Alocação de Banda Passante em Redes Auto-Ajustáveis

André Costa Drummond

Dissertação de Mestrado

Alocação de Banda Passante em Redes Auto-Ajustáveis

André Costa Drummond¹

Fevereiro de 2005

Banca Examinadora:

- Nelson Luis Saldanha da Fonseca (Orientador)
- Akebo Yamakami Departamento de Telemática Faculdade de Engenharia Elétrica e Computação, Unicamp
- Edmundo R. M. Madeira Instituto de Computação, Unicamp
- Flávio Keidi Miyazawa (Suplente) Instituto de Computação, Unicamp

 $^{^1 \}mathrm{Trabalho}$ parcialmente financiado pela CAPES

INIDADE	<u> 00</u>
Nº CHAMAI	DA
v	EX
TOMBO BC	1 68257
PROC 16	123-06
с	D X
PPECO I	

DATA 04/05/06

Nº CPD

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP

Bibliotecário: Maria Júlia Milani Rodrigues - CRB8a / 2116

Drummond, André Costa

D844a Alocação de banda passante em redes auto-ajustáveis / André Costa Drummond -- Campínas, [S.P. :s.n.], 2005.

Orientadores : Nelson Luis Saldanha da Fonseca; Mihail Devetsikiotis Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação.

 Redes de computadores. 2. Otimização matemática. 3. Conjuntos difusos. I. Fonseca, Nelson Luis Saldanha da. II. Devetsikiotis, Mihail. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Título em inglês: Bandwidth allocation in self-sizing networks

Palavras-chave em inglês (keywords): 1. Computer networks. 2. Mathematical optimization. 3. Fuzzy sets.

Área de concentração: Sistemas de Computação

Titulação: Mestre em Computação

Banca examinadora: Prof. Dr. Nelson Luis Saldanha da Fonseca (UNICAMP) Prof. Dr. Akebo Yamakami (UNICAMP) Prof. Dr. Flavio Keidi Miyazawa (UNICAMP) Prof. Dr. Edmundo R. M. Madeira (UNICAMP)

Data da defesa: 28/02/2005

Alocação de Banda Passante em Redes Auto-Ajustáveis

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por André Costa Drummond e aprovada pela Banca Examinadora.

Campinas, 28 de fevereiro de 2005.

Nelson Luis Saldanha da Fonseca (Orientador)

Mihail Devetsikiotis (Co-orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação. Tese defendida e aprovada em 28 de fevereiro de 2005, pela Banca examinadora composta pelos Professores Doutores:

Prof. Dr. Akebo Yamakami

FEEC - UNICAMP

Prof. Dr. Flávio Keidi Miyazawa IC - UNICAMP

Prof. Dr. Nelson Luis Saldanha da Fonseca IC - UNICAMP

© André Costa Drummond, 2006. Todos os direitos reservados.

A minha família, Elaine, Ana, Armando e Helena

Agradecimentos

Eu gostaria de agradecer primeiramente ao meu orientador, Prof. Nelson Fonseca, pelo suporte essencial.

Ao Prof. Akebo Yamakami, pelas grandes dicas que permitiram a evolução dos meus modelos matemáticos.

Aos meus amigos da pós-graduação do Instituto de Computação, em especial a Juliana e ao Evandro que, pacientemente, responderam um monte de perguntas.

A minha família, Ana, Armando, Helena e em especial minha companheira Elaine.

Resumo

Nesta dissertação, apresenta-se o *framework* de Redes Auto-Ajustáveis, como sendo um conjunto de técnicas capazes de prover Qualidade de Serviço (QoS), em redes multiserviço de alta velocidade. A implantação deste *framework* permitirá a introdução de novas aplicações multimídia com Qualidade de Serviço, nas redes *backbone* existentes.

Para se garantir os requisitos de QoS de um fluxo, estima-se a quantidade de banda passante requerida por este. A utilização de estimadores de banda passante equivalente baseados em medições é altamente recomendável dado que estes usam amostras de tráfego real para estimar a banda passante de um fluxo. Experimentos com tráfego sintético e real foram realizados para se comparar alguns dos principais estimadores presentes na literatura. Os estimadores *Gaussian Approximation Allocation* (GA) e o estimador baseado no *Processo Envelope do Movimento Browniano Fractal* (FEP) foram estudados em detalhes. O estimador FEP destaca-se pela sua capacidade de efetuar estimativas, em tempo-real, para diferentes classes de tráfego, e por considerar as dependências de longa duração.

Para se garantir uma boa alocação dos caminhos na rede, e das capacidades de banda passante nos enlaces, são necessárias a definição, e posterior resolução de um problema de otimização de fluxos multiproduto. Para a solução deste problema, foram estudadas técnicas de otimização fuzzy, que demonstraram uma capacidade maior de se adaptar a variabilidade, e aos aumentos de demanda, existentes no tráfego de uma rede de alta velocidade. Foram realizados experimentos que simularam redes auto-ajustáveis multiserviço de alta velocidade. A abordagem proposta supera, em todas as avaliações, os resultados do modelo clássico, encorajando a utilização de técnicas fuzzy na otimização de redes auto-ajustáveis.

Abstract

On this dissertation, a Self-Sizing Network Framework is capable of provide Quality of service (QoS), in high-speed multiservice networks, is presented. The implementation of this framework will allow the introduction of new multimedia applications with Quality of Service on current backbone networks.

In order to provide QoS to a flow the amount of bandwidth required is estimated. The utilization of measurement based effective bandwidth estimators is highly recommended considering that those use only samples of real traffic to estimate the effective bandwidth of a flow. Some of the main estimators were compared using synthetic and real traffic. The *Gaussian Approximation Allocation* (GA) estimator and the *Fractal Brownian Motion Envelope Process* (FEP) based estimator were analized.

To ensure adequate path selection and bandwidth allocation, it is required to define and solve a multicommodity flow optimization problem. To cope with that problem, fuzzy optimization technics were considered, enabling the network to cope with network traffic variability and dynamic bandwidth demand. Simulation experiments that simulate selfsizing high-speed multiservice networks were carried out. The proposed model provides better results than classic models, which encourages the use of fuzzy technics on self-sizing networks optimization.

Sumário

				viii
A	grade	ecimen	tos	ix
Re	esum	0		x
A	bstra	ct		xi
1	Intr	odução	0	1
2	Pro	blemas	s de Fluxo Multiproduto	4
	2.1	Rotea	mento de Múltiplos Produtos	5
	2.2	Fluxo	Multiproduto Inteiro	6
	2.3	Abord	agens de Solução para Problemas de Fluxo Multiproduto	7
3	Pro	grama	ção Linear Inteira	8
	3.1	Abord	agens Clássicas de Solução de Problemas de Programação Inteira	9
		3.1.1	Técnicas Cutting Plane	9
		3.1.2	Métodos Enumerativos	10
	3.2	Abord	agens Heurísticas de Solução de Problemas de Programação Inteira .	12
		3.2.1	Busca Tabu	13
		3.2.2	Simulated Annealing	13
		3.2.3	Algorítmos Genéticos	14
	3.3	Otimiz	zação Fuzzy	15
		3.3.1	Conjuntos Fuzzy	15
		3.3.2	Otimização com Objetivo e Restrições Fuzzy	17
4	Esti	mador	es de Banda Passante Equivalente	19
	4.1	Estima	adores de Banda Passante Equivalente	19
		4.1.1	Definição de Banda Passante Equivalente	20

		4.1.2 Uma comparação entre	Estimadores de Banda Passante Equivalente	20	
	4.2	Uma Comparação entre os Est	imadores FEP e GA	24	
		4.2.1 Definição dos Parâmetr	os para os Experimentos de Simulação	26	
		4.2.2 Estimador do Parâmetr	o de <i>Hurst</i>	27	
		4.2.3 Complexidade dos Esti	madores	28	
		4.2.4 Procedimentos Utilizad	os Para a Validação dos Estimadores de Banda		
		Passante Equivalente .		29	
	4.3	Resultados Numéricos		30	
		4.3.1 Experimentos de Simul	ação com Tráfego Sintético de Voz	31	
		4.3.2 Experimentos de Simul	ação com Tráfego de Vídeo	36	
		4.3.3 Experimentos de Simul	ação com Tráfego de Dados	40	
		4.3.4 Experimento de Simula	ção com Tráfego Sintético de Dados	44	
	4.4	Avaliação dos Resultados		48	
5	Red	des Auto-Ajustáveis		49	
	5.1	Redes Multiserviço de Alta Ve	locidade	49	
		5.1.1 Redes ATM \ldots .		50	
	5.2	Redes Auto-Ajustáveis		51	
		5.2.1 Modelo para o Particio	namento Ótimo de Banda	53	
6	Técnicas Fuzzy na Otimização de Redes Auto-Ajustáveis 5				
	6.1	Configuração dos Experimento	s de Simulação	56	
		6.1.1 Topologia de Rede Util	izada nos Experimentos de Simulação	57	
		6.1.2 Características do Tráf	ego Utilizado nos Experimentos de Simulação	59	
		6.1.3 Dinâmica dos Experim	entos de Simulação	60	
	6.2	Modelos Propostos para o Pro	blema do Particionamento Ótimo de Banda .	63	
		6.2.1 Primeiro Modelo Fuzzy		65	
		6.2.2 Segundo Modelo Fuzzy		67	
		6.2.3 Terceiro Modelo Fuzzy		68	
		6.2.4 Quarto Modelo Fuzzy.		70	
		6.2.5 Complexidade dos Mod	elos de Otimização	71	
	6.3	Resultados Numéricos		73	
		6.3.1 Estatísticas dos Cenário	os Utilizados nos Experimentos de Simulação	73	
		6.3.2 Avaliação dos Modelos	de Otimização Propostos	79	
		6.3.3 Resumo dos Resultados	Obtidos	99	
7	Con	onclusões	1	.03	
7	Con 7.1	onclusões Propostas de Trabalhos Future	1 95	. 03 105	

Bibliografia

Lista de Tabelas

4.1	Comparação de Desempenho de Estimadores de Banda Passante Equivalente	22
4.2	Parâmetros do sistema de avaliação dos estimadores	26
4.3	Ponto de intersecção das curvas dos estimadores FEP e GA em função de	
	H para o cenário padrão	27
4.4	Complexidade dos Estimadores de Banda	29
4.5	Variação do número de fontes ativas de voz.	35
4.6	Variação do número de traços de video	39
4.7	Variação da taxa de transmissão de dados	43
4.8	Variação do número de fontes ativas de dados	47
6.1	Pares de nós origem-destino e seus possíveis caminhos de interligação	58
6.2	Parâmetros das distribuições Exponenciais de chegada e de duração, para	
	as fontes, de cada classe de tráfego	59
6.3	Número máximo de fontes ativas e taxa máxima de transmissão, por fluxo,	
	de cada classe de tráfego	60
6.4	Tempos de execução dos modelos matemáticos, para três topologias distintas.	72
6.5	Número de fontes ativas nos dois níveis do trafego "onda quadrada", por	
	fluxo, de cada classe de tráfego	76
6.6	Taxas médias de utilização, perda e retardo, para todos os modelos de	
	otimização avaliados.	100
6.7	Desvio padrão e 90° percentil entre os valores da taxa média de perda e de	
	retardo dos pares origem-destino, para o quarto modelo fuzzy e o modelo	
	clássico.	102

Lista de Figuras

2.1	Rede direcionada com custos e capacidades nos arcos	6
3.1 3.2 3.3 3.4 3.5	Solução de um MIP pela técnica Cutting Plane	10 12 16 16 17
4.1	Cenário da simulação para avaliação dos estimadores	25
4.2	Logscale Diagram para um tráfego LRD com o parâmetro $H = 0.8 \dots$	28
4.3	Configuração básica do NS	31
4.4	Banda Efetiva v s Fatia de Tempo para os estimadores FEP e GA para o	
	tráfego de voz \ldots	33
4.5	Taxa média e banda passante estimada por FEP e GA, para o tráfego de voz	34
4.6	Taxa média e banda passante estimada por FEP e GA, para o tráfego de	
	voz, em cenários opostos	35
4.7	Taxa média e banda passante estimada por FEP e GA, para o tráfego de voz, na presença de variação do tráfego	36
4.8	Banda Efetiva v s Fatia de Tempo para os estimadores FEP e GA para o	
	tráfego de video	37
4.9	Taxa média e banda passante estimada por FEP e GA, para o tráfego de	
	video	38
4.10	Taxa média e banda passante estimada por FEP e GA, para o tráfego de	20
4 1 1	video, em cenários opostos	39
4.11	Taxa media e banda passante estimada por FEP e GA, para o tratego de	40
1 19	Panda Efetiva va Estis de Tempo para es estimadores EED e CA para e	40
4.12	tráfego de dados	<u>⁄</u> 1
4 13	Taxa média e handa passante estimada por FEP e GA para o tráfego de	-11
1.10	dados	42

4.14	Taxa média e banda passante estimada por FEP e GA, para o tráfego de	
	dados, em cenários opostos	43
4.15	Taxa média e banda passante estimada por FEP e GA, para o tráfego de dados, na presença de variação do tráfego	43
4.16	Banda Efetiva vs Fatia de Tempo para os estimadores FEP e GA para o tráfero sintético de dados	45
4.17	Taxa média e banda passante estimada por FEP e GA, para o tráfego	10
4.18	Taxa média e banda passante estimada por FEP e GA, para o tráfego	40
	sintético de dados, em cenários opostos	47
4.19	Taxa média e banda passante estimada por FEP e GA, para o tráfego sintético de dados, na presença de variação do tráfego	48
5.1	Visão geral do <i>Framework</i> das Redes Auto-Ajustáveis	53
6.1	Topologia da Rede	58
6.2	Configuração do Experimento de Simulação	61
6.3	Função de pertinência para a minimização do custo total da rede	65
6.4	Função de pertinência para a maximização da quantidade de banda pas-	
	sante alocada por fluxo	67
6.5	Função de pertinência para a minimização do número de fluxo de cada	
	classe de serviço, por enlace	69
6.6	Variação da Taxa Média, Desvio Padrão, Parâmetro de Hurst e Banda	
	Passante Equivalente para o tráfego agregado de Voz	74
6.7	Variação da Taxa Média, Desvio Padrão, Parâmetro de Hurst e Banda	
	Passante Equivalente para o tráfego agregado de Vídeo	75
6.8	Variação da Taxa Média, Desvio Padrão, Parâmetro de Hurst e Banda	70
<i>c</i> 0	Passante Equivalente para o tralego agregado de Dados $\dots \dots \dots$	70
0.9	Variação da Taxa Media, Desvio Padrão, Parametro de Hurst e Banda Passanto Equivalente para o tréfero "ondo quadrado" do Voz	77
6 10	Variação da Tava Módia, Deguio Padrão, Parâmetro do Hurst o Pando	11
0.10	Passante Equivalente para o tráfego "onda quadrada" de Vídeo	78
6 11	Variação da Taxa Média Desvio Padrão Parâmetro de <i>Hurst</i> e Banda	10
0.11	Passante Equivalente para o tráfego "onda quadrada" de Dados	78
6.12	Variação da taxa de utilização da rede, obtida com o modelo de otimização	
	clássico	80
6.13	Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
	o modelo de otimização clássico	81

Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o modelo de otimização clássico.	82
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o modelo de otimização clássico, para o tráfego "onda quadrada"	83
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o modelo de otimização clássico, para o tráfego "onda quadrada"	84
Variação da taxa de utilização da rede, obtida com o primeiro modelo de	
otimização fuzzy.	85
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o primeiro modelo de otimização fuzzy	86
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o primeiro modelo de otimização fuzzy.	87
Variação da taxa de utilização da rede, obtida com o segundo modelo de	
otimização fuzzy.	87
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o segundo modelo de otimização fuzzy	88
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o segundo modelo de otimização fuzzy	89
Variação da taxa de utilização da rede, obtida com o terceiro modelo de	
otimização fuzzy.	90
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o terceiro modelo de otimização fuzzy. \hdots	91
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o terceiro modelo de otimização fuzzy	92
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o terceiro modelo de otimização fuzzy, para o tráfego "onda quadrada". $\ .$	93
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o terceiro modelo de otimização fuzzy, para o tráfego "onda quadrada".	94
Variação da taxa de utilização da rede, obtida com o quarto modelo de	
otimização fuzzy.	95
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o quarto modelo de otimização fuzzy	96
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o quarto modelo de otimização fuzzy.	97
Variação da Taxa média de Perda de Pacotes e Retardo Médio, obtida com	
o quarto modelo de otimização fuzzy, para o tráfego "onda quadrada". $\ $.	98
Taxa média de Perda de Pacotes por par origem-destino da rede, obtida	
com o quarto modelo de otimização fuzzy, para o tráfego "onda quadrada".	99
	Taxa média de Perda de Pacotes por par origem-destino da rede, obtida com o modelo de otimização clássico

Capítulo 1 Introdução

Na maioria das redes *backbone*, a configuração de rotas e a alocação de bandas para serviços são feitas de forma estática. Hoje em dia, as detentoras de redes *backbones* com o maior volume de tráfego são, em geral, as intituições provedoras de acesso à Internet. Estas atendem inúmeros clientes, que, por sua vez, utilizam uma gama de aplicações variada incluindo serviços de voz, vídeo e dados. A forma utilizada por estas instituições para atender a todos os seus clientes e as suas aplicações tem sido a implantação de redes pontoa-ponto (ATM) superdimensionadas, ou seja, a utilização de enlaces e *switchs* operando com larguras de banda superiores à exigida pelo tráfego servido.

E sabido que as taxas de utilização em redes *backbone* são pequenas. Atribui-se a este fato a baixa velocidade existente atualmente nas redes de acesso ou *Last-mile Technology*, porém este problema já vem sendo endereçado, o que permitirá que os clientes, num futuro próximo, tenham um acesso a rede com taxas na ordem de centenas de megabits por segundo. Tal realidade cria um novo mercado para novas aplicações que consumirão banda passante de forma intensa. Logo, os dias de calmaria nas redes de núcleo estão contados. Os administradores não poderão mais contar com a baixa utilização de suas redes, e portanto serão obrigados a otimizar o acesso a seus recursos.

Tendo em vista esta realidade, um *framework* para a implementação de redes pontoa-ponto com comutação por circuito capazes de se ajustarem dinâmicamente as características do tráfego é apresentado nesta dissertação. Estas redes, chamadas de redes auto-ajustáveis [41], [28] e [29], possuem a capacidade de estimar a quantidade de banda requerida por cada fluxo de aplicações, de forma a suprir a Qualidade de Serviço (QoS) demandada.

O *framework* divide-se, basicamente, em quatro tarefas principais: coleta das estatísticas dos fluxos origem-destino em tempo-real; estimação da banda passante requerida por cada fluxo; resolução do problema de particionamento de banda e roteamento e, finalmente, a utilização de uma metodologia de controle para efetivar a nova configuração da rede.

Nesta dissertação, são realizados estudos referentes às etapas de estimação de banda e de otimização da rede. Assume-se que todos os nós são acessíveis diretamente, possibilitando sua configuração imediata. Técnicas de coleta passiva de estatísticas da rede, [25], [42] e [10] estão fora do escopo deste texto.

Uma parte vital do *framework* é o uso de estimadores de banda passante equivalente [11], que são ferramentas indispensáveis no contexto das redes alto-ajustáveis. Com eles podemos estimar a quantidade de banda passante que um determinado fluxo demanda para que seus requisitos de Qualidade de Serviço sejam satisfeitos. Neste trabalho, foram utilizados estimadores de banda passante equivalente baseados em medidas. Tais ferramentas se utilizam estatísticas dos fluxos (e.g. taxa média e desvio padrão), e possuem baixa complexidade de implementação, permitindo que a banda passante equivalente seja calculada em tempo-real.

Tendo as estimativas de banda passante para os fluxos existentes, o passo seguinte é a reconfiguração da rede, que busca minimizar o custo total, a taxa de perda de pacotes e o retardo médio entre chegadas. Para tal, é necessário definir-se um modelo de otimização que possa ser resolvido em tempo hábil, de forma que a rede seja reconfigurada a cada poucos segundos. Esta realocação envolve a redistribuição de banda passante para todos os tipos de fluxos existentes em cada par de comunicação da rede, além de definir os caminhos na rede, que serão utilizados por estes fluxos.

