LSR 5 aos 10 segundos de simulação.

Neste caso foi utilizado o método L3 *Control Driven* (LDP Dirigido ao Controle)

utilizado pelo MPLS para fazer o re-roteamento dos pacotes no momento em que foi introduzida a falha na rede. Este método utiliza o Protocolo de Distribuição de Rótulos (LDP) fazer o roteamento na rede.

A Figura 5.22 apresenta a vazão alcançada pelos tráfegos. Como pode ser observado, com o mecanismo de re-roteamento L3 Control Driven os tráfegos voltam a ocupar a largura de banda aos 11 segundos de simulação, demorando assim apenas 1 segundo.

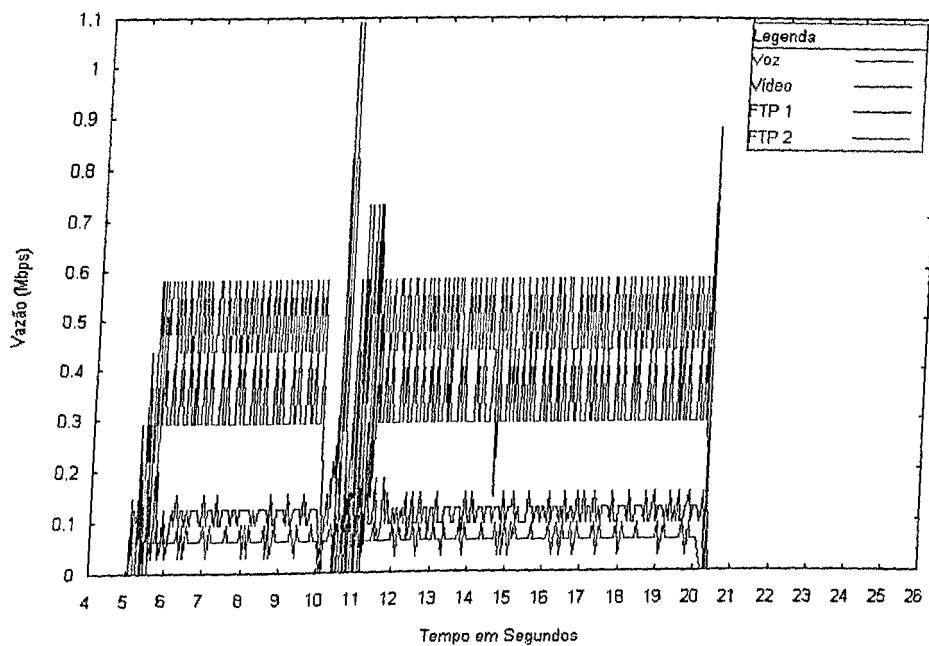


Figura 5.22: Vazão com DiffServ e MPLS e método L3 *Control Driven* de re-roteamento.

5.5.8 Conclusões

Atualmente as aplicações multimídia estão exigindo das redes o provimento de garantias de QoS. Neste trabalho, considerou-se um ambiente de rede, onde os tráfegos de voz e vídeo são classificados na borda da rede em sub-classes de tráfegos marcados pelo Diff-

Serv, e posteriormente mapeados em diferentes LSPs com diferentes requisitos de vazão, atraso fim-a-fim e *jitter*.

Desta forma, foram avaliados através de experimentos baseados em modelagem e simulação uma comparação entre vários ambientes de rede diferentes. Para um ambiente de rede sem nenhum mecanismo de QoS implementado, os limites requeridos pelas aplicações multimídia para seus parâmetros de QoS foram ultrapassados. Num ambiente de rede onde se implementou o mecanismo DiffServ, já observou-se o atendimento parcial aos requisitos de QoS. Para um ambiente com somente o mecanismo MPLS na rede não houve grandes contribuições devido a não provisão de QoS da arquitetura. No último experimento, onde se combinou os mecanismos MPLS e DiffServ no *backbone*, classificando o fluxo de tráfego em diferentes LSPs com seus respectivos requisitos de QoS, pode-se observar que o tráfego de voz e vídeo obteve os limites exigidos pelas aplicações sensíveis ao atraso.

Avaliou-se também, uma comparação entre métodos de re-roteamento MPLS e o método de re-roteamento IP convencional avaliado com somente o DiffServ na rede. O método L3 *Control Driven* restabeleceu o tráfego em outro caminho mais rapidamente que o método proposto por Haskin.