O framework de redes auto-ajustáveis exige que a rede seja reconfigurada de tempos em tempos para que a mesma possa adaptar-se às características do tráfego servido. Quanto menor for o tempo entre realocações melhor será o grau de adaptabilidade da rede. No entanto, todo o processo de avaliação e realocação da rede implica em um overhead de processamento muito grande. Logo, o estudo de novas técnicas de otimização, que permitam a diminuição do número de realocações, se faz necessário. Neste contexto, estuda-se a aplicação de técnicas de otimização fuzzy, como uma forma de propiciar um relaxamento nas restrições impostas pelo problema. Com isto, busca-se a obtenção de resultados mais robustos, capazes de acomodar a variabilidade inerente ao tráfego em redes de alta velocidade.

As contribuições desta dissertação são:

• Um estudo sobre banda passante equivalente, no qual são avaliados vários estimadores presentes na literatura. Realiza-se, também, uma análise detalhada, a partir de experimentos de simulação com tráfego real e sintético de voz, vídeo e dados de dois estimadores baseados em medidas: o estimador Gaussian Approximation Allocation e o estimador baseado no framework do Processo Envelope para o Movimento Browniano Fractal. • Avaliação da adequabilidade de técnicas de otimização *fuzzy* em redes auto-ajustáveis, a fim de se comprovar o aumento na eficiência dos algoritmos de realocação da rede, considerando as incertezas e imprecisões impostas pelo ambiente das redes auto-ajustáveis. Utiliza-se um modelo matemático de fluxo multiproduto, inspirado no modelo proposto em [17]. Em seguida, introduz-se a incerteza no problema através de restrições *fuzzy*. Propõem-se quatro novos modelos de otimização com objetivos diferentes.

Os principais resultados relatados nesta dissertação são:

- A constatação de que o estimador de banda passante equivalente baseado no *framework* do *Processo Envelope para o Movimento Browniano Fractal* é um estimador capaz de lidar com tráfego de voz, vídeo e dados de forma robusta e eficiente. Por considerar o parâmetro de *Hurst* e o tamanho do *buffer* em sua estimativa, seus valores não superestimam banda passante, permitindo um melhor aproveitamento dos recursos da rede.
- A introdução de um modelo de otimização *fuzzy* para o problema de realocação da rede, que é mais robusto e capaz de balancear melhor o tráfego entre os pares de comunicação da rede, em relação ao modelo presente na literatura.

O texto dessa dissertação está organizado da seguinte forma: o Capítulo 2 apresenta a classe de problemas de Fluxo Multiproduto Inteiro. O Capitulo 3 descreve o problema de Programação Linear Inteira geral e fuzzy, e apresenta algumas abordagens para sua solução. O Capítulo 4 introduz os estimadores de banda passante equivalente e apresenta uma avaliação detalhada dos estimadores selecionados. No Capítulo 5, o *framework* de redes auto-ajustáveis é apresentado. No Capítulo 6, a configuração dos experimentos de simulação, para uma topologia de rede, é detalhada, e os seus resultados apresentados. Finalmente, no Capítulo 7, as conclusões são derivadas.

Capítulo 2 Problemas de Fluxo Multiproduto

Quando se refere a redes de computadores, pensa-se em várias entidades distintas interligadas de alguma forma possibilitando a interação entre todas elas. Hoje em dia, vivenciamos redes espalhadas por praticamente todos os lugares por onde passamos, como redes de eletricidade, que fornecem a energia para os estabelecimentos; redes de telefone, que nos permitem comunicar com qualquer outro telefone no mundo; redes de abastecimento de água, que distribuem água tratada até os bairros em uma cidade e redes de computadores como a *Internet*. Todas estas redes, têm em comum desafios que dependem em grande parte do tipo de serviço prestado, e da qualidade exigida para os mesmos.

O que se deseja é transportar um produto (*commodity*) de um ponto a outro de forma a prover um bom serviço para o usuário da rede e a utilizar a rede eficientemente. Neste capítulo, a classe de problemas chamada de *Multicommodity Flow* ou Fluxo Multiproduto, é apresentada.

Fluxo Multiproduto é um problema de *Fluxos em Rede* que é, por sua vez, um conjunto de problemas que aborda principalmente as seguintes questões:

- **Problemas de menor caminho** : Qual é a melhor forma de se atravessar uma rede de um ponto a outro com o menor esforço?
- **Problemas de fluxo máximo** : Em redes que possuem as capacidades dos arcos definidas, como se pode transportar a maior quantidade possível de produtos sem ultrapassar tais capacidades?
- **Problemas de fluxo de custo mínimo** : Como se pode enviar produtos com o menor custo possível?

Na realidade, as questões acima foram apresentadas considerando a existência de apenas um único fluxo de produtos, o que não ocorre em redes reais. No mundo real, lida-se frequentemente com uma quantidade muito grande de produtos, o que torna o problema ainda mais complicado, como é o caso em redes de computadores nas quais cada conexão existente na rede é um produto.

Considerando um problema de fluxo de custo mínimo com k fluxos de produtos, no qual estes fluxos nunca competem entre si por um recurso comum da rede (ex. enlace), pode-se então enxergá-lo como k problemas de fluxo de custo mínimo com apenas um único fluxo, sendo sua solução obtida através de k execuções de um algorítmo para o cálculo do fluxo de custo mínimo qualquer.

O problema no qual vários fluxos competem entre si por recursos da rede, chama-se problema de Fluxo Multiproduto. Em outras palavras, se dois ou mais produtos tiverem que utilizar um mesmo meio de transmissão ao mesmo tempo para chegarem aos seus destinos, é necessário compartilharem a capacidade deste meio de forma que o fluxo total de produtos não ultrapasse a capacidade do meio de transmissão.

Considere uma rede representada pelo grafo G(N, A) com a matriz de incidência \aleph , sendo N o conjunto de nós e A o conjunto de enlaces. Cada enlace (i, j) possui uma capacidade associada u_{ij} que restringe o fluxo total de todos os produtos neste arco. Considere x_{ij}^k como o fluxo do produto k no arco (i, j), e x^k e c^k como o vetor de fluxos e o vetor de custos por unidade para o produto k. O problema de Fluxo Multiproduto pode ser definido como [2]:

$$Min\sum_{1\le k\le K}c^kx^k$$

sujeito a

$$\sum_{1 \le k \le K} x_{ij}^k \le u_{ij} \qquad \forall (i,j) \in A,$$
(2.1)

$$\aleph x^k = b^k \qquad \forall k = 1, 2, \dots, K, \tag{2.2}$$

$$0 \le x_{ij}^k \le u_{ij}^k \qquad \forall (i,j) \in A \ e \ k = 1, 2, \dots, K.$$

A restrição acumulada 2.1 limita o fluxo total de todos os produtos no arco (i, j) ao valor de u_{ij} . A expressão 2.2 representa o conjunto de restrições que garantem os requerimentos para o fornecimento, a demanda e o roteamento do fluxo de produtos k = 1, 2, ..., K.

2.1 Roteamento de Múltiplos Produtos

Em uma rede de computadores, o problema de Fluxo Multiproduto é tipicamente definido como o problema de encontrar o conjunto de caminhos para os fluxos de mensagens entre pares origem-destino (*produto*), minimizando o custo total da rede, sendo que cada enlace possui um custo associado.



Figura 2.1: Rede directionada com custos e capacidades nos arcos

Considere o exemplo da Figura 2.1 [22]. Os parâmetros dos enlaces são definidos pelos pares (custo, capacidade). Deseja-se efetuar o transporte de dois fluxos entre os nós 1 e 7, a banda passante do Fluxo 1 é 30 e a do Fluxo 2 é 15.

Resolvendo o problema através de técnicas de programação linear, encontram-se os caminhos de menor custo na rede:

- Fluxo 1:
 - envia 20 unidades de banda passante pelo caminho $[1 \rightarrow 2 \rightarrow 6 \rightarrow 4 \rightarrow 7];$

- envia 10 unidades de banda passante pelo caminho $[1 \rightarrow 3 \rightarrow 6 \rightarrow 5 \rightarrow 7];$

- Fluxo 2:
 - envia 10 unidades de banda passante pelo caminho $[1 \rightarrow 3 \rightarrow 6 \rightarrow 5 \rightarrow 7];$
 - envia 5 unidades de banda passante pelo caminho $[1 \rightarrow 2 \rightarrow 4 \rightarrow 7];$

Tem-se, então, uma solução que divide os fluxos de forma a obter o menor custo de transporte das mensagens pela rede. Esta solução é chamada de contínua pois pode-se fracionar o fluxo de mensagens em diversas rotas da origem para o destino através da rede.

2.2 Fluxo Multiproduto Inteiro

O problema de multicommodity flow inteiro é uma versão do problema de multicommodity flow quando cada fluxo de mensagens usa apenas uma rota da origem ao destino, ou seja, não se pode fracionar o fluxo de *commodities*. A solução inteira também é solução para o caso contínuo¹, porém o contrário dificilmente ocorre [4]. Fluxos inteiros são típicos de redes de computadores com caminhos virtuais, como por exemplo, canais virtuais em redes ATM.

2.3 Abordagens de Solução para Problemas de Fluxo Multiproduto

O problema de Fluxo Multiproduto tradicional pode ser resolvido em tempo polinomial com metodos de programação linear como o *Simplex* [30] ou o Método *Interior Point* [5]. Devido ao fato de que nesta dissertação estudar-se-ão redes nos quais os fluxos devem seguir apenas um caminho, utiliza-se o problema de Fluxo Multiproduto Inteiro. Tal problema é NP-completo² [12] e pode ser resolvido com metodos de programação inteira. No próximo capítulo, diversos métodos de resolução para este tipo de problema serão descritos.

¹Porém, a solução não é necessariamente ótima.

²Formalmente intratável.

Capítulo 3 Programação Linear Inteira

Programação linear trata do problema de maximizar, ou minimizar, uma função linear em um espaço poliédrico restringido por demais funções lineares. O modelo linear inteiro geral é dado por:

$$Max(Min)\sum_{j\in B}c_jx_j + \sum_{j\in I}c_jx_j + \sum_{j\in C}c_jx_j$$

sujeito a

$$\sum_{j \in B} a_{ij} x_j + \sum_{j \in I} a_{ij} x_j \sum_{j \in C} a_{ij} x_j \sim b_i \qquad (i = 1, \dots, m)$$
$$l_j \le x_j \le u_j \qquad (j \in I \cup C)$$
$$x_j \in \{0, 1\} \qquad (j \in B)$$
$$x_j \in inteiros \qquad (j \in I)$$
$$x_j \in reais \qquad (j \in C)$$

B: conjunto de variáveis zeros ou uns;

I: conjunto de variáveis inteiras;

C: conjunto de variáveis contínuas;

- \sim : representa $\geq \leq ou =$ para as restrições $i = 1, \ldots, m$;
- l_j : limite inferior da variável x_j ;
- u_j : limite superior da variável x_j ;

Se houverem apenas variáveis binárias, o problema é referido como Problema de Programação (Linear) 0-1 Pura (BP). Se não houverem variáveis contínuas, trata-se de um Problema de Programação (Linear) Inteira Pura (IP). Caso contrário, chama-se Problema de Programação (Linear) Inteira Combinada ou Mista (MIP).

Se todas as variáveis fossem contínuas, teríamos um Problema de Programação Linear contínuo (LP), que pode ser resolvido em tempo polinomial. Na prática, problemas de programação linear podem ser resolvidos eficientemente para casos de tamanhos razoáveis, ou mesmo grandes problemas com estruturas especiais. No entanto, quando algumas ou todas as variáveis são inteiras o problema se torna NP-Completo (formalmente intratável).

3.1 Abordagens Clássicas de Solução de Problemas de Programação Inteira

A busca da solução ótima para problemas de Programação Linear Inteira Mista (MIP), pode ser alcançada por uma variedade de abordagens, sendo duas as mais conhecidas, as que utilizam técnicas *Cutting Plane* e as que utilizam métodos enumerativos.

3.1.1 Técnicas Cutting Plane

A idéia fundamental por trás das técnicas *Cutting Plane* (plano de corte) é de adicionar restrições a um problema até que uma solução ótima ocorra com valores inteiros. No entanto, deve-se ser cauteloso em relação às restrições que são adicionadas a fim de não alterar o problema original [13].

Para exemplificar o uso desta técnica, considera-se o Problema 3.1 de programação linear inteira mista:

$$max \quad 2x_1 + x_2 \tag{3.1}$$

 $\begin{array}{rcl}
x_1 + x_2 &\leq 5 \\
-x_1 + x_2 &\leq 0 \\
6x_1 + 2x_2 &\leq 21
\end{array}$

$$x_1, x_2 \ge 0, \quad x_1 \quad interio$$

Ao se adicionar determinadas restrições, torna-se possível remover a restrição de integralidade das variáveis. Desta forma, o novo problema torna-se contínuo todavia possuindo a mesma solução do problema inteiro.



Figura 3.1: Técnica Cutting Plane

No gráfico à esquerda da Figura 3.1, apresenta-se a solução gráfica para o Problema 3.1. As linhas verticais, pertencentes ao domínio da solução, estão evidenciadas, bem como a reta da função objetivo que maximiza a solução. A solução ótima deste problema é facilmente identificada como sendo $(x_1; x_2) = (3; 1, 5)$.

Ao adicionar a restrições $\frac{x_1}{2} + x_2 \leq 3$ e $x_1 \leq 3$ ao problema anterior, considerase todas as variáveis como sendo contínuas, obtendo-se um problema de programação linear contínuo que possui a mesma solução ótima do problema anterior. Em sua solução gráfica (gráfico à direita na Figura 3.1) as novas restrições interligam os pontos inteiros mais externos mantendo todos as possíveis soluções dentro do novo domínio. Sendo assim, pode-se calcular a solução ótima utilizando um algoritmo para problemas de programação linear contínua, como o *Simplex* [5]. De forma geral, a técnica de planos de corte deve ser implementada em passos, sendo que, a cada passo, adiciona-se um plano de corte que elimine um ponto "inviável" e resolve-se o novo problema.

3.1.2 Métodos Enumerativos

A intenção destes métodos é de enumerar, de forma explícita ou implícita, todas as possíveis soluções para o problema de programação inteira. A solução viável que maximizar a função objetivo é considerada ótima. O algoritmo de enumeração mantem o controle das enumerações, enquanto se define o caminho a ser seguido (algoritmo de ponto). A partir da característica de integralidade e das restrições do problema, o algoritmo de ponto busca pelo conjunto de pontos inteiros que leva à melhoria da solução. A eficiência dos métodos enumerativos depende fortemente da efetividade do algoritmo de ponto utilizado [32].

Para enumerar as possíveis soluções, o algoritmo de enumeração deve avançar fazendo iterações. Considere um problema de minimização. Na abordagem básica, *Branch and Bound*, cada iteração é composta dos seguintes passos [3]:

- 1. Inicialização: Define-se z^* , a melhor solução atual do MIP, com um valor arbitrariamente pequeno. O nó inicial, com todas as variáveis livres (contínuas), é o x^0 . Resolve-se então o LP, e se a solução ótima for inteira para as variáveis inteiras, o MIP está resolvido. Caso contrário, insere este nó na lista de ativos (inicialmente vazia), define-se $x^l = x^0$ e executa-se o Passo 2.
- 2. Branching: Escolhe-se um nó N da lista de ativos. Seja x a solução ótima deste nó. Escolhe-se uma variável x_k que não satisfaça a restrição de integralidade, digamos, com valor v. Define-se dois novos nós, a partir de N, inserindo a restrição $x_k \leq \lfloor v \rfloor$ em um deles e $x_k \geq \lceil v \rceil$ no outro, definindo dois novos nós abaixo de x^l . Resolve-se o subproblema em cada um destes nós. Caso a solução encontrada em um nó seja maior que z^* , aceita-se este nó. Caso contrário descarta-se o mesmo. Busca-se em cada nó válido por uma solução melhor para o MIP, se for encontrada redefine-se z^* , descartando-se todos os nós com soluções inferiores. Executa-se o Passo 3.
- Teste de Finalização: Se a lista atual de nós válidos for vazia, ou a solução ótima para o MIP, z*, foi encontrada, ou não existe solução. Caso contrário, executa-se o Passo 4.
- 4. Bounding: Determina-se o nó válido x^l que possua a maior solução. Cria-se um nó imediatamente à direita ou à esquerda de x^l , de forma que se outro nó válido criado a partir de x^{l-1} for selecionado, um novo valor para z^{l-1} pode ser encontrado. Descarta-se o nó x^l e executa-se o Passo 2.

Para exemplificar o uso deste método, considera-se o Problema 3.2 de programação linear inteira mista:

$$max \quad 78x_1 + 77x_2 + 90x_3 + 97y_1 + 31y_2 \tag{3.2}$$

$$11x_{1} + 4x_{2} - 41x_{3} + 44y_{1} + 7y_{2} \leq 82$$

-87x_{1} + 33x_{2} + 24x_{3} + 14y_{1} - 13y_{2} \leq 77
$$61x_{1} + 69x_{2} + 69x_{3} - 57y_{1} + 23y_{2} \leq 87$$

$$x_{1}, x_{2}, x_{3} \geq 0, \quad y_{1}, y_{2} \geq 0,$$

$$x_{1}, x_{2}, x_{3} \quad inteiro$$

Na Figura 3.2 estão representadas as etapas na construção da árvore de enumeração para o Problema 3.2.



Figura 3.2: Arvore Branch and Bound

3.2 Abordagens Heurísticas de Solução de Problemas de Programação Inteira

Muitas vezes, em aplicações reais quando o tamanho do problema (número de variáveis e restrições) é muito grande, a busca pela solução ótima torna-se impraticável. Os métodos clássicos podem demorar muito tempo para chegar a uma solução o que inviabiliza sua utilização prática. Nesses casos, deve-se optar pela busca de uma solução que possa satisfazer parcialmente o objetivo, mas que seja obtida em tempo hábil.

As abordagens heurísticas trazem uma luz para este problema, pois utilizam técnicas de busca que, iterativamente, selecionam as melhores alternativas em um determinado espaço de soluções, sendo que o término da busca é determinado por um critério de parada arbitrário, resultando numa solução. Esta, por sua vez, pode ser ótima ou, na maioria dos casos, uma solução boa o suficiente considerando o problema abordado. As principais abordagens heurísticas são: *Busca Tabu, Simulated Annealing e Algorítmos Genéticos*

3.2.1 Busca Tabu

A Busca Tabu é um procedimento de busca agressivo que avança iterativamente de uma solução para outra através de movimentos na vizinhança do espaço de soluções. A palavra *Tabu* presente no nome desta abordagem refere-se à característica do seu método de busca que permite o acesso a espaços de solução previamente identificados como proibídos ou *tabus* na evidência de que melhores alternativas de soluções possam ser encontradas [14]. Esta característica permite que o algoritmo da Busca Tabu escape de pontos de solução ótimos locais.

Uma versão simplificada do algoritmo iterativo de Busca Tabu está descrita a seguir: *iter* é o número da iteração atual, x_now é a solução da iteração atual, x_best é a melhor solução encontrada até o momento e Tabu(iter) é o conjunto de movimentos tabu na iteração *iter*. Considera-se um movimento como sendo *admissível* se ele pertencer a Tabu(iter) e se sua solução apresentar uma qualidade suficientemente alta, permitindo que seu status tabu seja desconsiderado [15].

- Passo 0: iter = 0; Inicializa-se *x_now*; *x_best = x_now*; $Tabu(iter) = \emptyset$.
- Passo 1: Cria-se uma lista de possíveis movimentos na vizinhança de *x_now*. Avalia-se cada movimento da lista.
- Passo 2: Seleciona-se o movimento com a maior avaliação que não pertença a Tabu(iter), ou que seja um movimento admissível. Efetua-se o movimento e atualiza-se x_now.
- Passo 3: Se *x_now* é melhor que *x_best*, atualiza-se *x_now*.
- Passo 4: Se o critério de parada for satisfeito, termina-se com *x_best*. Senão, *iter = iter + 1*; Atualiza-se *Tabu(iter)*; Executa-se o Passo 1.

3.2.2 Simulated Annealing

A abordagem *Simulated Annealing* ou Recozimento Simulado baseia-se na analogia ao processo de derretimento e posterior resfriamento de materiais até o seu congelamento, levando-os a se transformarem em estruturas cristalinas de mínima energia. Durante este processo de resfriamento algumas medidas devem ser levadas em consideração: o resfriamento deve ser lento e deve-se permanecer um longo período em temperaturas muito próximas ao congelamento. Se isto não for feito, o resultado pode ser um cristal mal formado, ou uma formação não cristalina, ou seja, obtém-se um resultado ótimo local [23]. O algoritmo *Simulated Annealing* efetua uma busca aleatória pelo espaço de solução

procurando por pontos E_i de baixa energia. Nessas movimentações, a probabilidade p de se dar um passo é determinada pela distribuição de Boltzmann,

$$p = \exp\left(-\frac{E_{i+1} - E_i}{kT}\right)$$

se $E_{i+1} > E_i$ ou p = 1 se $E_{i+1} \leq E_i$. A temperatura T é inicializada com um valor alto (fase de derretimento), e uma busca aleatória é efetuada nesta temperatura. A cada temperatura, a busca deverá continuar o tempo necessário até que um equilíbrio seja atingido. Em seguida, a temperatura é reduzida suavemente (fase de resfriamento) de acordo com um fator de resfriamento, por exemplo $T \to \frac{T}{\mu_T}$, sendo μ_T um pouco maior que 1. A busca termina quando a temperatura chegar a zero.

Na aplicação deste algoritmo, a função objetivo do MIP é utilizada como a função de energia, e os valores das variáveis são definidos aleatoriamente respeitando as restrições impostas pelo problema de otimização. A cada temperatura, vários conjuntos de valores são avaliados buscando-se minimizar o resultado da função objetivo. O fato de que alguns conjuntos de valores aceitos pelo algoritmo podem levar a resultados piores, possibilita que o mesmo possa escapar de pontos ótimos locais.

3.2.3 Algorítmos Genéticos

Algoritmos genéticos [19] baseiam-se na teoria da evolução das espécies através da seleção natural. Ao se considerar os cruzamentos de uma espécie, as suas mutações aleatórias juntamente com os desafios impostos pelo meio ambiente, ao longo de várias gerações, esta espécie evolui selecionando, naturalmente, os seus indivíduos mais fortes e mais preparados.

Baseados nos conceitos de evolução, os algoritmos genéticos foram criados como uma heurística de busca que se diferencia da maioria por utilizar uma representação codificada do conjunto de parâmetros do problema (variáveis) ao invés dos próprios parâmetros e por efetuar a busca a partir de uma população de pontos, e não apenas a partir de um ponto [18].

Em algoritmos genéticos, as variáveis do problema são codificadas, em geral na forma binária, em um vetor chamado de cromossomo. Uma população de cromossomos representa várias instâncias definidas aleatoriamente para as variáveis. Cada iteração do algoritmo é chamada de geração. Ao final de cada geração, é feita uma avaliação que calcula a função de *Fitness* para cada cromossomo. Esta função deve ser definida de forma a contemplar as restrições e a função objetivo do MIP. Assim, os cromossomos com melhor *Fitness* são os melhores candidatos a uma solução ótima para o problema. Para efetuar a busca por novas soluções e para garantir a evolução do algoritmo, são definidas três operações básicas sobre os cromossomos: mutação, *crossover* e reprodução. Na mutação, os valores existentes em cada cromossomo podem ser aleatoriamente alterados, no caso de cromossomos binários, alguns valores 0s tornam-se 1s e vice-versa. No *crossover*, partes de um cromossomo são trocadas com o de outro, como se eles fossem embaralhados entre si. Finalmente, na reprodução, os cromossomos com melhores valores de *Fitness* são multiplicados enquanto cromossomos com valores ruins são eliminados.

Basicamente, as etapas de um algoritmo genético são as seguintes :

- Passo 1: Inicializa a população de cromossomos;
- Passo 2: Avalia cada cromossomo da população;
- Passo 3: Se o critério de parada for satisfeito, termina com o cromossomo que possui a melhor avaliação;
- Passo 4: Selecionar alguns elementos da população para formar um nova população;
- Passo 5: Recombinar os elementos da nova população (mutação e *crossover*); Retornar para o Passo 2.

3.3 Otimização Fuzzy

A otimização fuzzy pode ser definida como uma extensão do problema clássico de otimização no qual o objetivo e/ou restrições são considerados flexíveis, aproximados ou incertos. Estes são representados como conjuntos fuzzy que têm como qualidade principal a capacidade de representar as quantidades imprecisas envolvidas no problema a ser modelado.

3.3.1 Conjuntos Fuzzy

Um conjunto é dito fuzzy quando sua definição é dada de forma subjetiva, um exemplo seria a definição do conjunto das pessoas consideradas altas. Outro fator que pode caracterizar os conjuntos fuzzy é a inexistência de fronteiras precisas. No dia-a-dia, a importância da modelagem fuzzy torna-se muito clara, pois raramente existem situações de classificação objetivas, em geral o que se encontra são casos nos quais a subjetividade é predominante.

Na teoria clássica de conjuntos, um determinado elemento é dito como pertencente ou não a um certo conjunto, por exemplo, considerando números inteiros no intervalo [10, 17] tem-se que os números 11, 13 e 17 pertencem ao conjunto P dos números primos, já os números 10, 12, 14, 15 e 16 não. De uma forma mais formal, pode-se definir este conjunto através de uma função característica P(x) que informa a aceitação de x no conjunto dos números primos, sendo 1 e 0 as repostas sim e não respectivamente. Sua definição é dada na Equação 3.3, e a Figura 3.3 ilustra sua função característica.

$$P(x) = \begin{cases} 1 & \text{se } x \text{ \acute{e} divisível apenas por 1 e por } x; \\ 0 & \text{caso contrário.} \end{cases}$$
(3.3)



Figura 3.3: Função característica do conjunto dos números primos no intervalo [10, 17]

Considera-se um outro exemplo no qual se quer definir o conjunto das pessoas jovens [24]. Neste caso fica clara a subjetividade imposta pela definição do conjunto. Considerando casos particulares, observa-se claramente uma indefinição. Uma pessoa de 5 anos é obviamente jovem, uma pessoa de 60 certamente não é jovem. No entanto, como classificar uma pessoa de 30 anos?

Em casos como o descrito acima, define-se o conjunto das pessoas jovens como um conjunto fuzzy \tilde{J} que admite uma certa incerteza de forma que cada pessoa, com idade x, possua um grau de aceitação no conjunto dos jovens expresso pela função de pertinência $\mu_{\tilde{J}(x)}$, que é uma função que mapeia uma entrada x para uma saída pertencente ao intervalo [0, 1]. Quanto mais próximo de 1 for o resultado, mais pertinente será o elemento x ao conjunto representado. Desta forma, pode-se considerar o conjunto das pessoas jovens como sendo modelado pela função de pertinência ilustrada na Figura 3.4.



Figura 3.4: Função de pertinência do conjunto das pessoas jovens

Em um contexto mais simples, as funções de pertinência podem ser utilizadas para representar a mudança entre os graus extremos de pertinência 0 e 1 de forma linear, como pode-se ver nos gráficos da Figura 3.5, e cuja definição analítica geral é dada por:



Figura 3.5: Funções de pertinência linear gerais

3.3.2 Otimização com Objetivo e Restrições Fuzzy

O Problema 3.4, abaixo, que representa o caso geral de otimização com objetivo e restrições fuzzy pode ser descrito como:

$$\widetilde{Minimizar f(x)}$$

$$g_i(x) \widetilde{\geq} b_i, \qquad i = 1, \dots, m$$

$$x_j \ge 0, \qquad j = 1, \dots, n$$

$$(3.4)$$

o que atribui um novo sentido ao objetivo e às restrições. As novas restrições representam a incerteza do sistema modelado a partir de equações fuzzy como as restrições do Problema 3.4 que podem ser entendidas como a representação da expressão " $g_i(x)$ é preferencialmente maior ou igual a b_i ".

Em [43], um método para a resolução do Problema 3.4 baseado no Princípio de Decisão Fuzzy [6] é apresentado. Como no problema considerado o objetivo também é fuzzy, devese escolher um nível de aspiração máxima b_0 para o mesmo, e tranformá-lo na restrição fuzzy $f(x) \leq b_0$ (problema de minimização).

Os níveis de factibilidade do objetivo e das restrições podem ser expressos respectivamente através das funções de pertinência $\mu_{f(x)} \in \mu_{g_i(x)}$, no qual d_i , i = 0, ..., m representa o grau de violação de cada restrição.

$$\mu_{f(x)} = \begin{cases} 1 & \text{se} \quad f(x) \le b_0; \\ \frac{d_0 - f(x) + b_0}{d_0} & \text{se} \quad b_0 < f(x) \le b_0 + d_0; \\ 0 & \text{se} \quad f(x) > b_0 + d_0. \end{cases}$$
(3.5)

$$\mu_{g_i(x)} = \begin{cases} 1 & \text{se } g_i(x) \ge b_i; \\ \frac{g_i(x) - b_i + d_i}{d_i} & \text{se } b_i - d_i \le g_i(x) < b_i; \\ 0 & \text{se } g_i(x) < b_i - d_i. \end{cases}$$
(3.6)

Princípio de Bellman e Zadeh para decisão fuzzy. Considere um conjunto X de infinitas alternativas a serem avaliadas por k diferentes objetivos. Seja $C_j(x) \in [0, 1]$, $j = 1, \ldots, k$ uma função de pertinência que indica o nível de satisfação do j-ésimo objetivo pelo ponto x, então o nível D(x) de satisfação de todos os objetivos pelo ponto x é dado por:

$$D(x) = Min\{C_1(x), C_2(x), \dots, C_k(x)\},$$
(3.7)

de forma que a melhor alternativa x^* é determinada por:

$$D(x^*) = Max_{x \in X} D(x) \tag{3.8}$$

Pode-se transformar o Problema 3.4 em:

$$MaxD(x) = Min\{\mu_{f(x)}, \mu_{g_1(x)}, \mu_{g_2(x)}, \dots, \mu_{g_m(x)}\}$$
(3.9)

$$x \ge 0$$

obtém-se então o seguinte problema de maximização do nível de satisfação:

$$Maximizar \qquad \lambda \tag{3.10}$$

$$1 - \frac{f(x) - b_0}{d_0} \ge \lambda$$

$$1 + \frac{g_i(x) - b_i}{d_i} \ge \lambda, \qquad i = 1, \dots, m$$

$$x \ge 0$$

$$\lambda \in [0, 1]$$

do qual pode-se obter a solução x e o nível de satisfação λ .
Capítulo 4

Estimadores de Banda Passante Equivalente

No cenário das redes multiserviço de alta-velocidade há uma grande variedade de aplicações que demandam requisitos de Qualidade de Serviço distintos. Para que se possa garantir QoS às aplicações multimídia, é necessário estimar a quantidade de banda passante requerida por estas, denominada de Banda Passante Equivalente ou Efetiva (EB).

Basicamente, há duas formas de se estimar a Banda Passante Equivalente, com estimadores analíticos ou baseados em medições. Os estimadores analíticos são derivados a partir de uma caracterização completa do processo de chegada, e requerem parâmetros que nem sempre são observados em um fluxo finito, como o valor de pico do processo. Os estimadores de banda passante baseados em medições produzem avaliações da demanda de banda passante utilizando apenas amostras do tráfego, o que os torna a ferramenta ideal para o ajuste dinâmico da banda passante alocada para as classes de serviço em redes multiserviço.

Neste capítulo, compara-se o estimador Gaussian Approximation Allocation (GA) e o estimador Fractal Envelope Process (FEP) que é uma variação do Norros EB Allocation (NEB) derivado a partir de hipóteses realistas. Optou-se por comparar estes dois estimadores pois eles representam um compromisso (trade-off) entre eficácia e eficiência, sendo o estimador GA mais simples e consequentemente mais eficiente e o estimador FEP mais eficaz, dado que sua expressão considera o tamanho do buffer e as dependências de longa duração.

4.1 Estimadores de Banda Passante Equivalente

Nesta seção, define-se o conceito de banda passante equivalente e são apresentados vários estimadores de banda existentes na literatura.

4.1.1 Definição de Banda Passante Equivalente

Em [11] a banda passante equivalente associada a uma fonte é definida da seguinte forma: considere X[0,t] como a quantidade de trabalho que chega de uma fonte no intervalo [0,t]. Assumindo que X[0,t] tem incrementos estacionários, a banda passante equivalente é definida como:

$$\alpha(s,t) = \frac{\log E(e^{sX[0,t]})}{st} \qquad 0 < s, t < \infty \tag{4.1}$$

sendo s e t chamados de parâmetro de espaço e tempo e E é a esperança do processo. Algumas das propriedades mais interessantes dessa definição são:

(i) Se $X[0,t] = \sum_i X_i[0,t]$, e as fontes $X_i[0,t]$ são independentes, então

$$\alpha(s,t) = \sum_{i} \alpha_i(s,t) \tag{4.2}$$

 (ii) Para qualquer valor fixo de t, α(s, t) é crescente em s, e permanece entre o valor médio e o valor de pico da taxa de chegada medida sobre um intervalo de comprimento t, isto é

$$\frac{EX[0,t]}{t} \le \alpha(s,t) \le \frac{\bar{X}[0,t]}{t} \tag{4.3}$$

sendo $\overline{X}[0,t]$ é o valor supremo encontrado no processo. A forma de $\alpha(s,t)$ próximo de s = 0 é determinada pela média, variância e os momentos superiores de X[0,t], enquanto que a forma de $\alpha(s,t)$ próximo de $s = \infty$ é principalmente influenciada pela distribuição de X[0,t] próximo ao seu máximo.

4.1.2 Uma comparação entre Estimadores de Banda Passante Equivalente

Considerando o cenário das redes multiserviço auto-ajustáveis, um estimador de banda passante equivalente deve ser robusto para poder lidar com as características heterogêneas do tráfego, ou seja, ser capaz de estimar a EB na presença de fluxos formados por aplicações de diversos tipos. O estimador de EB deve possuir também uma implementação simples o suficiente para que possa ser executado em um espaço de tempo curto, e assim possibilite a obtenção de estimativas em tempo-real.

Em [16] vários estimadores de EB foram avaliados quanto à sua complexidade de implementação e sua eficácia:

Direct EB Allocation (DEB) : é um estimador baseado na equação (4.1), na qual t é o parâmetro de tempo e está relacionado com as escalas de tempo reponsáveis pelo transbordo do *buffer*, e s é parâmetro de espaço e é calculado a partir de uma

hipótese assintótica de decaimento exponencial para a probabilidade de transbordo do *buffer*:

$$P(Q > B) = e^{-s(C)B} \tag{4.4}$$

Courcoubetis EB Allocation (CEB) : utiliza uma definição de banda passante equivalente alternativa que é mais simples:

$$eb = m + \frac{IDs}{2B} \tag{4.5}$$

nos quais m, B, $s \in ID$ são a taxa média, tamanho do *buffer*, parâmetro de espaço e índice de dispersão de X[0,t]. O parâmetro de espaço é calculado utilizando a Equação 4.4;

- Many Sources Asymptotic EB Allocation (MSAEB) : também é baseado na equação (4.1), mas requer a solução de um problema de otimização na estimação dos valores de s e t;
- **ON-OFF EB Allocation (OOEB)** : é baseado na equação (4.1), mas utiliza um processo ON-OFF como processo de chegada ao invés de X[0, t];
- **Norros EB Allocation (NEB)** : é um estimador para o Movimento Browniano Fractal que considera o parâmetro de *Hurst*:

$$eb = m + \left(H^{H} * (1 - H)^{1 - H} * \sqrt{-2 * \ln(P_{loss})}\right)^{\frac{1}{H}} * a^{\frac{1}{2H}} * K^{\frac{H-1}{H}} * m^{\frac{1}{2H}}$$
(4.6)

nos quais a, m, P_{loss}, K , e H são o coeficiente de variação, a taxa média, a probabilidade de perda de pacotes, o tamanho do *buffer* e o parâmetro de *Hurst*;

DRDMW (Improved Empirical EB Allocation) : é baseado em métodos empíricos de estimação de banda passante, e considera o parâmetro de *Hurst* na sua equação da probabilidade de transbordo:

$$P(Q > B) = e^{-s(C)B^{2-2H}}$$
(4.7)

Gaussian Approximation Allocation (GA) : é um método baseado na distribuição de chegada Gaussiana:

$$EB = m + \sigma * \sqrt{-2 * \ln(P_{loss}) - \ln(2 * \pi)}$$
(4.8)

nos quais $m \in \sigma$ são a taxa média e o desvio padrão da distribuição de chegada.

Estimador/Cenário	MSAEB	DEB	OOEB	CEB	NEB	DRDMW	GA
Buffer Pequeno	Muito	Ruim	Ruim	Ruim	Bom	Ruim	Ótimo
	Bom						
Buffer Grande	Muito	Muito	Muito	Muito	Muito	Muito	Ruim
	Bom	Bom	Bom	Bom	Bom	Bom	
Tráfego SRD	Bom	Muito	Bom	Muito	Bom	Muito	Bom
		Bom		Bom		Bom	
Tráfego LRD	Bom	Ruim	Ruim	Ruim	Ótimo	Ruim	Ruim
Várias Fontes	Ótimo	Muito	Ruim	Muito	Muito	Muito	Ótimo
		Bom		Bom	Bom	Bom	
Uma Fonte	Ruim	Muito	Bom	Muito	Bom	Muito	Ruim
		Bom		Bom		Bom	
Complexidade	Ruim	Ruim	Ruim	Muito	Bom	Bom	Ótimo
				Bom			
Requisitos de Memória	Bom	Bom	Ótimo	Bom	Bom	Ótimo	Ótimo

Os resultados apresentados em [16], sobre o desempenho destes estimadores em cenários de rede, encontram-se na Tabela 4.1.

Tabela 4.1: Comparação de Desempenho dos Estimadores

Dentre os estimadores citados, apenas o NEB e GA não requerem a estimativa dos parâmetros de espaço e tempo (s, t), pois são expressões aproximadas que foram derivadas independentemente da fórmula de banda passante equivalente. Juntamente com os estimadores CEB e DRDMW, são os únicos que possuem um baixo nível de complexidade. NEB e DRDMW, são os únicos estimadores que utilizam o parâmetro de Hurst, levando em consideração o tráfego com dependência de longa duração (LRD).

O estimador NEB também apresentou bons resultados nos testes com buffers grandes e com várias fontes de tráfego, e foi o único com uma classificação ótima frente ao tráfego LRD. O estimador *GA* é o mais simples existente na literatura, e devido ao fato de não considerar o tamanho do buffer em seu cálculo tende a ser um limitante superior em suas estimativas de banda. Esta característica torna-o uma opção interessante para efetuar uma avaliação comparativa. Finalmente, tendo em vista os resultados apresentados em [16] e nossas aspirações quanto às qualidades de um estimador de EB, o estimador NEB é o mais indicado a ser utilizado no ambiente das redes auto-ajustáveis.

Analisando mais detalhadamente a Equação 4.6 do NEB, aproxima-se o coeficiente de variação a pelo índice de dispersão (variância sobre a média) no caso de tráfego com dependência de curta duração (SRD), já no caso de tráfego com dependência de longa

duração (LRD), a é aproximado pela expressão:

$$a = r^{2H-1} x_0^{-2H} \left(2x_0 E(U \wedge x_0) - E(U \wedge x_0)^2 \right)$$
(4.9)

sendo r a taxa de transmissão de rajadas, x_0 é uma constante e U uma variável aleatória relacionada à distribuição do tamanho das rajadas, e cujo valor é limitado por x_0 e por $(U \wedge x_0)$, e \wedge denota a conjunção entre U e x_0 .

Há também um framework baseado no Processo Envelope do Movimento Browniano Fractal (FEP) [8] que obteve o mesmo resultado alcançado por Norros [31], porém sem que fosse necessário fazer suposições restritivas, referentes ao tamanho do buffer (infinito) necessário para a aplicação da Teoria dos Grandes Desvios, sendo, portanto, uma opção mais interessante do que o NEB.