Capítulo 6

Conclusões Gerais

A Internet hoje trabalha com o serviço de melhor esforço, onde cada usuário da rede envia seus dados e compartilha a largura de banda com todos os fluxos de dados dos outros usuários. Com o surgimento de novas aplicações multimídia e em tempo real, com requisitos diferentes das aplicações tradicionais e rígidos em relação aos parâmetros de QoS, sinaliza a necessidade de serviços adicionais a serem oferecidos pela Internet. A garantia dos parâmetros de QoS é indispensável para a viabilização das aplicações multimídia na Internet, uma vez que estas são sensíveis aos rígidos parâmetros de QoS.

Várias pesquisas estão sendo desenvolvidas para que seja possível implementar QoS na Internet. Dentre elas, conforme apresentado, tem-se a arquitetura de Serviços Integrados (IntServ)[2], a arquitetura de Serviços Diferenciados (DiffServ) [3], a Engenharia de Tráfego (TE) [4] [6] e o encaminhamento baseado em rótulos (MPLS) [5].

O modelo Intserv permite um controle mais fino dos recursos do provedor de serviços, a um custo maior em termos de processamento e de memória necessários à manutenção de

estados por fluxo. O modelo Diffserv possibilita a gerência de recursos com uma granularidade menor, em termos de agregado de fluxos, permitindo uma maior escalabilidade.

A tecnologia MPLS surge para o suporte à engenharia de tráfego com a provisão de serviços com qualidade. Ao se utilizar recursos de forma eficiente, minimizando-se a ocorrência de gargalos e congestionamentos, amplia-se a quantidade de usuários atendidos, bem como a qualidade do serviço prestado. Um dos principais desafios da engenharia de tráfego consiste no mapeamento otimizado dos fluxos de tráfego dos usuários aos recursos disponíveis no provedor. Em um domínio MPLS, este mapeamento é realizado através da inserção dos referidos fluxos nos LSPs definidos no provedor. A associação entre fluxos de tráfego a LSPs é realizada pelos LSRs de borda através da adição de rótulos aos pacotes que compõem os referidos fluxos. A manutenção de LSPs de contingência (*backup* LSPs) contribuem para o aumento da confiabilidade fornecida pelo domínio MPLS. A pré-configuração destes LSPs torna possível a implementação de mecanismos de recuperação bastante eficientes.

Com a possibilidade de estabelecer caminhos explícitos, o MPLS torna-se uma promissora ferramenta para implementar a engenharia de tráfego, o que melhora efetivamente o desempenho e escalabilidade dos ISPs e das redes *backbone*, além de facilitar o suporte de novos serviços, tais como as aplicações em tempo real. Desta forma a possibilidade de ligar os rótulos a classes de equivalência de encaminhamento (FECs), se apresenta como um grande precursor da arquitetura DiffServ que atuando em conjunto com o MPLS, permite no roteador de borda de um domínio DiffServ - MPLS o mapeamento das classes de serviço DiffServ em FECs e os fluxos de dados ao longo do domínio são tratados a

partir da leitura do rótulo pelos roteadores interiores ao domínio.

O principal objetivo do MPLS é agilizar o roteamento e o balanceamento de carga através da engenharia de tráfego. Como ele não oferece qualidade de serviço diretamente, necessita de outro mecanismo para prover QoS. Assim o DiffServ é o mecanismo de provisão de QoS mais indicado para ser utilizado juntamente com o MPLS por ser bastante escalável e possuir características em comum com o MPLS.

Neste trabalho, considerou-se um ambiente de rede, onde os tráfegos multímidia oriundos de um ISP local são classificados na borda da rede *backbone* em sub-classes de tráfego definidas pelo DiffServ. Em seguida, estes tráfegos são mapeados em diferentes LSPs com diferentes características de vazão, atraso e *jitter*. Assim, o estudo baseado em modelagem e simulação da interoperabilidade entre a arquitetura MPLS e DiffServ destaca-se os E-LSPs e L-LSPs. A necessidade de integração DiffServ e MPLS foi verificada por meio de resultados de simulações, destacando-se a engenharia de tráfego MPLS e o aspecto do re-roteamento para explicar a tal integração.