Segundo [8], o processo envelope $\hat{A}_H(t)$ para o movimento Browniano Fractal $A_H(t)$ com média \bar{a} e desvio padrão σ é dado por

$$\hat{A}_H(t) = \bar{a}t + k\sigma t^H \tag{4.10}$$

sendo $k = \sqrt{-2 \ln \varepsilon}$ a probabilidade de que $A_H(t)$ exceda $\hat{A}_H(t)$ no tempo t. Considerando uma fila com uma taxa de serviço c alimentada por um processo envelope $\hat{A}_H(t)$, então a quantidade de células na fila $\hat{q}(t)$ é

$$\hat{q}(t) = \hat{A}_H(t) - ct \ge 0$$
(4.11)

Definindo a escala de tempo t^* , como sendo a escala na qual occorre a máxima utilização da fila, tem-se que $q_{max} = max(\hat{q}(t)) = \hat{q}(t^*)$. Para calcular q_{max} , deve-se encontrar t^* de forma que

$$\frac{d\hat{q}(t^*)}{dt} = 0$$
(4.12)

ou equivalentemente,

$$\frac{d\hat{A}_H(t^*)}{dt} = c \tag{4.13}$$

Substituindo t^* na Equação 4.11 pode-se concluir que:

$$q_{max} = \hat{A}_H(t^*) - ct^* \tag{4.14}$$

Resolvendo a Equação 4.14 e isolando a taxa de serviço da fila, pode-se obter a equação do estimador FEP para o cálculo da EB:

$$\hat{c} = \bar{a} + K^{\frac{H-1}{H}} * (\sqrt{-2 * \ln(P_{loss})} * \sigma)^{\frac{1}{H}} * H(1-H)^{\frac{1-H}{H}}$$
(4.15)

onde:

 \hat{c} : banda passante equivalente;

 \bar{a} : taxa média;

 σ : desvio padrão;

 P_{loss} : probabilidade de transbordo do *buffer*;

K: tamanho do buffer;

H : parâmetro de *Hurst*.

Como o estimador FEP utiliza o parâmetro de *Hurst*, sendo $H \in [0, 5; 1)$, pode-se calcular a Equação 4.15 para o tráfego SRD (H = 0, 5) como:

$$\hat{c} = \bar{a} + \frac{0.5 \left(\sigma * \sqrt{-2 * \ln(P_{loss})}\right)^2}{K}$$

$$(4.16)$$

e para o tráfego LRD extremo $(H \rightarrow 1)$ como:

$$\hat{c} \to \bar{a} + \sigma * \sqrt{-2 * \ln(P_{loss})}$$

$$(4.17)$$

Fica claro na Equação 4.16 que para o tráfego SRD o valor estimado pelo FEP depende fortemente do tamanho do *buffer* utilizado, o que é esperado, tendo em vista que tratase de um processo de chegada com ausência de auto-similaridade, e que portanto não impacta de forma significativa na taxa de utilização do *buffer* [35].

Por outro lado, a Equação 4.17 trata o pior caso do tráfego LRD, no qual há autosimilaridade, implicando em um fluxo com grande variabilidade no tamanho das rajadas, levando a ineficiência do *buffer* [35]. Na Equação 4.17, o tamanho do *buffer* não é considerado, e sua forma reduz-se praticamente a Equação 4.8, do estimador GA. Portanto, é de se esperar que para casos em que o parâmetro de *Hurst* assuma valores muito altos, a estimativa do FEP aproxime-se da estimativa feita pelo GA.

4.2 Uma Comparação entre os Estimadores FEP e GA

O objetivo dos experimentos de simulação é avaliar os estimadores FEP e GA quanto à sua capacidade de alocar a banda passante suficiente para a provisão de *QoS*.



Figura 4.1: Cenário da simulação

Na Figura 4.1 é apresentado o cenário dos experimentos de simulação que incluem o tráfego a ser modelado (a), *buffer* (b), taxa de serviço (f), o método de coleta de dados do tráfego (c) e (d) e o estimador de EB (e). O objetivo é definir a taxa de serviço (banda passante equivalente) de modo a prover os requisitos de QoS desejados. Para tal, o método de coleta é fundamental de forma a se poder alimentar os estimadores com informações confiáveis.

Para se estimar os parâmetros do tráfego de chegada é necessário definir a fatia de tempo ou *timeslot* (c) em que será agregado o tráfego. Deve-se definir um intervalo de tempo no qual é feita a contagem do número de células. Cada *timeslot* tem, então, um valor representando uma quantidade específica de tráfego (d), ou seja, o número de chegadas de pacotes, por unidade de tempo. Uma janela de observação, contendo um ou mais *timeslots*, é utilizada para o cálculo da taxa média e desvio padrão do tráfego. Estes parâmetros são utilizados pelos estimadores de EB (e) para o cálculo da banda passante. O valor final calculado é a banda passante equivalente, que é utilizada para o dimensionamento da taxa de serviço da fila (f).

Em [16], a escolha correta dos valores do timeslot e do tamanho da janela de observação é descrita como passo essencial para a obtenção de um bom resultado para os estimadores de banda passante. É necessário, também, definir os requisitos de QoS para o cálculo do estimador GA, além do tamanho da fila utilizada e o parâmetro de Hurst para o cálculo do estimador FEP. Estas definições são discutidas a seguir.

4.2.1 Definição dos Parâmetros para os Experimentos de Simulação

Em [16] os estimadores de banda foram testados utilizando-se uma fonte sintética baseada em um processo fractal renovável com uma taxa média de 20 Kbytes/s e um tamanho médio do pacote de 200 bytes, com isso pretendia-se simular a passagem de pacotes em uma rede *Ethernet*. O parâmetro de *Hurst* foi fixado em 0, 7, o tamanho do buffer em 5 Kbytes e a probabilidade de perda de pacotes (QoS) em 10^{-3} .

Neste trabalho, redes ponto-a-ponto de alta velocidade, presentes em *backbones* e na *Internet*, são consideradas. Nos *backbones* da *Internet* baseados na tecnologia ATM, enlaces óticos OC3 são amplamente utilizados. Nessas redes, é observado uma grande variabilidade de taxas de transmissão, grandes *buffers* e requisitos de QoS justos.

O tamanho médio do *buffer* utilizado nos experimentos foi de 8000 células, definido baseado em informações disponíveis no sítio da *Cisco* na *Internet* (*www.cisco.com*), que é um dos maiores fornecedores de roteadores atualmente [37]. Em sua página, as características de roteadores comerciais são descritas. O requisito de *QoS* foi definido como a taxa de descarte de pacotes e seu valor foi fixado em 10^{-6} .

O tráfego utilizado nos experimentos foi segregado por classe (voz, vídeo e dados) e as taxas de transmissão foram definidas como em [17]. Os parâmetros do sistema utilizados nos experimentos estão descritos na Tabela 4.2.

	Buffer (células)	QoS	Voz (Kbps)	Vídeo (Kbps)	Dados (Kbps)
Máximo	80.000	10^{-3}	23.000	128.000	20.000
Padrão	8.000	10^{-6}	-	-	-
Mínimo	2.000	10^{-9}	5.000	55.000	1.500

Tabela 4.2: Variação dos parâmetros do sistema - Buffer (células), Taxas (Kbps)

Considerando os parâmetros padrão para os experimentos de simulação, que são: buffer de 8000 células (424000 bytes) e probabilidade de transbordo do buffer de 10^{-6} , podese calcular o ponto de intersecção das curvas dos estimadores FEP e GA em função do parâmetro de Hurst. Obtêm-se, assim, os valores para o desvio padrão σ a partir dos quais a estimativa do FEP supera a do GA, i.e., para um determinado valor de H, se o valor do desvio padrão do tráfego for menor que o valor indicado na Tabela 4.3, a estimativa do FEP será menor que a estimativa do GA. Estes resultados são úteis para analisar cenários nos quais há grande variabilidade no tráfego.

Н	$0,\!5$	$0,\!55$	$0,\!6$	$0,\!65$	0,7	0,75	$0,\!8$	0,85	0,9	0,95
σ (Kbps)	2494	2855	3296	3849	4562	5518	6864	8891	12219	17785

Tabela 4.3: Ponto de intersecção das curvas dos estimadores em função de H

4.2.2 Estimador do Parâmetro de Hurst

Para lidar com as características dinâmicas do tráfego, utiliza-se um estimador para o parâmetro de Hurst capaz de efetuar medidas em tempo-real. O estimador escolhido foi o Abry-Veitch, ou simplesmente AV [1], dada sua baixa complexidade de implementação e boa precisão. Em [26], discute-se o problema da estimação de H para o tráfego no qual ocorrem efeitos não-estacionários, como mudanças de nível ou tendências lineares ou polinomiais. Nestes casos, técnicas tradicionais de estimação costumam falhar. Por outro lado, o estimador AV é recomendado como uma opção robusta, mesmo na presença de efeitos não-estacionários.

O estimador AV [38] é um estimador semi-paramétrico dos parâmetros do domínio do espectro da dependência de longa duração baseado nas *Transformadas Wavelet Discretas*. As transformadas *wavelet*, em geral, podem ser entendidas como uma forma mais flexível das transformadas de Fourier, sendo que Y(t) é transformado no domínio das escalas de tempo, e não no domínio da freqüência. Pode-se dizer que as tranformadas *wavelet* são um método para se observar uma série temporal simultâneamente em um vasto número de escalas *a*. As escalas, que podem ser pensadas como o inverso da freqüência, são medidas em unidades logarítmicas *j*, sendo $a = 2^j$.

Para traços finitos de dados com comprimento n, j = 1 a $j_{max} \approx \log_2(n)$. Para medir as propriedades de segunda ordem, um conjunto de estatísticas μ_j é calculado para cada ja partir de coeficientes discretos das transformadas wavelet. O resultado pode ser visto em um gráfico log-log chamado de *Logscale Diagram*, que é um tipo de estimativa espectral.



Figura 4.2: Logscale Diagram para um tráfego LRD com o parâmetro H = 0.8

Na Figura 4.2 pode-se ver um exemplo de Logscale Diagram gerado a partir de um traço sintético de dados no qual ocorre dependência de longa duração, com H = 0, 8. A partir do gráfico, deve-se definir o intervalo $j \in [j_1, j_2]$ onde um alinhamento do gráfico é observado. Para definir o parâmetro de Hurst, o estimador efetua uma regressão linear no intervalo já definido. No caso da Figura 4.2, o alinhamento é observado no intervalo $j \in [5, 11]$, e está explicitado por uma linha tracejada.

A tarefa de definir o intervalo de alinhamento no *Logscale Diagram* é em certo grau subjetiva, o que requer a inspeção visual para sua realização de forma correta. Porém, para utilizar o estimador AV em um ambiente *on-line* é necessário automatizar este procedimento. Para solucionar este problema, foi implementada a heurística proposta em [39], que define o intervalo de alinhamento de forma a minimizar o erro na estimação do parâmetro de *Hurst*.

4.2.3 Complexidade dos Estimadores

Ambos os estimadores utilizam expressões matemáticas simples, que podem ser calculadas em tempo constante, logo possuem a complexidade de tempo O(1). A expressão do estimador *FEP* é mais complexa do que a do *GA*, e requer um tempo quatro vezes maior para ser computada, porém, como ambas as expressões são calculadas em poucos milésimos de segundos, esta diferença não é significativa na prática.

Todavia, em um cenário dinâmico, no qual as estimativas são realizadas freqüentemente a partir de amostras do tráfego corrente, os valores para taxa média (m), desvio padrão (σ) e parâmetro de *Hurst* (H) devem ser recalculados toda vez que o estimador for utilizado. Logo, deve-se considerar a complexidade de tempo do cálculo dessas estatísticas na complexidade final para cada estimador.

Assumindo que a amostra de tráfego considerada possui tamanho n, é sabido que tanto o cálculo da média quanto o cálculo do desvio padrão são realizados com complexidade de tempo O(n). No caso do estimador *FEP*, há também a necessidade de calcular o parâmetro de *Hurst*, que requer um estimador próprio. Neste trabalho, foi utilizado o estimador AV [1] que também possui a complexidade de tempo O(n).

Portanto, pode-se definir a complexidade de tempo do estimador GA como sendo o resultado da soma das complexidades de tempo do cálculo da média, do desvio padrão e da expressão do estimador, tem-se, então: O(n) + O(n) + O(1) = 2.O(n) = O(n). No caso do estimador *FEP* adiciona-se mais um fator O(n) para o cálculo do parâmetro de *Hurst*, ou seja: O(n) + O(n) + O(n) + O(1) = 3.O(n) = O(n).

Ambos os estimadores possuem a complexidade O(n), o que garante um crescimento linear no tempo de execução com o aumento da amostra de tráfego considerada. Na Tabela 4.4 são apresentados alguns resultados de tempo de execução para tamanhos arbitrários de n (Pentium III 800MHz). Sendo $n \leq 100.000$ nos experimentos realizados neste artigo.

n	GA	FEP
10.000	$30 \mathrm{ms}$	$170 \mathrm{~ms}$
20.000	$57 \mathrm{ms}$	260 ms
50.000	140 ms	$540 \mathrm{~ms}$
100.000	266 ms	$998 \mathrm{ms}$

Tabela 4.4: Tempo de CPU requerido pelos estimadores $FEP \in GA$ para entradas de diversos tamanhos.

É importante ressaltar que a complexidade do estimador FEP é de O(n) somada a complexidade do estimador do parâmetro de Hurst, portanto, a escolha de um estimador de H rápido é de grande importância, principalmente nos cenário no qual as estimativas devem ocorrer em curtos espaços de tempo.

4.2.4 Procedimentos Utilizados Para a Validação dos Estimadores de Banda Passante Equivalente

Para efetuar a validação dos estimadores, os mesmos foram submetidos a diferentes cenários no qual foram avaliadas as seguintes características: definição da escala de tempo de interesse, que é o *timeslot* de agregação do tráfego, i.e., a unidade de tempo utilizada na coleta do tráfego, a partir da qual pode-se obter as estatísticas do fluxo, como a taxa

média e o desvio padrão. Variando seu valor, pode-se identificar *timeslots* nos quais o comportamento dos estimadores é bom, i.e., quando não ocorre uma superalocação de banda passante. A intenção com estes testes não é tentar encontrar o *timeslot* ótimo, i.e., a menor alocação de banda passante equivalente que garanta os requisitos de *QoS*. Pretende-se evitar os intervalos de tempo nos quais se observa um desempenho indesejável, oquando a banda passante é superalocada. Como o cenário no qual as avaliações ocorrerem é dinâmico, requerendo estimativas de banda em intervalos de tempo curtos, o ideal é escolher o menor *timeslot* possível que satisfaça condições mínimas de qualidade para os dois estimadores escolhidos. Assim, obtém-se um maior número de pontos a serem analisados por intervalo entre estimativas.

Além da análise da escala de tempo de interesse, foram realizadas também avaliações com janelas de observação de tamanhos diferentes, de forma a se poder observar a robustez dos estimadores. Uma janela de observação define a quantidade de tempo considerada para o cálculo da EB, ou seja, a quantidade de *timeslots*.

Em seguida, avaliou-se a EB alocada pelos dois estimadores ao longo do tempo em comparação à taxa média do tráfego. Assim, pode-se ter uma boa idéia do comportamento dos estimadores e da distância que eles mantêm entre suas estimativas. Como parâmetros neste teste, utilizou-se o *timeslot* de agregação definido anteriormente, e também os parâmetros padrões da Tabela 4.2 para o sistema, ou seja um buffer de 8000 células e um requisito de QoS de 10^{-6} . Em outra análise, foram considerados os parâmetros máximos e mínimos da Tabela 4.2, onpode-se observar como cada estimador reagiu individualmente e como a relação entre suas estimativas foi afetada.

Finalmente, os estimadores foram submetidos a um tráfego gerado por um número variável de fontes, sendo possível observar a capacidade de reação às mudanças de carga repentinas.

4.3 Resultados Numéricos

Nesta seção, os resultados de quatro experimentos envolvendo tráfego de voz, vídeo e dados são apresentados. Em cada experimento foram considerados três valores para as taxas do tráfego gerado, referentes aos valores definidos na Tabela 4.2. O tempo total de simulação foi de 100 segundos para cada cenário. O tráfego foi agregado utilizando *timeslots* com diferentes durações: 3, 2, 1, 0.1, 0.01 e 0.001 segundos e o tamanho da janela de observação variou em 3, 6, 10 e 30 *timeslots*.

O simulador NS-2 [34] foi utilizado para efetuar as simulações. Dois nós foram definidos com um enlace unidirecional de 155Mbps e 1ms de retardo. O tráfego foi transmitido utilizando o protocolo UDP (Figura 4.3).

Os resultados foram organizados, para cada tipo de tráfego, em quatro conjuntos de



Figura 4.3: Configuração básica do NS

gráficos. O primeiro conjunto apresenta a banda passante equivalente estimada por FEP e GA, sendo que, cada um dos gráficos possui quatro curvas para diferentes tamanhos da janela de observação, i.e., o número de pontos utilizados no cálculo do estimador. O eixo horizontal é apresentado em escala logarítmica para possibilitar a visualização dos resultados para *timeslots* menores do que 1 segundo.

O segundo conjunto de gráficos apresenta a variação da EB estimada por ambos os estimadores, juntamente com a taxa média do tráfego em função do tempo. O terceiro conjunto é semelhante ao anterior, porém, utiliza apenas o cenário padrão da Tabela 4.2 para as taxas do tráfego. Avalia-se também os casos nos quais o *buffer* é máximo e o requisito de QoS é baixo, e no qual o *buffer* é mínimo e o requisito de QoS é alto, utilizando os parâmetros definidos na Tabela 4.2. O quarto gráfico ilustra um cenário dinâmico no qual o número de fontes ativas sofre mudanças repentinas.

É importante ressaltar que os requisitos de QoS foram satisfeitos integralmente pelos dois estimadores (FEP e GA), restando portanto avaliar a sua eficiência nos experimentos a seguir.

4.3.1 Experimentos de Simulação com Tráfego Sintético de Voz

Três cenários de tráfego foram obtido através do agregado de 100, 400 e 700 fontes *Expo*nential ON/OFF homogêneas. A configuração das fontes *Exponential* segue os parâmetros propostos por [33] para uma fonte de voz:

Período ON: 800 ms;

Período OFF: 1200 ms;

Tamanho do pacote: 53 bytes (1 célula ATM);

Taxa de pico: 72 kbps (170 células/segundo).

Avaliação da escala de tempo de interesse

A Figura 4.4 apresenta as estimativas geradas pelo estimador FEP (coluna da esquerda) e pelo estimador GA (coluna da direira), para o tráfego agregado de 100, 400 e 700 fontes. Como se pode observar nos gráficos, a variação no número de fontes exponenciais não afetou o comportamento das curvas de estimação de banda passante. Tanto para o estimador FEP como para o GA o valor para qual ocorreu a menor estimativa foi 10ms. Para *timeslots* menores os valores estimados pelo GA apresentaram um aumento exagerado em relação aos valores estimados pelo FEP que foram pouco maiores. Para *timeslots* maiores o resultado foi o oposto, os valores estimados pelo FEP tiveram um crescimento exagerado, enquanto as estimativas do GA apresentaram um crescimento mais conservador.

Quanto a variação das estimativas frente a janela de análise, todos os gráficos apresentaram uma boa convergência com *timeslots* menores ou iguais a 10ms. Em seguida essa convergência piorou e, para *timeslots* maiores do que 1s, é observado novamente uma tendência à convergência das curvas.

Dado os resultados desses testes, decidiu-se utilizar um *timeslot* de agregação de 10ms como padrão para análise do tráfego de voz, pois neste valor foram observados as menores estimativas com um bom grau de robustez em relação a variação da janela de análise.



Figura 4.4: Banda Efetiva vs Timeslot para os estimadores FEP e GA, para o tráfego agregado de 100, 400 e 700 fontes sintéticas de voz.

Comparação dos Estimadores

Os gráficos da Figura 4.5 da variação das estimativas demonstram uma clara superioridade do estimador FEP em relação ao GA. Dado que o tráfego com fraca dependência de longa duração (0, 6 < H < 0, 7) possui uma intensidade de rajadas baixa, uma estimativa de banda passante equivalente próxima ao da taxa média é suficiente para garantir a *QoS* desejada. Tal fato não ocorreu com as estimativas do GA, que se comportou como um limitante superior largo. Caso este estimador seja utilizado, haverá o desperdício de banda passante. Como exemplo, considerando o gráfico acima a direita (400 fontes) vê-se que a taxa média permaneceu em aproximadamente 11,5Mbps enquanto que as estimativas do



GA foram de 14,5Mbps, 21% acima da média.

Figura 4.5: Taxa Média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 100, 400 e 700 fontes sintéticas de voz.

Impacto da Variação dos Parâmetros nos Estimadores

No teste dos cenários extremos, foi considerado apenas o tráfego agregado de 400 fontes. Pode-se ver no gráfico à esquerda da Figura 4.6, no qual se considera um buffer muito grande e uma *QoS* baixa, que as estimativas do estimador FEP praticamente coincidiram com a taxa média. Este resultado é previsível caso se considere a Equação 4.16 do estimador FEP, no qual o tamanho do buffer exerce um grande impacto na estimativa. Neste cenário, a estimativa do GA foi de aproximadamente 13,5Mbps. Já no gráfico à direita da Figura 4.6 quando o cenário é o de pior caso, os valores estimados pelo FEP tornaram-se um limitante superior justo. Neste caso a estimativa do GA foi de 15,5Mbps. Logo, em ambos os cenários, o estimador GA superestimou a demanda de banda passante.



Figura 4.6: Taxa média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 400 fontes sintéticas de voz, para um *Buffer* de 80000 e 2000 células, e requisitos de QoS de 10^{-3} e 10^{-9} .

Comportamento dos Estimadores com Tráfego Dinâmico

Na Tabela 4.5 está descrito o cenário no qual há uma variação do número de fontes exponenciais agregadas no tempo. Na Figura 4.7, pode-se observar o comportamento dos estimadores. Como já observado antes os valores estimados pelo FEP comportaramse como um limitante superior justo em relação a taxa média. Por outro lado, o GA comporta-se como um limitante superior largo. Observa-se, também, que o estimador GA reage de forma mais brusca às mudanças na taxa média de transmissão, devido a sua maior sensibilidade a variabilidade do tráfego. Neste ponto, o parâmetro $H \approx 0,6$ faz com que a Equação 4.15 do estimador FEP aproxime-se de sua Equação 4.16 para tráfego SRD. A divisão pelo tamanho do *buffer* impede que mudanças abruptas na variância do tráfego reflitam diretamente em sua estimativa.

Tempo	0	20	40	60	80
Fontes	100	400	700	100	700

Tabela 4.5: Variação do número de fontes ativas de voz.



Figura 4.7: Taxa Média e banda passante estimada por FEP e GA em função do tempo, na presença de variação do número de fontes sintéticas de voz.

4.3.2 Experimentos de Simulação com Tráfego de Vídeo

Os três cenários de tráfego simulados foram obtidos através do agregado de 60, 180 e 360 fontes *MPEG4 e H263*. Foram utilizados traces reais dos filmes *Jurassic Park, Silence of the Lambs e Star Wars IV* obtidos no sítio do *Telecommunication Networks Group* [27].

Avaliação da escala de tempo de interesse

Pode-se ver na Figura 4.8, que a variação no número de fontes de vídeo não afetou o comportamento das curvas de estimação de banda. Tanto para as estimativas do estimador FEP como para as do GA o *timeslot* em que ocorreu a menor estimativa foi 1 segundo. Para *timeslots* menores ambos os estimadores apresentaram um aumento exagerado. Para *timeslots* maiores o resultado indicou um crescimento suave para ambos os estimadores.

Quanto a variação das estimativas frente a janela de análise, todos os gráficos apresentaram uma boa convergência com *timeslots* maiores ou iguais a 10ms. Para 1ms houve uma clara divergência entre os resultados, mostrando que se trata de uma escala de tempo em que há uma grande variabilidade.

Dado os resultados desses testes decidiu-se utilizar um *timeslot* de agregação de 1 segundo como padrão para análise do tráfego de vídeo pois se obteve um bom grau de robustez em relação a variação da janela de análise.



Figura 4.8: Banda Efetiva vs Timeslot para os estimadores FEP e GA, para o tráfego agregado de 60, 180 e 360 traços de video.

Comparação dos Estimadores

Os gráficos da Figuras 4.9, da variação das estimativas demonstram uma similaridade muito grande entre os valores estimados pelo FEP em relação aos do GA. Neste caso, o tráfego possui forte dependência de longa duração (0, 85 < H < 1) possuindo uma alta intensidade das rajadas, levando o estimador FEP a alocar bastante banda, o que foi previsto pela Equação 4.17 do estimador FEP para tráfego LRD extremo. Com o tráfego agregado de vídeo ambos os estimadores obtiveram desempenho semelhante, não havendo vantagens no uso do estimador FEP.



Figura 4.9: Taxa Média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 60, 180 e 360 traços de video.

Impactos da Variação dos Parâmetros nos Estimadores

No teste dos cenários extremos foi considerado apenas o tráfego agregado de 180 traços. Novamente, ambos os estimadores apresentaram estimativas similares. Podemos ver nos resultados que o que mudou foi apenas a distância entre as estimativas e a taxa média, sendo esta menor (77Mbps) no cenário observado no gráfico à esquerda da Figura 4.10 e maior (80Mbps) no caso observado no gráfico à direita da Figura 4.10. Este é um resultado muito interessante dado que a Equação 4.8 do estimador GA não considera o tamanho do *buffer* utilizado. Isto demonstra o cenário no qual há a presença de um fluxo com um valor de H alto, modelado pela Equação 4.17 do estimador FEP, que também não considera o *buffer*.



Figura 4.10: Taxa média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 180 traços de video, para um *Buffer* de 80000 e 2000 células, e requisitos de QoS de 10^{-3} e 10^{-9} .

Comportamento dos Estimadores com Tráfego Dinâmico

Na Tabela 4.6 descreve-se o cenário que inclui uma variação do número de traços de vídeo agregados no tempo. Na Figura 4.11 pode-se observar o comportamento dos estimadores. Como observado anteriormente ambos os estimadores se comportam da mesma forma. Observa-se também que ambos os estimadores reagem de forma mais brusca as mudanças na taxa média de transmissão, neste caso o parâmetro $H \rightarrow 1$. As Equações 4.8 e 4.17 de ambos os estimadores desconsideram o *buffer*, logo, as mudanças na variabilidade do tráfego não são mais suavizadas. Por exemplo, a estimativa no tempo 80 segundos, ambos os estimadores calculam um valor cerca de dez vezes a taxa média.

Tempo	0	20	40	60	80
Traços	60	180	360	60	360

Tabela 4.6: Variação do número de traços de video.



Figura 4.11: Taxa Média e banda passante estimada por FEP e GA em função do tempo, na presença de variação do número de traços de video.

4.3.3 Experimentos de Simulação com Tráfego de Dados

Os três cenários de tráfego simulados foram obtido através do agregado de fontes TCP gerando as taxas de 3, 15 e 30 Mbps. Foram utilizados traces reais de enlaces ATM OC3 obtidos no sítio do NLANR [9].

Avaliação da escala de tempo de interesse

Como se pode ver nos gráficos da Figura 4.12, a variação na taxa média de transmissão gerou um comportamento incomum nas estimativas do estimador FEP, o que impediu a definição clara do *timeslot* de interesse. Já as estimativas do GA formaram uma curva descendente com o aumento do *timeslot*.

Este é um caso médio, sendo $H \approx 0, 8$. Quanto a variação das estimativas frente a janela de análise, todos os gráficos das estimativas do FEP apresentaram uma convergência ruim independentemente do *timeslots* considerado, o que indica a presença de grande variabilidade no tráfego em qualquer escala de tempo. As estimativas do GA apresentaram um comportamento de aumento suave da convergência a medida que o *timeslot* aumentava.

Dado os resultados desses testes decidiu-se utilizar um *timeslot* de agregação de 1 segundo como padrão para análise do tráfego de dados, pois a indefinição dos valores do estimador FEP levou a escolha da escala de interesse baseada nos resultados estimados pelo GA.



Figura 4.12: Banda Efetiva v
s Timeslot para os estimadores FEP e GA, para o tráfego de dados com taxas de 3, 15 e 30 Mbps.

Comparação dos Estimadores

Na Figura 4.13, as variações das estimativas para fluxos de 3, 15 e 30Mbps, mostram que os valores estimados pelo FEP, em geral, foram valores intermediários em relação a taxa média e as estimativas do GA, salvo no caso do gráfico de baixo (30 Mbps) no qual a estimativa do FEP superou a do GA em alguns pontos. Neste caso, existe tráfego LRD com $H \ge 0,75$. O valor do desvio padrão do tráfego nestes pontos superou os 5500 Kbps segundo a Tabela 4.3.



Figura 4.13: Taxa Média e banda passante estimada por FEP e GA em função do tempo, para o tráfego de dados com taxas de 3, 15 e 30 Mbps.

Impactos da Variação dos Parâmetros nos Estimadores

No teste dos cenários extremos, quando foi utilizado apenas o tráfego de 15 Mbps, houve uma grande diferença entre os estimadores dependendo do cenário. No cenário observado no gráfico à esquerda da Figura 4.14, os valores estimados pelo FEP se comportaram como um limitante superior justo em relação a taxa média, já no caso observado no gráfico à direita da Figura 4.14 ele se transformou em um limitante superior largo, superando em vários pontos as estimativas do GA. Dado que se utiliza um valor de H de aproximadamente 0, 75, o comportamento das estimativas de banda passante equivalente do FEP, é regido pela Equação 4.15, sendo que a variação no tamanho do *buffer*, em conjunto com a grande variabilidade do tráfego, são os fatores que conduzem a estas estimativas.

No pior caso (gráfico à direita), quando o tamanho do *buffer* é de 2000 células e o requisito de QoS é de 10^{-9} , os valores de desvio padrão necessários para que as estimativas do FEP ultrapassem as do GA são bem menores que os indicados na Tabela 4.3. Por exemplo, para um H = 0,85 basta que o desvio padrão seja maior que 1940Kbps para que isso ocorra, justificando os resultados observados no gráfico.



Figura 4.14: Taxa média e banda passante estimada por FEP e GA em função do tempo, para o tráfego de dados com 15 Mbps, para um *Buffer* de 80000 e 2000 células, e requisitos de QoS de 10^{-3} e 10^{-9} .

Comportamento dos Estimadores com Tráfego Dinâmico

Na tabela 4.7 está descrito o cenário no qual foi inserida uma variação da taxa de transmissão no tempo. Na Figura 4.15 pode-se observar o comportamento dos estimadores. Novamente os valores estimados pelo FEP apresentam um comportamento que ultrapassa as estimativas do GA, isso pode ser observado nos tempos 20, 40 e 80 segundos. Nestes, devido à mudança abrupta na taxa média, o desvio padrão assumiu valores muito altos.

Tempo	0	20	40	60	80
Taxa	3Mb	$15 \mathrm{Mb}$	$30 \mathrm{Mb}$	3Mb	$30 \mathrm{Mb}$

Tabela 4.7: Variação da taxa de transmissão de dados.



Figura 4.15: Taxa Média e banda passante estimada por FEP e GA em função do tempo, na presença de variação na taxa de transmissão de dados.

4.3.4 Experimento de Simulação com Tráfego Sintético de Dados

Os três cenários de tráfego simulado foram obtidos através do agregado de 10, 50 e 100 fontes *Pareto ON/OFF* homogêneas. A configuração das fontes *Pareto* segue os parâmetros proposto por [40] para uma fonte de dados:

Período ON: 250 ms;

Período OFF: 250 ms;

Tamanho do pacote: 53 bytes (1 célula ATM);

Shape: 1,5

Taxa de pico: 400 kbps (943 células/segundo).

Avaliação da escala de tempo de interesse

Como se pode ver nos gráficos da Figura 4.16, da mesma forma que ocorreu com o estimador FEP para o experimento com fluxos reais de dados, observa-se uma grande variabilidade em todas as escalas de tempo. Neste caso os valores estimados pelo GA também apresentaram grande variabilidade. Apesar disso, fica claro que a menor estimativa ocorreu com um *timeslot* de 1 ms para o estimador FEP, escala em que também foi detectada a menor variabilidade. Sendo esta, portanto, a escala utilizada nos experimentos a seguir.



Figura 4.16: Banda Efetiva vs Timeslot para os estimadores FEP e GA, para o tráfego agregado de 10, 50 e 100 fontes sintéticas de dados.

Comparação dos Estimadores

Os gráficos da Figura 4.17 da variação das estimativas, gerados a partir da agregação de 10, 50 e 100 fonte, mostram que praticamente não houve diferença qualitativa entre eles. Os valores estimados pelo FEP permaneceram como um limitante superior justo em relação a taxa média, e as estimativas do GA como um limitante superior largo. O pior caso ocorreu no gráfico para 10 fontes agregadas (gráfico de cima à esquerda) quando uma das estimativas quase atingiu o triplo da outra. O melhor caso ocorreu no gráfico para 100 fontes agregadas (gráfico de baixo). A diferença foi de aproximadamente 50%. Em relação ao mesmo experimento com tráfego real de dados, observa-se que a variabilidade

do tráfego é bem menor, levando a estimativa do FEP a ficar mais próxima da taxa média, apesar de H = 0, 8.



Figura 4.17: Taxa Média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 10, 50 e 100 fontes sintéticas de dados.

Impacto da Variação dos Parâmetros nos Estimadores

Nos cenários ilustrados pelos gráficos da Figura 4.18, foi utilizado apenas o tráfego agregado com 500 fontes. A maior diferença observada entre os cenários de melhor (gráfico à esquerda) e pior (gráfico à direita) caso foi o afastamento de ambos os estimadores em relação a taxa média, as estimativas do FEP passaram de um limitante superior justo para um patamar quase 20% superior. O valor estimado pelo GA passou de aproximadamente 15Mbps para 18Mbps.



Figura 4.18: Taxa média e banda passante estimada por FEP e GA em função do tempo, para o tráfego agregado de 50 fontes sintéticas de dados, para um *Buffer* de 80000 e 2000 células, e requisitos de QoS de 10^{-3} e 10^{-9} .

Comportamento dos Estimadores com Tráfego Dinâmico

Na Tabela 4.8 descreve-se o cenário no qual foi inserida uma variação do número de fontes *Pareto* agregadas no tempo. Na Figura 4.19 pode-se observar o comportamento dos estimadores. Ambos os estimadores mantêm suas características de limitante justo para o FEP e limitante largo para o GA, porém, neste cenário, observa-se que as mudanças de taxa média não refletiram em picos muito abruptos por parte dos estimadores nos pontos 20, 40 e 80 segundos, o que é interessante pois indica a não-captura da variabilidade do tráfego por parte do estimador GA. Os valores estimados pelo FEP permaneceram como limitante superior justo durante todo o período.

Tempo	0	20	40	60	80
Fontes	10	50	100	10	100

Tabela 4.8: Variação do número de fontes ativas de dados.



Figura 4.19: Taxa Média e banda passante estimada por FEP e GA em função do tempo, na presença de variação do número de fontes sintéticas de dados.

4.4 Avaliação dos Resultados

Os valores produzidos pelo estimador GA são um limitante superior largo quando considerado o tráfego LRD com $H \rightarrow 0, 5$, porém suas estimativas para o tráfego com $H \rightarrow 1$ são promissoras, tendo em vista sua simplicidade de implementação. No entanto, sua formulação o torna exageradamente sensível a mudanças na variabilidade do tráfego.

O estimador FEP capturou muito bem as variações impostas pelo tráfego, em toda a faixa de LRD (0, 5 < H < 1). Demonstrou ser robusto o suficiente, para lidar em um ambiente dinâmico com mudanças abruptas de taxa. No entanto, sua complexidade de implementação depende fortemente do estimador de H utilizado. Considerando o estimador AV para o parâmetro de *Hurst* juntamente com o estimador FEP, obtém-se uma ferramenta eficiente para a estimação de banda passante equivalente em ambientes de tempo-real.

Capítulo 5 Redes Auto-Ajustáveis

No projeto de redes, dois importantes requisitos devem ser levados em consideração: qualidade de serviço (QoS) e a eficiência da rede no transporte de informações. As redes de alta velocidade irão suportar uma gama de serviços de voz, vídeo e dados. Portanto, os recursos destas redes (enlaces, *switchs*, etc) serão compartilhados pelos diferentes tipos de mídia, que demandam diferentes requisitos de QoS e geram tráfego com características distintas.

No Capítulo 4, os estimadores de banda passante equivalente foram apresentados. Estes estimadores apresentam uma solução para o problema da definição da quantidade de banda passante requerida por um fluxo, para a provisão de seus requisitos de QoS. Neste Capítulo, introduz-se um *framework* de Redes Auto-Ajustáveis. Este *framework* possibilita a provisão de QoS para todos os fluxos origem-destino em redes multiserviço de alta velocidade.

Redes de alta velocidade suportam o agregado de vários fluxos, pertencentes a diferentes classes de tráfego. Nestas redes, há o problema de se acomodar todos os fluxos existentes, nos enlaces disponíveis, de forma a garantir os requisios de QoS de cada fluxo em particular. Este problema pode ser definido como um problema de Particionamento Ótimo de Banda (OBP), e a sua solução deve definir o roteamento dos fluxos e a quantidade de banda passante que será alocada em cada enlace da rede, para cada classe de tráfego.

5.1 Redes Multiserviço de Alta Velocidade

Uma das principais tecnologias em redes de alta velocidade geograficamente distribuídas é a ATM (Modo de Transferência Assincrôno), que foi amplamente estudada no final dos anos noventa, e é muito utilizada em redes *backbone* públicas e em instituições privadas. As redes baseadas na tecnologia IP^1 , amplamente utilizadas na Internet, não apresentam formas nativas de controle de provisão de QoS. No entando, o uso de GMPLS² com IP sobre WDM³ aponta como sendo a arquitetura provável para as redes de alta velocidade do futuro. Todavia, tais redes ainda estão em fases de estudo e implantação.

5.1.1 Redes ATM

As redes ATM assemelham-se as redes *frame relay* por tirar vantagem da confiabilidade, inerente dos equipamentos digitais modernos, para prover a comutação de pacotes de forma mais rápida do que as antigas redes legados, e.g., X.25 e *frame relay*. Além disso, as taxas de transmissão das redes ATM superam as das redes *frame relay* em várias ordens de grandeza.

As redes ATM efetuam a transferência dos dados em pacotes discretos, e permitem que multiplas conexões lógicas possam ser multiplexadas em uma única interface física. A informação que trafega em cada conexão lógica é organizada em pacotes de tamanho fixo chamados de células. No protocolo da camada ATM, a quantidade de informação de controle requerida em cada célula é mínima, permitindo que se possa trabalhar com altas taxas de transmissão. Além disso, o uso de células de tamanho fixo simplifica o trabalho efetuado em cada nó da rede, facilitando o uso do ATM com altas taxas de transmissão [35]. Em geral, nas redes ATM, são utilizados enlaces óticos OC3 e $OC12^4$ que possuem larguras de banda de, respectivamente, 155Mbps e 622Mbps.

Para que seja possível a interconexão entre protocolos de camadas superiores (ex. TCP/IP), o ATM implementa uma camada de adaptação (AAL⁵), que é responsável basicamente por tratar erros de transmissão, efetuar a segmentação e reconstrução de células em pacotes e vice-versa, e o controle do fluxo.

As conexões lógicas em ATM são chamadas de conexões de canais virtuais (VCCs), e caracterizam as redes ATM como sendo redes comutadas por circuito. Uma VCC é configurada entre dois usuários finais na rede, um nó origem e um nó destino, e um fluxo de taxa variável de células é mantido por esta conexão. Em redes ATM, existe uma segunda camada de processamento que trata de conexões de caminhos virtuais (VPC). Uma VPC engloba um conjunto de VCCs que possuem os mesmos nós origem-destino. A introdução do conceito de caminhos virtuais foi feita para poder simplificar o controle sobre a rede. Desta forma, vários canais virtuais que interligam os mesmos pontos da rede podem ser gerenciados de forma conjunta.