A avaliação foi feita em relação à vazão alcançada pelos diferentes tipos de pacotes, ao atraso fim-a-fim experimentado pelos pacotes e ainda em relação à variação do atraso da chegada dos pacotes ao nó destino (*jitter*), em diferentes situações. Desta forma avaliou-se, através de experimentos baseados em simulação, vários ambientes de rede diferentes. Para um ambiente de rede sem nenhum mecanismo de QoS implementado, os limites requeridos pelas aplicações multimídia foram ultrapassados. Num ambiente de rede onde se implementou o mecanismo DiffServ, observou-se o atendimento parcial aos requisitos de QoS. Para um ambiente com somente o mecanismo MPLS na rede os

resultados equiparam-se com sem nenhum mecanismo de QoS implementado devido a não provisão de QoS da arquitetura. No último experimento, onde se combinou os mecanismos MPLS e DiffServ no *backbone*, classificando o fluxo de tráfego em diferentes LSPs com seus respectivos requisitos de QoS, pode-se observar que o tráfego de voz e vídeo obtiveram os limites exigidos pelas aplicações sensíveis ao atraso.

Além disso, realizou-se uma comparação entre os métodos de re-roteamento MPLS e re-roteamento IP convencional incluindo somente o DiffServ na rede. O método L3 *Control Driven* foi capaz de restabelecer o tráfego mais rapidamente que o método proposto por Haskin.

Para trabalhos futuros, sugere-se uma implementação de negociação dinâmica de SLA entre Sistemas Autônomos. Além disso, uma modelagem mais aproximada da Internet com tráfegos reais.

Referências Bibliográficas

- [1] XIAO, X. AND NI, L. Internet QoS: A Big Picture. IEEE Network, vol. 13, no. 2, pp. 1-13, March 1999.
- [2] BRADEN, R., CLARK, D. D. E SHENKER, S. Integrated Services in the Internet Architecture: An Overview. Internet RFC 1633, June 1994.
- [3] BLAKE, S. et all. An Architecture for Differentiated Services. Internet RFC 2475, December 1998.
- [4] AWDUCHE, O. D., et all. A Framework for Internet Traffic Engineering. Internet draft, draft-ietf-tewg-framework-03.txt, March 2001.
- [5] ROSEN, E., VISWANATHAN, A. AND CALLON, R. Multiprotocol Label Switching Architecture. Internet RFC 3031, January. 2001.
- [6] AWDUCHE et all, Requirements for Traffic Engineering over MPLS, Internet RFC 2702, September, 1999.

- [7] JACOBSON, V. and NICHOLS, K. An Expedited Forwarding PHB. Internet RFC 2598, June 1999.
- [8] HEINAMEN, J., et all, Assured Forwarding PHB Group. Internet RFC 2597, June 1999.
- [9] CARPENTER, B, NICHOLS, K, Differentiated Services in the Internet, Proceedings of the IEEE, Vol 90, No 9, September, 2002.
- [10] ARMITAGE, G., Quality of Service in IP Networks, Morgan Kaufmann, Indianapolis, IN, 2000.
- [11] BRADEN, R., et all Resource Reservation Protocol (RSVP)-version 1 Functional Specification. Internet RFC 2205, September 1997.
- [12] ANDERSSON, L., et all. LDP Specification. IETF Request for Comments (RFC) 3036. Janeiro de 2001
- [13] STARDUST, C., The Need for QoS, White Paper, Julho 1999.
- [14] SHENKER, S., GUERIN, R., PARTRIDGE, C. Specification of Guaranteed Quality of Service. IETF Request for Comments (RFC) 2212. Setembro de 1997
- [15] WROCLWSKI, J. Specification of the Controlled-Load Network Element Service. IETF Request for Comments (RFC) 2211. Setembro de 1997