²Generalized Multiprotocol Label Switch

¹Internet Protocol

³Wavelength Division Multiplexing

⁴Optical Carrier

⁵ATM Adaptation Layer

As redes ATM são projetadas para transportar diferentes classes de tráfego simultâneamente, incluindo fluxos de voz, vídeo e dados. A forma como estes fluxos são tratados, depende da classe de serviço ao qual o fluxo foi atribuído. As seguintes categorias de serviços foram definidas para as redes ATM:

- Serviços de tempo-real
 - Taxa constante de bits (CBR): é a classe mais simples, e é utilizada por aplicações que requerem uma taxa constante de transmissão (ex. telefonia);
 - Taxa variável de bits em tempo-real (rt-VBR): é uma classe dedicada a aplicações sensíveis ao retardo (ex. vídeo MPEG4);
- Serviços normais
 - Taxa variável de bits (VBR): é uma classe que garante boa qualidade nas taxas de perda e no retardo, e se baseia na taxa de pico, na taxa média e na variabilidade do fluxo (ex. transações bancárias);
 - Taxa disponível de bits (ABR): é uma classe para aplicações com alta variabilidade que define uma quantidade de banda passante mínima, que pode ser aumentada de acordo com a disponibilidade de banda nos enlaces (ex. interconexão entre LANs⁶);
 - Taxa não especificada de bits (UBR): é uma classe que utiliza a banda passante disponível nos enlaces, é utilizada por aplicações que toleram retardo variável e algumas perdas (ex. fluxos TCP).

5.2 Redes Auto-Ajustáveis

Em [28], introduziu-se um novo conceito de engenharia de tráfego e operação para redes ATM. A este conceito foi dado o nome de "Operação de Redes Auto-Ajustáveis", e trata-se de um método que permite que as redes possam ser rapidamente operadas, e redimensionadas de forma flexível pelo sistema.

O conceito proposto em [28] tratava do ajuste automático de banda passante nos caminhos virtuais (VP) existentes na rede. Os resultados apresentados atestaram a efetividade do método, com uma redução de 30% a 40% no total de banda passante alocada para os VPs, em relação ao sistema tradicional de alocação utilizado nas redes ATM. Basicamente, o sistema monitora a taxa de perda de células nas interfaces da rede, compara com os requisitos de QoS demandados pelo VP, e decide se aumenta, diminui ou mantém a

⁶Local Area Network

quantidade de banda passante alocada. Com esta solução, relativamente simples, foi possível obter um ganho considerável, acima de 30% na utilização da rede, evitando o desperdício de banda passante, e possibilitando que novas aplicações pudessem aproveitar a largura de banda disponível.

Em [41], recomenda-se que as redes multiserviço dividam sua capacidade entre classes de serviço, e adaptativamente modifiquem seus parâmetros de roteamento. Discute-se a necessidade da configuração das redes em tempo-real, através de medições do tráfego, e o conceito de "bandas elásticas", capazes de oferecer diferenciação de QoS entre as classes de tráfego. Em [41], uma metodologia para a criação de redes auto-ajustáveis é apresentada, na qual se destacam os seguintes componentes que devem ser implementados:

- Particionamento adaptativo da rede física em bandas;
- Um método de gerenciamento para cada banda da rede, baseado em um particionamento ótimo do tráfego entre caminhos e conexões independentes, utilizando as conexões independentes para fluxos com baixa intensidade;
- Um algoritmo para alocar, de forma eficiente, a capacidade da rede entre as bandas, levando em conta a capacidade física da rede;
- Um método de controle para redes multiclasse baseado em medições do tráfego, e nos requisitos de QoS de cada banda na rede;
- Um método para a coleta de informações e a implementação das decisões de alocação de capacidade para cada nó.

Baseado na metodologia apresentada em [41], foi proposto um *framework* em [17] para o auto-ajuste e otimização de redes multiserviço de alta velocidade. A divisão da rede física em bandas é discutida, sendo que cada banda corresponde a um serviço, caracterizando uma sub-rede virtual que opera sobre a rede física. É apresentada uma abordagem de otimização baseada em dois passos: o primeiro sendo o cálculo da banda passante equivalente para cada par origem-destino de cada banda; e o segundo, a implementação de um modelo de otimização combinatória para determinar o particionamento das capacidades entre as bandas, que minimize o custo total do sistema, e que garanta a satisfação das restrições físicas dos enlaces da rede. O estimador de banda passante equivalente, utilizado em [17], foi o *Direct EB Allocation*⁷.

Os fatores considerados para a formulação do problema de particionamento das bandas virtuais foram [17]:

1. Bandas: voz, vídeo e dados;

 $^{^7\}mathrm{Este}$ estimador foi avaliado no Capítulo 4

- 2. QoS: probabilidade de perdas de células;
- 3. Roteamento: o tráfego de cada par origem-destino possui dois caminhos alternativos, sendo utilizados os caminhos de menor distância;
- 4. Objetivo: minimizar o custo total da rede;
- 5. Controle de Admissão: o princípio da admissão é o compartilhamento total de cada banda, e o particionamento virtual entre as bandas.

Uma visão geral do *framework* de redes auto-ajustáveis proposto em [17], pode ser visto na Figura 5.1. Nesta figura, estão representados: a topologia física da rede, alimentada por diferentes bandas virtuais, que representam diferentes classes de tráfego; a coleta das estatísticas do tráfego, efetuada nos nós origem para todos os pares de comunicação; o cálculo da banda passante equivalente, a partir das estatísticas coletadas; a resolução do problema de particionamento de banda, que utiliza-se dos valores das bandas passantes equivalente e de informações relativas a rede (topologia e custos e capacidades dos enlaces); por fim, o reajuste das capacidades alocadas e dos caminhos utilizados para todos os pares de comunicação de cada banda virtual.



Figura 5.1: Framework das Redes Auto-Ajustáveis

5.2.1 Modelo para o Particionamento Ótimo de Banda

Utilizando os parâmetros do tráfego dos pares origem-destino, um modelo de otimização para a alocação dinâmica de banda passante, e roteamento dos caminhos pelos enlaces, pode ser definido. Considere a seguinte notação:

- P: conjunto de pares origem-destino;
- B : conjunto de bandas virtuais;
- L : conjunto de enlaces da rede;
- w_{bl} : custo para a banda $b \in B$ no link $l \in L$;
- CAP_l : capacidade total do enlace $l \in L$.
- eb_{pb} : banda passante equivalente estimada para a banda $b \in B$ do par $p \in P$;
- R_{pb} : conjunto de duas rotas alternativas para a banda $b \in B$ do par $p \in P$;
- a_{pbrl} : recebe 1 se a rota $r \in R_{pb}$ para a banda $b \in B$ do par $p \in P$ passa pelo link $l \in L$;
- c_{bl} : variável real que recebe a capacidade alocada para a banda $b \in B$ no link $l \in L$;
- x_{pbr} : variável binária que recebe o valor 1 se o tráfego da banda $b \in B$ do par $p \in P$ segue a rota $r \in R_{pb}$, e 0 caso contrário;

O problema para o Particionamento Ótimo de Banda (OBP) pode ser definido, similarmente a [17], como: encontrar os valores para $c_{bl} \in x_{pbr}$ que satisfaçam o seguinte:

$$Min\sum_{b\in B}\sum_{l\in L}c_{bl}.w_{bl}$$
(5.1)

sujeito a

$$\sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb}.a_{pbrl}.x_{pbr} \leq c_{bl} \qquad b \in B, \ l \in L$$
(5.2)

$$\sum_{b \in B} c_{bl} \leq CAP_l \qquad l \in L \tag{5.3}$$

$$\sum_{eB} x_{pbr} = 1 \qquad b \in B, \ p \in P \tag{5.4}$$

$$c_{bl} \geq 0 \qquad b \in B, \ l \in L \tag{5.5}$$

$$x_{pbr} \in \{0, 1\} \qquad p, b \ como \ acima \ e \ r \in R_{pb}. \tag{5.6}$$

A Função Objetivo 5.1 minimiza o custo total do sistema. A Restrição 5.2 diz que, para cada serviço, o tamanho de uma banda em cada enlace c_{bl} deve ser capaz de comportar toda a demanda de tráfego desta classe de todos os pares origem-destino. Os requisitos de QoS estão embutidos nesta restrição através da banda passante equivalente eb_{pb} . A Restrição 5.3, garante que a soma das bandas não exceda a capacidade física do enlace, e
a Restrição 5.4 garante que apenas um caminho seja escolhido para cada banda em cada par origem-destino.

O problema para o Particionamento Ótimo de Banda (OBP), como apresentado acima, pertence a classe dos problemas de Fluxo Multiproduto Inteiro, com uma particularidade pois, neste caso, as capacidades alocadas por fluxo (c_{bl}) , são variáveis do problema. O problema OBP está estruturado na forma de um problema de Programação Linear Inteira Mista, pois todas as equações envolvidas no problema são lineares, e por possuir uma variável matriz contínua (c_{bl}) e uma variável matriz binária (x_{pbr}) .

O problema OBP, pode ser resolvido utilizando-se qualquer um dos métodos apresentados no Capítulo 3. Porém, para grandes instâncias do problema (redes com muitos nós e enlaces), provavelmente, os métodos deteminísticos tornem-se inviáveis, requerendo um tempo muito grande para sua solução. Neste caso, deve-se optar por abordagens de solução heurísticas ou aproximadas, que sejam capazes de solucionar o problema em tempo-real. Apesar de não garantirem a solução ótima, tais abordagens são, em muitos casos, a única forma de se lidar com problemas muito complexos.

Após resolver o problema de otimização, obtém-se o novo esquema de roteamento da rede, e as novas capacidades a serem alocadas para cada banda virtual, nos enlaces da rede. Em [17] é utilizado um método de controle global, para efetivar as novas configurações na rede. Este método, assume o acesso direto a todos os nós, permitindo a configuração imediata de toda a rede. Todavia, existem na literatura, abordagens mais complexas, como a apresentada em [29], que utiliza um método de controle local, extendendo os conceitos das redes auto-ajustáveis para cenários distribuídos.

Capítulo 6

Técnicas Fuzzy na Otimização de Redes Auto-Ajustáveis

Neste Capítulo, propõem-se modelos fuzzy para o problema do Particionamento Ótimo de Banda, apresentado no Capítulo 5. Tais modelos, possuem a capacidade de lidar com a variabilidade do tráfego servido, e oferecem uma flexibilidade muito grande na adição de novos objetivos ao problema, permitindo a criação de modelos que contemplem, de forma mais ampla, requisitos de QoS para o tráfego servido pela rede em uma rede auto-ajustável.

Para a avaliação dos modelos matemáticos, realizaram-se simulações com o intuito de criar as condições existentes em uma rede multiserviço de alta velocidade real. Nas simulações, foram utilizados agregados de fontes para a geração de tráfego de voz, vídeo e dados com altas taxas de transmissão.

Ao longo das simulações, as seguintes estatísticas foram coletadas: taxa média de transmissão; desvio padrão; parâmetro de *Hurst* e a banda passante equivalente estimada. Em relação a Qualidade de Serviço, coletou-se, também, a taxa de utilização da rede, a taxa média de perda de pacotes e o retardo médio entre chegadas.

Em seguida, apresenta-se a configuração dos experimentos de simulação, e, posteriormente, descreve-se os modelos matemáticos propostos. Por fim, discute-se os resultados numéricos obtidos.

6.1 Configuração dos Experimentos de Simulação

Nesta seção, descreve-se o ambiente utilizado para a realização dos experimentos de simulação. A seguir, é apresentada uma descrição geral do ambiente de simulação, seguida pela apresentação da topologia de rede utilizada nos experimentos e pela descrição das características do tráfego gerado. Por fim, apresenta-se uma descrição detalhada da dinâmica do processo de simulação utilizado.

Para construir o ambiente de simulação, utilizou-se o simulador NS-2 [34], que é um simulador de eventos discretos muito utilizado na literatura. Em linhas gerais, as tarefas realizadas para a criação do ambiente de simulação foram:

- Configuração do simulador NS-2 com uma topologia de rede alimentada por fluxos de voz, vídeo e dados;
- Desenvolvimento de alguns programas e scripts para:
 - Coleta dos dados dos fluxos da rede;
 - Cálculo das estatísticas de interesse;
 - Estimação da banda passante equivalente e do parâmetro de Hurst;
 - Resolução do problema de particionamento de banda;
 - Formatação dos resultados para reconfiguração da rede.

Nos experimentos de simulação realizados, utilizou-se o estimador FEP para o cálculo da banda passante equivalente, e o estimador AV para o cálculo do parâmetro de $Hurst^1$.

6.1.1 Topologia de Rede Utilizada nos Experimentos de Simulação

Para a avaliação do *framework* de redes auto-ajustáveis, utiliza-se a topologia de rede proposta em [17]. A topologia é composta de cinco nós e sete enlaces com 155Mbps de capacidade de transmissão e 1 ms de retardo. Na Figura 6.1, pode-se visualizar a rede, cu-jas distâncias entre nós, em kilometros, estão rotuladas nos enlaces. Define-se os custos por unidade de banda, para todas as classes de serviço, como sendo [2, 6; 2, 5; 2, 4; 2, 3; 2, 4; 2, 5; 2, 5], para cada enlace enumerado na Figura 6.1.

¹Ambos os estimadores foram estudados no Capítulo 4.



Figura 6.1: Topologia da Rede, com a distância entre os nós em Kms

Considera-se dez pares origem-destino, sendo que, para cada par considera-se dois caminhos possíveis, escolhidos como o menor caminho e o segundo menor caminho na rede. Os pares origem-destino e os seus respectivos caminhos estão descritos na Tabela 6.1.

Par	Nós	Caminho 1	Caminho 2
1	1 - 2	$1 \longrightarrow 2$	$1 \rightarrow 3 \rightarrow 2$
2	1 - 3	$1 \longrightarrow 3$	$1 \rightarrow 2 \rightarrow 3$
3	1 - 4	$1 \rightarrow 3 \rightarrow 4$	$1 \rightarrow 2 \rightarrow 4$
4	1 - 5	$1 \rightarrow 3 \rightarrow 5$	$1 \rightarrow 2 \rightarrow 5$
5	2 - 3	$2 \longrightarrow 3$	$2 \rightarrow 4 \rightarrow 3$
6	2 - 4	$2 \longrightarrow 4$	$2 \rightarrow 3 \rightarrow 4$
7	2 - 5	$2 \longrightarrow 5$	$2 \rightarrow 3 \rightarrow 5$
8	3 - 4	$3 \longrightarrow 4$	$3 \rightarrow 2 \rightarrow 4$
9	3 - 5	$3 \longrightarrow 5$	$3 \rightarrow 2 \rightarrow 5$
10	4 - 5	$4 \rightarrow 3 \rightarrow 5$	$4 \rightarrow 2 \rightarrow 5$

Tabela 6.1: Pares de nós origem-destino e seus possíveis caminhos de interligação

Utilizou-se como política de gerenciamento de filas o CBQ ou *Enfileiramento Baseado* em Classe, que permite a comutação de pacotes baseada em classes de serviço, bem como a alocação dinâmica de banda passante para cada classe. A política CBQ é implementada no NS-2 através da criação de filas virtuais, representando cada classe de tráfego, em todos os enlaces da rede. Configurou-se cada fila virtual com uma capacidade de 424000 bytes (8000 células ATM), referente ao tamanho do *buffer* de um roteador comercial, definido na Seção 4.2.1, do Capítulo 4.

6.1.2 Características do Tráfego Utilizado nos Experimentos de Simulação

A rede provê três tipos de serviço: voz, vídeo e dados, e é logicamente dividida em três bandas virtuais, uma para cada classe de tráfego. O tráfego para cada classe de serviço foi gerado no simulador NS-2 através do agregado de fontes sintéticas para o tráfego de voz e dados, e do agregado de traços reais, para o tráfego de vídeo. Para a geração do tráfego de voz utilizou-se fontes *Exponecial On-Off*. Para o tráfego de dados utilizou-se fontes *Pareto On-Off* e para o tráfego de vídeo utilizou-se traços reais de vídeo comprimido. Todas as fontes de tráfego foram configuradas da mesma forma que a apresentada no Capítulo 4, nas Seções 4.3.1, 4.3.2 e 4.3.4.

Como, neste experimento, pretende-se simular um ambiente real de rede, a taxa de chegada e o tempo de duração das fontes de tráfego são simuladas através de distribuições exponenciais. Os parâmetros utilizados para as distribuições exponenciais foram baseados em [21], [20] e [7], e podem ser vistos na Tabela 6.2.

Classes de	Média da distribuição	Média da distribuição
Tráfego	de chegada de fontes	da duração das fontes
Voz	$400 \mathrm{ms}$	300 s
Vídeo	1 s	1000 s
Dados	$400 \mathrm{\ ms}$	300 s

Tabela 6.2: Parâmetros das distribuições Exponenciais de chegada e de duração, para as fontes, de cada classe de tráfego.

Implementou-se no simulador, um limitador de tráfego, para garantir um valor máximo para as taxas médias de transmissão de cada fluxo na rede, de forma que, enquanto um determinado fluxo mantivesse sua taxa média no máximo, não era permitida a chegada de novas fontes. O número máximo de fontes ativas permitidas, e suas respectivas taxas máximas de transmissão estão descritas na Tabela 6.3.

Classes de	Número máximo	Taxa média
Tráfego	de fontes ativas	máxima
Voz	400	12 Mbps
Vídeo	90	$40 { m ~Mbps}$
Dados	$\overline{65}$	15 Mbps

Tabela 6.3: Número máximo de fontes ativas e taxa máxima de transmissão, por fluxo, de cada classe de tráfego.

No processo de geração de tráfego para os experimentos de simulação, despreza-se os primeiros 300 segundos de simulação, que são referentes a fase transiente da rede. Este intervalo era necessário para que a dinâmica da ativação de fontes de tráfego atingisse um equilíbrio, não caracterizando mais uma tendência linear de crescimento presente no ínicio do processo.

6.1.3 Dinâmica dos Experimentos de Simulação

A Figura 6.2 ilustra a configuração do ambiente de simulação. Definiu-se tal configuração, devido a algumas restrições inerentes ao simulador *NS-2* e aos recursos de computação disponíveis. Primeiramente, o simulador não possui a facilidade de *check points*, o que impede a continuação de uma simulação após esta ter sido abortada. Em segundo lugar, devido ao fato de se tratar da simulação de uma rede de alta velocidade, mesmo considerando uma topologia de pequeno porte, a taxa total de transmissão agregada aproximou-se de 1 Gbps. Logo, a quantidade de dados a serem analisados ocupavam muito espaço em disco, algo em torno de 500 Mbytes por segundo de simulação.

Para contornar esta situação, optou-se por configurar o simulador de forma que, a cada momento de reconfiguração da rede, todos os resultados necessários fossem calculados. Desta forma, pode-se remover os dados referentes ao período anterior, evitando a necessidade de acumular muita informação em disco. Para tal, criou-se uma configuração de monitoramento capaz de avaliar o estado da rede de forma incremental, o que permite a análise dos resultados a medida em que a simulação avança. Esta configuração, agiliza todo o processo de experimentação, já que não é necessário aguardar o término de uma simulação para iniciar o cálculo de seus resultados.

Para a resolução do problema de otimização, considerou-se, inicialmente, a utilização de algoritmos genéticos, devido a sua vasta utilização na resolução de problemas de otimização fuzzy, presente na literatura. No entanto, considerando a topologia apresentada, o tempo necessário para o cálculo da solução através deste tipo de algoritmo, foi superior a 1 minuto. Este resultado impede a sua aplicação em um ambiente de tempo-real,

não sendo, portanto, uma contribuição válida para o *framework* de redes auto-ajustáveis. Dado o desempenho do algoritmo genético, decidiu-se avaliar o desempenho através de algorítmos determinísticos. Para isto, optou-se pela utilização do pacote de otimização *Xpress - MP* [36], que implementa um algoritmo *Branch and Bound* para localizar, e garantir, a solução ótima do problema. Com este otimizador, pode-se resolver o mesmo problema em menos de 10 milissegundos.



Figura 6.2: Configuração do Experimento de Simulação

Pode-se seguir a sequência de passos existentes durante cada simulação a partir da numeração dos eventos existentes na Figura 6.2. São eles:

1. O simulador NS-2 gera um arquivo em disco contendo todos os eventos ocorridos durante a simulação. Definiu-se um período entre reconfigurações da rede de dois segundos. Ao final de cada período, o simulador grava em disco os eventos ocorridos.

Em seguida, sua execução é interrompida, enquanto aguarda o término do processo de otimização, quando então poderá reconfigurar a rede;

- 2. A partir do arquivo de eventos, calcula-se a taxa de perda de pacotes e o retardo entre chegadas para cada classe de tráfego de cada par origem-destino da rede;
- 3. Agrega-se o tráfego de cada classe utilizando os seguintes valores de *timeslot*: 10 milisegundos para voz, 1 segundo para vídeo e 1 milisegundo para dados. Valores estes, definidos nas seções 4.3.1, 4.3.2 e 4.3.4 do Capítulo 4;
- 4. Grava-se um arquivo contendo os fluxos de tráfego agregado, para cada classe de tráfego de todos os pares origem-destino;
- 5. Utiliza-se o estimador AV para o cálculo do parâmetro de Hurst de todos os fluxos. Considerando o estudo realizado no Capítulo 4, definem-se valores de referência para as estimativas de H, em relação as classes de tráfego, sendo: H = 0,7 para voz, H = 0,8 para dados e H = 0,9 para vídeo. Implementou-se um sistema de controle das estimativas durante a simulação, de forma que, permitiu-se uma variação máxima de 10%, em relação aos valores de referência, para se aceitar um valor de H estimado. Esta medida é necessária para se evitar a utilização de valores irreais², gerados pela existência de efeitos não estacionários no tráfego.

Para permitir uma maior robustez no cálculo de H, todo o tráfego agregado, de cada classe de serviço, para cada par origem-destino, foi acumulado ao longo de toda a simulação. Pode-se assim, obter o parâmetro H referente a todo o histórico da simulação até o momento da medição. Quanto maior o número de pontos analisados pelo estimador AV, melhor são as suas estimativas;

- 6. Calcula-se a taxa média e o desvio padrão, e utiliza-se o estimador FEP para o cálculo da banda passante equivalente de cada fluxo. Como requisito de QoS, foi utilizada a probabilidade de perda de pacotes na rede, com o valor 10⁻⁶. O tamanho de *buffer* utilizado foi de 424000 bytes;
- 7. Gera-se um vetor com os valores da banda passante equivalente de cada classe de serviço para cada par origem-destino da rede;
- 8. Envia-se os seguintes parâmetros para o otimizador: conjunto de enlaces, conjunto de nós, conjunto de bandas virtuais, custos de utilização, conjunto de caminhos na rede e banda passante equivalente. Devido a problemas com a licença de uso do otimizador *Express MP*, não foi possível instalá-lo na mesma máquina onde

 $^{^2 \}mathrm{Valores}$ for a da escala [0,5;1), ou com desvios superiores a 10% do esperado.

executa-se o simulador. Para solucionar este problema, criou-se um programa que envia os dados a serem otimizados para a máquina onde está o otimizador, e outro programa que executa o otimizador e retorna os resultados. O tempo gasto com todo este procedimento, incluindo o envio dos dados, a resolução do problema de otimização e o recebimento da resposta, é inferior a um segundo;

- 9. O otimizador retorna os resultados contendo as quantidades de banda a serem alocadas por classe de tráfego em cada enlace, e os caminhos a serem utilizados na rede, para cada classe de cada par origem-destino;
- Uma nova tabela de roteamento e uma nova tabela de alocação de banda são enviadas ao simulador. Este, reconfigura a rede, remove o arquivo de eventos atual e retoma a simulação.

6.2 Modelos Propostos para o Problema do Particionamento Ótimo de Banda

Discutiu-se no Capítulo 5 a necessidade da criação de um modelo matemático para se otimizar a distribuição de banda passante, bem como definir os caminhos a serem seguidos na rede pelos fluxos de cada classe de serviço. Apresentou-se, também, o modelo utilizado em [17] para a solução do problema de Particionamento Ótimo de Banda. Este modelo será referenciado, de agora em diante, como o modelo clássico.

O *framework* das redes auto-ajustáveis utiliza estimadores de banda passante equivalente baseados em medições, que são ferramentas baseadas em equações aproximadas. Outro fator que traz incerteza para o sistema é o fato de as realocações da rede ocorrerem em intervalos de dois segundos. Portanto, ao se estimar a banda passante requerida por um determinado fluxo, assume-se que este fluxo manterá suas características por um tempo similar ao intervalo entre realocações. Como o tráfego pode sofrer alterações significativas no intervalo entre realocações da rede, um modelo de otimização capaz de relaxar as restrições do sistema pode ser uma boa alternativa ao modelo clássico.

Tendo em vista estes fatos, considerou-se que o problema de Particionamento Otimo de Banda deveria ser remodelado de forma a considerar as imprecisões impostas pelo sistema. Utilizou-se, assim, técnicas de otimização fuzzy para se representar o objetivo a ser otimizado, e, desta forma, buscar amenizar as imprecisões existentes no *framework* de redes auto-ajustáveis.

Inicialmente, adaptou-se o modelo clássico para um modelo fuzzy. Em seguida, novas restrições fuzzy foram sendo acrescentadas ao problema, buscando dirimir alguns problemas encontrados na avaliação fuzzy. Ao todo foram criados, e avaliados, quatro modelos de otimização fuzzy. Cada novo modelo, representa uma versão mais restrita do modelo anterior, salvo no quarto, e último modelo, no qual propõe-se uma relaxação da função objetivo do problema.

De forma geral, as diferenças existentes, e os objetivos buscados por cada um dos modelos fuzzy foram os seguintes:

- Modelo 1: Relaxação na minimização do custo total da rede, buscando obter valores mais largos para as alocações de banda passante. Com isto, pretende-se minimizar as perdas de pacotes nas filas da rede;
- Modelo 2: Acrescentou-se restrições fuzzy para induzir a maximização da banda passante alocada por fluxo. Novamente, busca-se a minimização das perdas de pacotes na rede;
- Modelo 3: Acrescentou-se restrições fuzzy para induzir a minimização do número de fluxos existentes por enlace. Neste modelo, buscou-se equilibrar o balanceamento das perdas entre os pares de comunicação;
- Modelo 4: Particionou-se a função objetivo, relaxando a maximização do grau de satisfação do problema, criando um fator para cada conjunto de restrições fuzzy. Com isso, buscou-se não inibir o aumento da satisfação dos conjuntos de restrições, podendo, assim, alcançar valores de satisfação maiores. Este modelo, reproduz as características do terceiro modelo, porém, busca atingir resultados melhores na taxa de perdas de pacotes, e no balanceamento entre os pares de comunicação.

A seguir, os quatro modelos de otimização fuzzy são descritos. Todos os modelos apresentados utilizam a seguinte notação:

- ${\cal P}\,:\,{\rm conjunto}$ de pares origem-destino;
- B : conjunto de bandas virtuais;
- L : conjunto de links da rede;
- w_{bl} : custo para a banda $b \in B$ no link $l \in L$;
- CAP_l : capacidade total do link $l \in L$.
- eb_{pb} : capacidade estimada para a banda $b \in B$ do par $p \in P$;
- R_{pb} : conjunto de duas rotas alternativas para a banda $b \in B$ do par $p \in P$;

 a_{pbrl} : recebe 1 se a rota $r \in R_{pb}$ para a banda $b \in B$ do par $p \in P$ passa pelo link $l \in L$;

 λ : variável real que define o grau de satisfação da solução obtida;

- c_{bl} : variável real que recebe a capacidade alocada para a banda $b \in B$ no link $l \in L$;
- x_{pbr} : variável binária que recebe o valor 1 se o tráfego da banda $b \in B$ do par $p \in P$ segue a rota $r \in R_{pb}$, e 0 caso contrário;

6.2.1 Primeiro Modelo Fuzzy

Adaptou-se o modelo clássico, apresentado no Capítulo 5, para um modelo fuzzy, a partir da transformação da função objetivo em uma restrição fuzzy. O modelo de otimização clássico busca minimizar o custo total da rede. Este custo é calculado através da soma das multiplicações das capacidades alocadas pelos custos dos enlaces, para cada classe de serviço, em cada enlace. Tem-se que, o valor da capacidade alocada para uma classe pode variar entre: a soma das banda passante equivalentes estimadas para os fluxos existentes no enlace; e a capacidade física do enlace. De fato podem ocorrer casos em que a capacidade alocada em um enlace, para uma classe de serviço, seja nula, o que implica na ausência de qualquer fluxo desta classe de serviço passando pelo enlace em questão.

Portanto, pode-se considerar a função de pertinência que representa a minimização do custo total da rede, de forma simplificada, como apresentado na Figura 6.3. O custo da capacidade alocada situa-se entre o custo da banda passante equivalente e o custo da capacidade física do enlace, de forma que, ao maximizar o grau de satisfação λ , minimiza-se o custo total da rede.



Figura 6.3: Função de pertinência para a minimização do custo total da rede

Pode-se, então, substituir a função objetivo:

$$Min\sum_{b\in B}\sum_{l\in L}c_{bl}.w_{bl} \tag{6.1}$$

por uma restrição fuzzy:

$$1 - \frac{\sum_{b \in B} \sum_{l \in L} c_{bl}.w_{bl} - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb}.a_{pbrl}.x_{pbr}.w_{bl}}{\sum_{l \in L} CAP_l.Max_b(w_{bl}) - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb}.a_{pbrl}.x_{pbr}.w_{bl}} \ge \lambda$$
(6.2)

Para o cálculo do custo máximo da rede, utilizou-se a equação $\sum_{l \in L} CAP_l Max_b(w_{bl})$, que representa a soma das capacidades físicas de cada enlace da rede multiplicada pelo maior custo existente entre as classes de serviço, ou seja, supondo que o custo para o tráfego de vídeo seja o maior entre os existentes, considera-se que este fluxo ocupa todo o enlace.

A Restrição 6.2 derivada do Objetivo 6.1, é uma função não linear, pois nela ocorre uma divisão entre variáveis do problema, x_{pbr} . Tal característica é indesejável em um modelo de otimização, pois requer métodos específicos, em geral aproximados, para a resolução do problema. Para se evitar lidar com um modelo de otimização inteira não linear, optou-se por remover a variável x_{pbr} do divisor da Restrição 6.2. Com esta alteração, o valor de λ , que antes variava entre 0 e 1, agora passou a variar entre 0 e o resultado da divisão (Equação 6.3).

$$\frac{\sum_{b\in B}\sum_{l\in L}\sum_{p\in P}\sum_{r\in R_{pb}}eb_{pb}.a_{pbrl}.x_{pbr}.w_{bl}}{\sum_{l\in L}CAP_l.Max_b(w_{bl})}$$
(6.3)

Este valor é influenciado diretamente pela taxa de utilização do enlace para a classe de serviço em questão. Quanto maior for a utilização do enlace, mais próximo de 1 será o valor de λ . De fato, isto não afeta o comportamento geral do modelo, pois o que se busca é a maximização de λ . O único impasse relacionado a esta adaptação é o fato de que a nova restrição não representa mais uma função de pertinência, segundo sua definição restrita, pois sua imagem não esta contida no intervalo [0; 1]. Define-se, então, o primeiro modelo fuzzy para o problema do Particionamento Ótimo de Banda como:

sujeito a

$$1 - \frac{\sum_{b \in B} \sum_{l \in L} c_{bl} \cdot w_{bl} - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb} \cdot a_{pbrl} \cdot x_{pbr} \cdot w_{bl}}{\sum_{l \in L} CAP_l \cdot Max_b(w_{bl})} \geq \lambda$$

$$\sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb} \cdot a_{pbrl} \cdot x_{pbr} \leq c_{bl} \quad b \in B, \ l \in L$$

$$\sum_{b \in B} c_{bl} \leq CAP_l \quad l \in L$$

$$\sum_{r \in R_{pb}} x_{pbr} = 1 \quad b \in B, \ p \in P$$

$$c_{bl} \geq 0 \quad b \in B, \ l \in L$$

$$\lambda \geq 0$$

$$(6.4)$$

 $x_{pbr} \in \{0,1\}$ $p, b \ como \ acima \ e \ r \in R_{pb}.$

6.2.2 Segundo Modelo Fuzzy

Propõe-se uma segunda abordagem, na qual se busca a maximização da capacidade alocada para cada banda em cada enlace da rede. A partir da adição de uma nova restrição fuzzy ao modelo anterior, pode-se alcançar este objetivo. Como a capacidade alocada a cada fluxo da rede pode variar entre 0 e a capacidade física do enlace, considera-se uma função de pertinência para esta restrição, como a ilustrada na Figura 6.4.



Figura 6.4: Função de pertinência para a maximização da quantidade de banda passante alocada por fluxo

Adicionando-se a Restrição 6.5 ao modelo anterior, obtém-se o segundo modelo fuzzy

para o problema do Particionamento Ótimo de Banda:

$$Max \quad \lambda$$

sujeito a

$$1 - \frac{\sum_{b \in B} \sum_{l \in L} c_{bl} . w_{bl} - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb} . a_{pbrl} . x_{pbr} . w_{bl}}{\sum_{l \in L} CAP_l . Max_b(w_{bl})} \geq \lambda$$

$$\sum_{l \in L} CAP_l . Max_b(w_{bl})$$

$$\sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb} . a_{pbrl} . x_{pbr} \leq c_{bl} \quad b \in B, \ l \in L$$

$$\sum_{p \in P} \sum_{r \in R_{pb}} eb_{pb} . a_{pbrl} . x_{pbr} \leq CAP_l \quad l \in L$$

$$\sum_{b \in B} c_{bl} \leq CAP_l \quad l \in L$$

$$\sum_{r \in R_{pb}} x_{pbr} = 1 \quad b \in B, \ p \in P$$

$$c_{bl} \geq 0 \quad b \in B, \ l \in L$$

$$\lambda \geq 0$$
(6.6)

 $x_{pbr} \in \{0, 1\}$ $p, b \ como \ acima \ e \ r \in R_{pb}.$

6.2.3 Terceiro Modelo Fuzzy

Além de se buscar minimizar o custo total da rede, e maximizar a capacidade alocada por fluxo, propõe-se a minimização no número de fluxos de cada classe por enlace da rede. Com a adição desta nova restrição, o modelo deverá ser capaz de garantir um melhor balanceamento na utilização dos recursos da rede.

O número de fluxos existentes, de uma determinada classe, em um enlace, pode variar entre 0 e um valor máximo que depende da quantidade de rotas existentes na rede que utilizam o enlace em questão. Para se calcular o número máximo de fluxos possíveis, deve-se avaliar todos os caminhos para todos os pares de comunicação. Caso qualquer um dos caminhos existentes, de cada par, utilize o enlace em questão, deve-se somar um ao total de fluxos.

A matriz binária a_{pbrl} do modelo fuzzy, indica se um determinado enlace l é utilizado pelo caminho r do par origem-destino p para uma determinada classe de tráfego b. A variável binária x_{pbr} indica qual caminho deve ser utilizado pelo par p da classe de tráfego b. Portanto, $a_{pbrl}.x_{pbr}$ é igual a 1 se o enlace l é utilizado por um fluxo da classe de serviço b do par p. O caso de utilização máxima de um enlace ocorre quando se considera que sempre há um caminho que utiliza o enlace, i.e., quando x_{pbr} é sempre 1. A função de pertinência para a nova restrição fuzzy está ilustrada na Figura 6.5.



Figura 6.5: Função de pertinência para a minimização do número de fluxo de cada classe de serviço, por enlace

A função de pertinência pode, então, ser representada pela Restrição 6.7. O divisor desta equação nada mais é do que a representação da função lógica $(a_{pb1l} \lor a_{pb2l})$ para todos os pares origem-destino. Como existe apenas dois caminhos possíveis para cada par, utiliza-se r = 1 e r = 2.

Define-se, então, o terceiro modelo fuzzy para o problema do Particionamento Ótimo de Banda como:

sujeito a

$$1 - \frac{\sum_{b \in B} \sum_{l \in L} c_{bl} . w_{bl} - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} e^{b} e_{bpl} . a_{pbrl} . x_{pbr} . w_{bl}}{\sum_{l \in L} CAP_{l} . Max_{b}(w_{bl})} \geq \lambda$$

$$\frac{\sum_{b \in B} c_{bl}}{CAP_{l}} \geq \lambda \quad l \in L$$

$$1 - \frac{\sum_{p \in P} \sum_{r \in R_{pb}} a_{pbrl} . x_{pbr}}{\sum_{p \in P} \frac{a_{pbl} . a_{pbrl} . x_{pbr}}{(a_{pbl} . a_{pbrl} . x_{pbr})}} \geq \lambda \quad b \in B, \ l \in L \quad (6.7)$$

$$\sum_{p \in P} \sum_{r \in R_{pb}} e^{b} e_{bpb} . a_{pbrl} . x_{pbr} \leq c_{bl} \quad b \in B, \ l \in L$$

$$\sum_{b \in B} c_{bl} \leq CAP_{l} \quad l \in L$$

$$\sum_{r \in R_{pb}} x_{pbr} = 1 \quad b \in B, \ p \in P$$

$$c_{bl} \geq 0 \quad b \in B, \ l \in L$$

$$\lambda \geq 0$$

$$(6.8)$$

 $x_{pbr} \in \{0, 1\}$ $p, b \ como \ acima \ e \ r \in R_{pb}.$

6.2.4 Quarto Modelo Fuzzy

Até agora, foram definidos modelos de otimização com a maximização de uma única variável λ , representando o grau de satisfação para todas restrições fuzzy do problema. Como, no último modelo fuzzy considera-se três conjuntos de restrições fuzzy, criou-se uma variável λ para representar o grau de satisfação de cada conjunto. Com esta alteração, buscou-se evitar a interferência existente entre os conjuntos de restrições, o que poderá melhorar a qualidade do resultado final da maximização.

Finalmente, define-se o quarto modelo fuzzy para o problema do Particionamento Ótimo de Banda como:

$$Max \quad \lambda_1 + \lambda_2 + \lambda_3$$

sujeito a

$$1 - \frac{\sum_{b \in B} \sum_{l \in L} c_{bl}.w_{bl} - \sum_{b \in B} \sum_{l \in L} \sum_{p \in P} \sum_{r \in R_{pb}} e^{b} e$$

 $x_{pbr} \in \{0, 1\}$ $p, b \ como \ acima \ e \ r \in R_{pb}.$

6.2.5 Complexidade dos Modelos de Otimização

Na Tabela 6.4 estão descritos os resultados da avaliação do tempo de execução para todos os modelos matemáticos apresentados. Além da topologia avaliada nesta dissertação, foram utilizadas mais duas topologias reais, o *backbone* da empresa de telecomunicações americana $AT \mathscr{C}T$ e o *backbone* legado da rede interestadual americana *NSFNet*.

Os tempos indicados na Tabela 6.4 se referem ao tempo de CPU do processo, e foram obtidos em uma máquina AMD Athlon 1.2GHz com 768MB de memória.

Topologia				Tempo	Número de	Número de
(Modelos)	Nós	Enlaces	Pares	de CPU	Variáveis	Restrições
Avaliada	5	7	10			
Clássico				$0{,}092~{\rm s}$	81	81
Fuzzy 1				$0{,}088~{\rm s}$	82	82
Fuzzy 2				$0{,}092~{\rm s}$	82	88
Fuzzy 3				$0,\!125 {\rm ~s}$	82	109
Fuzzy 4				$0{,}397~\mathrm{s}$	84	113
AT&T	10	13	45			
Clássico				0,111 s	309	228
Fuzzy 1				$0{,}126~{\rm s}$	310	229
Fuzzy 2				$0{,}130~{\rm s}$	310	241
Fuzzy 3				$0,227 {\rm \ s}$	310	280
Fuzzy 4				$0{,}907~\mathrm{s}$	312	284
NSFNet	14	20	91			
Clássico				$0,169 { m \ s}$	606	415
Fuzzy 1				$0{,}190~{\rm s}$	607	416
Fuzzy 2				$0{,}204~{\rm s}$	607	435
Fuzzy 3				$0{,}843~{\rm s}$	607	495
Fuzzy 4				$3{,}420~\mathrm{s}$	609	499

Tabela 6.4: Tempos de execução dos modelos matemáticos.

A complexidade dos modelos cresce com o acréscimo de variáveis e restrições, atingindo o máximo no quarto modelo fuzzy, que apresenta os piores tempos de execução. No pior caso, com a topologia *NSFNet* (14 nós e 20 enlaces), o quarto modelo fuzzy demanda pouco mais de 3 segundos para sua execução.