- [16] REKHTER, Y. e ROSEN, E. Carrying Label Information in BGP-4. Trabalho em Desenvolvimento. IETF Working Draft draft-ietf-mpls-bgp4-mpls-05.txt. Janeiro de 2001
- [17] AWDUCHE, D., et all. RSVP-TE: Extensions to RSVP for LSP Tunnels. Trabalho em Desenvolvimento. IETF Working Draft draft-ietf-mpls-rsvp-lsp-tunnel-08.txt. Fevereiro de 2001
- [18] ANDERSSON, L., CAIN, B., JAMOUSSI, B. Requirement Framework for Fast Re-route with MPLS. Trabalho em Desenvolvimento. IETF Working Draft draftandersson-reroute-frmwrk-00.txt. Outubro de 1999
- [19] KAMIENSKI, C. A. e SADOK D, Qualidade de Serviço na Internet, 18º Simpósio Brasileiro de Redes de Computadores, SBRC, Belo Horizonte, maio, 2000.
- [20] Stardust.com. QoS protocols e architecture, QoS Forum White Paper, July 1999.
- [21] JAMOUSSI, B, et all, Constraint-based LSP Setup using LDP, IETF RFC 3212, January 2002.
- [22] JAMOUSSI, B., et all, IP Traffic Engineering Using MPLS Explicit Routing in Carrier Networks: Between the Signaling Approaches CR-LDP and RSVP, Nortel Networks white paper, 2000.

- [23] MAGALHÃES, M. e CARDOZO, E. Introdução à Comutação IP por Rótulos através de MPLS, 19º Simpósio Brasileiro de Redes de Computadores, SBRC, Florianópolis, maio, 2001.
- [24] DAVIE, B. and REKHTER, Y., MPLS Technology and Applications, Morgan Kaufmann, San Francisco, CA, 2000.
- [25] CRAWLEY, E., et all, A Framework for QoS-based Routing in the Internet, RFC 2386, Agosto de 1998.
- [26] XIAO, X., Providing Quality of Service in the Internet, Tese de Doutorado, Department of Computer Science and Engineering, Michigan State University, Março de 2000.
- [27] OWENS, K. et all, A Path Protection/Restoration Mechanism for MPLS Networks, work in progress, draft-chang-mpls-path-protection-02, Nov. 2000.
- [28] YASUHIRO K. (Toshiba), YOSHIHIRO O. (Toshiba), NAGAMI N. (Toshiba), Two Modes of MPLS Explicit Label Distribution Protocol, draft-katsube-mpls-two-ldp-00.txt, work in progress, Sept 1997.
- [29] HASKIN, D. A Method for Setting an Alternative Label Switched Paths to Handle Fast Reroute, draft-haskinmpls-fast-reroute-05, work in progress, May 2001.
- [30] NICHOLS, K. et all, Definition of the Differentiated Services Field (DS Byte) in the Ipv4 and Ipv6 Headers. Internet RFC 2474, December 1998.

- [31] HEINANEN, J., et all, A Three Color Marker, Internet draft, draft-heinanen-diffserv-tcm-01.txt, February 1999.
- [32] HEINANEN, J., et all, A Two Rate Three Color Marker. Internet RFC 2698, September 1999.
- [33] FONSECA, M., AGOUMINE, N. End to End IP QoS Assurance Using Policy Based Multi-Agents SLA Management. Revista da Sociedade Brasileira de Telecomunicações, Vol 18, pp. 23-33, Junho, 2003.
- [34] WESTERINEN, A. et all, Terminology for policy-based management, Network Working Group, RFC 3198, November 2001.
- [35] CHIU, J. et all, Supporting End-to-End QoS in DiffServ/MPLS Networks, IEEE Communication Magazine, pp.261-266, Jan 2003.
- [36] LE FAUCHEUR, F., LAI, W., Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering, Internet RFC 3564, July 2003.
- [37] MURPHY, S., Network configuration of a DiffServ network carrying VPN traffic, <http://www.teltec.dcu.ie/~murphys/mypapers/public/>, disponível em 12/09/2002.
- [38] XIAO, X., HANNAN, A., BAILEY, B., Traffic Engineering with MPLS in the Internet, IEEE Network, March/April, 2000.
- [39] JAIN, R., SUN, W., and BHANIRAMKA P., Quality of Service using Traffic Engineering over MPLS: An Analysis, IETF draft-bhani-mpls-te-anal-00.txt, March 1999.

- [40] LE FAUCHEUR, F. et all, Multi-Protocol Label Switching (MPLS) Support of Differentiated Services, Internet RFC 3270, May 2002.
- [41] GROSSMAN, D., New Terminology for DiffServ, IETF Internet Draft, draft-ietf-diffserv-new-terms-04.txt, March 2001.
- [42] CLARK, D., et all, Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism, SIGCOMM, 1992.
- [44] NS-2, The Network Simulator <http://www.isi.edu/nsnam/ns>, disponível em 05/11/2003.
- [45] GREIS, M. Tutorial for the Network Simulator ns, VINT Project, <http://www.isi.edu/nsnam/ns/tutorial/index.html>, disponível em 05/11/2003.
- [46] AHN, G. and CHUN, W., Overview of mpls network simulator: Design and implementation, <http://flower.ce.cnu.ac.kr/fog1/mns/index.html>, disponível em 25/04/2002.
- [47] MURPHY, S. The ns MPLS/DiffServ patch, <http://www.teltec.dcu.ie/~murphys/ns-work/mpls-DiffServ/> Apr 2001.
- [48] RAGHAVAN, S., LAW, R., DiffServ and MPLS - Concepts and Simulation Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, 2001.

- [49] SAAD, T., et all, DiffServ-enable adaptative traffic engineering over MPLS, ICII, Volume 2 pp. 128-133, 2001.
- [50] MOH, M., WEI, B., ZHU, J. H. Supporting differentiated services with per-class traffic engineering in MPLS, Internetional Conference on Computer Communications and Networks, pp. 354-360, 2001.
- [51] ROUHANA, N., HORLAIT, E., Differentiated services and integrated services use of MPLS, ISCC, pp. 194-199.
- [52] RAGHAVAN, S, An MPLS-based Quality of Service Architecture for Heterogeneous Networks Faculty of the Virginia Polytechnic Institute and State University, November, 2001, 107 p.
- [53] SHAHSAVARI, M.M., AL-TUNSI, A.A., MPLS Performance Modeling Using Traffic Enginnering To Improve QoS Routing on IP Networks, Proceedings IEEE Southeast-Con, pp. 152-157, 2002.
- [54] KESHAV, S., Engineering Approach to Computer Networking, An: ATM Networks, the Internet, and the Telephone Network, Addison-Wesley, May, 1997.

Trabalhos Publicados pelo Autor:

- [55] OLIVEIRA JR., A.C. e GUARDIEIRO, P.R. QoS fim-a-fim para aplicações multimídia com DiffServ e Engenharia de Tráfego baseada em MPLS, Congresso Nacional de Tecnologia da Informação e Comunicação, SUCESU'2003, Salvador-BA, Brasil, 22 a 24 Abril 2003.
- [56] OLIVEIRA JR, A.C. e GUARDIEIRO, P.R. Um Estudo sobre QoS fim-a-fim para aplicações multimídia em Redes MPLS com DiffServ e Engenharia de Tráfego. Aceito para publicação na 2da. Conferencia Iberoamericana en Sistemas, Cibernética e Informática, CISCI 2003, Orlando, Florida (EE.UU.), 31 de Julho a 02 de Agosto 2003.
- [57] OLIVEIRA JR, A.C and GUARDIEIRO, P.R. End-to-end Quality of Service for Multimedia Applications over IP Networks with DiffServ and MPLS Traffic Engineering. Accepted for publication in The 7th World Multiconference on Systemics Cybernetics and Informatics, SCI 2003, Orlando, Florida, USA, July 27-30, 2003.
- [58] OLIVEIRA JR, A.C. e GUARDIEIRO, P.R. Um Estudo sobre a Obtenção de QoS Fim-a-Fim para Aplicações Multimídia em Redes IP com DiffServ e Engenharia de Tráfego MPLS. XX Simpósio Brasileiro de Telecomunicações - SBrT 2003, Rio de Janeiro, Brasil, ISBN 85-89748-01, 04 a 08 de Outubro de 2003.
- [59] OLIVEIRA JR, A.C. and GUARDIEIRO, P.R. Quality of Service Over DiffServ and Traffic Engineering with MPLS, Accepted for publication in IASTED International

Conference on Communication Systems and Networks (CSN 2003), Benalmadena, Spain, September 08-10, 2003.

- [60] OLIVEIRA JR, A.C. and GUARDIEIRO, P.R. A Study of End-to-End Quality of Service for Real Time Applications over MPLS Networks with Traffic Engineering and DiffServ, Accepted for publication in IEEE International Conference on Software, Telecommunications and Computer Networks (SOFTCOM 2003), Venice, Italy, October 07-10, 2003.