O tempo de execução pode ser reduzido facilmente com a utilização de máquinas mais avançadas, ou através de hardware dedicado. Vê-se, então, que mesmo com topologias maiores, como as existentes nos *backbones* de alta velocidade, pode-se resolver os modelos de otimização fuzzy através de métodos de programação inteira em pouco espaço de tempo (aproximadamente 1 segundo). Portanto, o aumento de complexidade existente nos modelos fuzzy não impede a sua utilização no *framework* de redes auto-ajustáveis.

6.3 Resultados Numéricos

Nesta seção, os resultados obtidos com os experimentos de simulação são apresentados. Todos os modelos de otimização propostos são avaliados comparativamente com o modelo de otimização clássico, discutido na Seção 5.2.1 do Capítulo 5. Para a execução dos experimentos de simulação, utilizaram-se máquinas com dois processadores Intel Xeon de 2,4GHz e 1 gigabyte de memória.

6.3.1 Estatísticas dos Cenários Utilizados nos Experimentos de Simulação

Nesta seção, as estatísticas comuns do tráfego para todos os experimentos de simulação são apresentadas. Como se utilizou as mesmas sementes aleatórias para a geração do tráfego para todos os experimentos, e pelo fato de que o processo de realocação da rede não afeta o tráfego de chegada, todas as simulações apresentam as mesmas estatísticas para o tráfego servido. Desta forma, pode-se comparar com precisão, os resultados apresentados pelos modelos de otimização ao longo de toda a simulação.

Os gráficos apresentados nesta seção foram obtidos a partir do cálculo da média das estatísticas entre todos os pares origem-destino para as classes de voz, vídeo e dados, ao longo da simulação. Cada figura apresentada contém quatro gráficos, que descrevem a taxa média de transmissão, o desvio padrão, o parâmetro de *Hurst*, calculado pelo estimador AV [1], e a banda passante equivalente, calculada pelo estimador FEP [8].

Dois cenários de tráfego foram avaliados, o primeiro é o resultado da agregação de tráfego gerada pelas fontes apresentadas na Seção 6.1.2 do Capítulo 6. Tal configuração levou a rede a operar com uma taxa de utilização de 75%. No total, foram simulados 160 segundos de geração de tráfego, o que demandou 26 dias de execução para cada simulação.

O segundo cenário foi configurado de forma a imprimir mudanças bruscas na taxa média do tráfego. Na realidade, configura-se um comportamento de "onda quadrada" para que se possa avaliar a robustez dos modelos de particionamento ótimo de banda. No total, foram simulados 90 segundos de geração de tráfego, com taxa média variando abruptamente a cada 10 segundos. Cada simulação demandou 14 dias de execução.

Primeiro Cenário de Tráfego

Na Figura 6.6 são apresentadas as estatísticas para o tráfego de voz. Os valores da taxa média variaram entre 10400 Kbps e 11000 Kbps. Os valores do desvio padrão variaram entre 500 Kbps e 660 Kbps, o que indica um tráfego com baixa variabilidade. As estimativas para H permaneceram constantes em 0,7 por 110 segundos, provavelmente devido a um número insuficiente de pontos a serem analisados até então. Após este período

as estimativas apresentaram uma variação entre 0,63 e 0,72. As estimativas de banda passante equivalente variaram entre 10600 Kbps e 11300 Kbps, um reflexo do valor de H e da baixa variabilidade do tráfego, apresentando valores 2,5% acima da taxa média.



Figura 6.6: Estatísticas para o tráfego de Voz

Na Figura 6.7 são apresentadas as estatísticas para o tráfego de vídeo. Os valores da taxa média variaram entre 34500 Kbps e 37500 Kbps. Os valores do desvio padrão variaram entre 300 Kbps e 900 Kbps, o que indica um tráfego com grande variabilidade. As estimativas para H variaram entre 0,88 e 0,98 nos primeiros 70 segundos, depois permaneceram constantes em 0.99, o que indica a presença de tráfego não estacionário. As estimativas de banda passante equivalente variaram entre 36500 Kbps e 40500 Kbps, apresentando valores cerca de 8% maiores que os valores de sua taxa média.



Figura 6.7: Estatísticas para o tráfego de Vídeo

Na Figura 6.8 são apresentadas as estatísticas para o tráfego de dados. Os valores da taxa média variaram entre 12100 Kbps e 13000 Kbps. Os valores do desvio padrão variaram entre 1350 Kbps e 1600 Kbps, a maior variabilidade entre as três classes de serviço. As estimativas para H variaram entre 0,76 e 0,86, e apresentaram uma convergência para o valor 0,7. As estimativas de banda passante equivalente variaram entre 13000 Kbps e 15500 Kbps, apresentando valores cerca de 13% superiorer aos valores de sua taxa média.



Figura 6.8: Estatísticas para o tráfego de Dados

Segundo Cenário de Tráfego

Para a geração do tráfego "onda quadrada", configura-se o simulador para gerar dois níveis de tráfego alternados com duração de dez segundos cada. O número de fontes ativas presentes em cada nível está descrita na Tabela 6.5.

Classes de	Número de fontes	Número de fontes
Tráfego	ativas (Nível baixo)	ativas (Nível alto)
Voz	260	400
Vídeo	45	90
Dados	32	65

Tabela 6.5:

Neste cenário, o mais importante é observar os pontos nos quais ocorre o crescimento abrupto na taxa média, são eles: 10, 30, 50 e 70 segundos. Nestes pontos, pode-se observar valores de pico para o desvio padrão, sendo estes valores: 1400 Kbps para o tráfego de voz (Figura 6.9); 1300 Kbps para o tráfego de vídeo (Figura 6.10); 1650 Kbps para o tráfego de dados (Figura 6.11).

As estimativas do parâmetro H tenderam a valores fora de escala (< 0, 5 ou \geq 1), o que é esperado considerando os efeitos não estacionários presentes no tráfego, impostos pelo cenário. Os valores de H permaneceram fixados em 0,7 para o tráfego de voz (Figura 6.9) e 0,99 para o tráfego de vídeo (Figura 6.10). Porém, no caso do tráfego de dados, na Figura 6.11, as estimativas variaram entre 0,78 e 0,88 nos primeiros 50 segundos, após este período, permaneceu em 0,88.



Figura 6.9: Estatísticas para o tráfego de Voz



Figura 6.10: Estatísticas para o tráfego de Vídeo



Figura 6.11: Estatísticas para o tráfego de Dados

6.3.2 Avaliação dos Modelos de Otimização Propostos

Nesta seção, são apresentados os resultados obtidos através dos experimentos de simulação executados, considerando o modelo de otimização clássico, e os quatro modelos de otimização fuzzy.

Todos os modelos de otimização são avaliados segundo os seguintes critérios:

- Variação da taxa média de banda passante alocada ao longo da otimização;
- Variação da taxa média de perda de pacotes ao longo da otimização, para o tráfego de voz, vídeo e dados;
- Variação do retardo médio entre chegadas de pacotes ao longo da otimização, para o tráfego de voz, vídeo e dados;
- Taxa média de perda de pacotes de cada par origem-destino, para o tráfego de voz, vídeo e dados;

O primeiro cenário de tráfego é utilizado com todos os modelos de otimização. O segundo cenário de tráfego ("onda quadrada") é utilizado apenas na avaliação do modelo clássico, do terceiro modelo fuzzy e do quarto modelo fuzzy, devido ao fato de os outros modelos terem apresentado altas taxas de perdas de pacotes, tornando os seus resultados muito insatisfatórios.

Avaliação do Modelo Clássico

Na Figura 6.12, pode-se observar a variação da utilização total da rede para o modelo clássico. Seus valores variaram entre 73% e 79%, com uma média de 76%. Tal alocação permite a utilização do restante de banda para novas aplicações, aumentando a quantidade de trabalho suportada pela rede. Este valor deve-se ao fato de que, em geral, o algoritmo clássico aloca o mínimo de banda passante exigida, ou seja, o valor da capacidade de banda passante alocada por classe de serviço em cada enlace, tende a ser igual ao valor da banda passante equivalente estimada para o mesmo.



Figura 6.12: Variação da taxa de utilização da rede.

Na Figura 6.13, têm-se os valores para a taxa média de perda de pacotes e para o retardo médio entre chegadas, para o tráfego de voz, vídeo e dados. Tanto para o tráfego de voz, quanto para o de dados, praticamente não houve perdas. Para o tráfego de vídeo a taxa média de perdas variou entre 0.5% e 5.5%, com média de 2.6%, valores estes justificados pelas altas taxas de transmissão praticadas no tráfego de vídeo, e pelo valor de H, que indica uma grande variabilidade no tamanho das rajadas de pacotes.

Os valores para o retardo médio foram muito baixos para o tráfego de dados, variando entre 1 e 10 milisegundos, com média de 3 milisegundos. Para o tráfego de voz, o retardo variou entre 10 e 60 milisegundos, com média de 22 milisegundos e para o tráfego de vídeo, entre 10 e 45 milisegundos, com média de 29 milisegundos.



Figura 6.13: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados.

Na Figura 6.14, estão apresentadas as taxas médias totais por par origem-destino, segregadas pela classe de tráfego. A identificação dos pares de comunicação está descrita na Tabela 6.1. Vê-se, claramente, que os pares 2, 3, 4 e 10 são os maiores responsáveis pelas perdas de pacotes na rede. Estes pares compartilham os enlaces 2, 4 e 5 (Figura 6.1), o que gera uma carga maior para estes pares, enquanto os outros podem utilizar os enlaces restantes, menos carregados. Portanto, fica claro a incapacidade deste modelo em balancear o tráfego entre todos os pares.



Figura 6.14: Taxa média de Perda de Pacotes por par origem-destino da rede.

Na Figura 6.15, tem-se as mesmas estatísticas apresentadas anteriormente, porém agora considerando o tráfego "onda quadrada". O interessante é observar os valores nos instantes 10, 30, 50 e 70 segundos, pois são os pontos nos quais a taxa média do tráfego sofre uma elevação repentina. Nos resultados para o tráfego de voz, no instante 70 segundos, ocorreram perdas de pacotes de 20%, e o retardo chegou a atingir mais de 200 milisegundos. Para o tráfego de vídeo, no instante 50 segundos, registra-se 30% de perda de pacotes, e o retardo supera os 80 milisegundos. Considerando o tráfego de dados, o pico da taxa de perda ocorreu no instante 30 segundos, e foi de mais de 10%, com o retardo atingindo 140 milisegundos.

Nos resultados para o tráfego de voz e vídeo, o tempo necessário para o restabelecimento dos valores normais, foi de quase 10 segundos, o que indica uma dificuldade de adaptação deste modelo de otimização.



Figura 6.15: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados, para o tráfego "onda quadrada".

Na Figura 6.16, pode-se observar que para o tráfego "onda quadrada" as perdas de pacotes ocorreram de forma mais distribuída entre todos os pares, com excessão do par 8 que não apresentou nenhuma perda.



Figura 6.16: Taxa média de Perda de Pacotes por par origem-destino da rede, para o tráfego "onda quadrada".

Avaliação do Primeiro Modelo Fuzzy

Para o primeiro modelo fuzzy, os resultados apresentam apenas 60 segundos de simulação. A simulação foi interrompida devido a uma falha de energia do sistema, após 10 dias. Todavia, os resultados obtidos até então, foram suficientes para indicar que o modelo para realocação da rede é incapaz de garantir QoS, apresentando uma convergência para perdas massivas de pacotes. Portanto, não se inicializou uma nova instância de simulação.

Na Figura 6.17, pode-se observar que o primeiro modelo fuzzy aloca mais banda que o modelo clássico, com seus valores variando entre 83% e 89%, com média de 87%. Isso se deve, ao fato de haver o relaxamento do objetivo de minimização do custo, permitindo que uma quantidade maior de banda passante seja alocada.



Figura 6.17: Variação da taxa de utilização da rede.

Na Figura 6.18, vê-se a clara inferioridade do primeiro modelo fuzzy em relação ao modelo clássico. As perdas de pacotes atigem 100% para o tráfego de voz, 90% para o tráfego de dados, e variam entre 15% e 30% para o tráfego de vídeo. Os valores gerados para o retardo do tráfego de voz e dados são baixos, porém isto se deve as altas taxas de perda dessas duas classes, restando poucos pacotes para competir pelas filas dos enlaces. A variação do retardo médio para o tráfego de vídeo foi muito próxima dos valores para o modelo clássico, mas novamente, a taxa de perda de pacotes foi bastante alta.

Estes resultados, foram impulsionados pelo desequilíbrio ocorrido na alocação de fluxos por enlace. Em alguns enlaces, foram alocados vários fluxos, enquanto em outros, não houve a alocação de qualquer fluxo. Devido ao fato de a largura de banda ser menor para o tráfego de voz e dados, estas classes foram as mais prejudicadas, ocorrendo uma acúmulo maior de fluxos por enlace.



Figura 6.18: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados.

Na Figura 6.19, fica claro que todos os pares de comunicação contribuem com a alta taxa de perda de pacotes. Salvo para o tráfego de video, cujas taxas de perdas mais expressivas concentram-se nos pares 4, 7, 9 e 10.



Figura 6.19: Taxa média de Perda de Pacotes por par origem-destino da rede.

Avaliação do Segundo Modelo Fuzzy

Na Figura 6.20, observa-se o resultado da adição da restrição para o aumento da capacidade alocada neste modelo de otimização. A taxa de alocação varia entre 89% e 99%, com média de 94%. Com este tipo de alocação, a capacidade física total da rede é praticamente toda utilizada.



Figura 6.20: Variação da taxa de utilização da rede.

Apesar da alta taxa de utilização da rede, os resultados foram muito parecidos com os do primeiro modelo fuzzy. Novamente, ocorreram problemas no balanceamento dos fluxos pelos enlaces, acumulando vários fluxos em apenas poucos enlaces, enquanto outros permaneciam ociosos. Pode-se ver na Figura 6.21, que as taxas média de perda para o tráfego de voz e dados chegaram a 100%, o que invalida os baixos valores obtidos para o retardo médio dessa classes de tráfego. Para o tráfego de vídeo, as taxas de perda variaram entre 0 e 12%, com média de 2,7%. O retardo médio entre chegadas variou entre 5 milisegundos e 40 milisegundos, com uma média de 18 milisegundos, valores estes muito bons, comparados aos resultados obtidos com o modelo clássico. Considerando-se apenas o tráfego de vídeo, o segundo modelo fuzzy apresenta taxas de perda e de retardo inferiores as do modelo clássico.



Figura 6.21: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados.

Na Figura 6.22, observa-se um equilíbrio entre as taxas médias de perdas para todos os

pares de comunicação, considerando as três classes de tráfego. No entanto, considerando os resultados para o tráfego de voz e dados, em que ocorreram taxas de perdas massivas de pacotes, este modelo não oferece garantias mínimas de QoS, o que o torna uma opção inviável.



Figura 6.22: Taxa média de Perda de Pacotes por par origem-destino da rede.

Avaliação do Terceiro Modelo Fuzzy

Na Figura 6.23, pode-se observar que o impacto da nova restrição, que minimiza o número de fluxos de cada classe por enlace, acarretou na diminuição da taxa média de alocação da rede. A taxa variou entre 78% e 95%, com média de 89%, valor este situado entre as taxas do primeiro e do segundo modelo fuzzy.



Figura 6.23: Variação da taxa de utilização da rede.

Na Figura 6.24, as estatísticas para o tráfego de voz e dados, indicam dois períodos de resultados distintos para as taxas médias de perdas de pacotes. O primeiro, compreendido entre 0 e 70 segundos apresenta taxas de perda de até 20%, e o segundo com taxas atingindo 40% de perda. Para o tráfego de vídeo obteve-se um resultado muito satisfatório, pois a taxa de perda variou entre 0 e 7%, com média de 1,7%, e o retardo médio variou entre 3 milisegundos e 40 milisegundos, com média de 16 milisegundos. Os resultados para o tráfego de vídeo, superam os resultados do modelo clássico, pois apresentam valores para a taxa média de perda e para o retardo médio quase 50% menores que os do modelo clássico.


Figura 6.24: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados.

Na Figura 6.25, pode-se observar o grande desequilíbrio da taxa de perda para o tráfego de voz entre os pares de comunicação. Ocorreram grandes taxas de perda, apenas para os pares 1, 6, 7 e 10, atingindo o valor de 90% para o par 7. Para o tráfego de dados a intensidade maior de perdas ocorreu nos pares 4, 6 e 7. Somente o par de comunicação 2 não apresentou nenhuma perda de pacotes.

A adição de restrições para o balanceamento do tráfego pelos enlaces da rede resolveu, em parte, o problema do acúmulo de fluxos ocorrido nos modelos fuzzy anteriores. Todavia, este modelo não foi capaz de impedir o acúmulo de fluxos nos enlaces 4 e 5 da rede (Figura 6.1), o que acabou gerando altas taxas de perda nos pares de comunicação 4, 6 e 7.



Figura 6.25: Taxa média de Perda de Pacotes por par origem-destino da rede.

Na Figura 6.26, estão apresentadas as estatísticas para o tráfego "onda quadrada". Considerando as classes de voz e dados, é observada a tendência do terceiro modelo fuzzy em acumular um número maior de perdas, relação ao modelo clássico. A taxa de perda supera os 50% para o tráfego de voz no instante 75 segundos, e atinge 20% nos instantes 35, 55 e 75 segundos para o tráfego de dados. Os resultados para a variação do retardo médio, considerando o tráfego de voz e dados, foram similares para ambos os modelos de otimização.

Para o tráfego de vídeo, porém, os resultados foram animadores, apresentando taxas de perda de pacotes até 67% menores (aos 12 segundos) do que as apresentadas pelo modelo clássico. Os valores para o retardo médio variaram entre 4 milisegundos e 45 milisegundos, com média de 16 milisegundos, um valor quase 50% inferior a média de 29 milisegundos do modelo clássico.



Figura 6.26: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados, para o tráfego "onda quadrada".

Na Figura 6.27, observa-se novamente o desequilíbrio presente entre as taxas médias de perda dos tráfegos de voz e dados, para os pares de comunicação. Com especial atenção para o par 7 que apresentou perdas massivas em ambas as classes de tráfego.



Figura 6.27: Taxa média de Perda de Pacotes por par origem-destino da rede, para o tráfego "onda quadrada".

Avaliação do Quarto Modelo Fuzzy

Na Figura 6.28, pode-se ver a variação da taxa de alocação para o quarto modelo de otimização fuzzy. A taxa, variou entre 80% e 88%, com média de 85%, sendo este valor o mais baixo entre os algoritmos fuzzy avaliados. Devido a relaxação da função objetivo, o modelo tendeu a alocar menos banda passante, o que permite um melhor aproveitamento da capacidade física da rede. Todavia, este valor ainda é superior aos 76% de alocação média apresentados pelo modelo clássico.



Figura 6.28: Variação da taxa de utilização da rede.

Na Figura 6.29, observa-se que praticamente não ocorreram perdas de pacotes nas classes de voz e dados. Na classe de vídeo, houve uma queda de 50% na quantidade de pacotes perdidos, em relação ao modelo clássico, apresentando uma média de 1,2% de perdas. Os resultados para a variação do retardo médio também foram superiores. A taxa média para o retardo no tráfego de voz foi de 17 milisegundos, contra 22 milisegundos do modelo clássico. Para o tráfego de vídeo a média foi de 19 milisegundos, contra 29 milisegundos. A média do retardo permaneceu igual para o tráfego de dados, que foi de 3 milisegundos.



Figura 6.29: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados.

Observa-se, na Figura 6.30, que as as taxas médias de perda de pacotes do tráfego de vídeo entre todos os pares de comunicação, são equivalentes. Estes resultados, são superiores aos do modelo clássico pois, salvo no caso do par 10, as perdas ocorridas na rede são melhor distribuídas entre todos os fluxos origem-destino.

O quarto modelo de otimização fuzzy, oferece bons níveis de QoS para o tráfego servido, apresentando os mais baixo índices de perda dentre os modelos avaliados. A relaxação imposta ao objetivo do problema, permitiu um melhor desempenho na minimização do número de fluxos por enlace da rede. Com isto, apesar de o modelo garantir uma taxa de utilização menor, a taxa média de perdas também caiu.



Figura 6.30: Taxa média de Perda de Pacotes por par origem-destino da rede.

Na Figura 6.31, observa-se uma equivalência entre os resultados da taxa média de perda para os dois modelos de otimização. No entanto, os resultados do quarto modelo fuzzy foram melhores nos pontos de pico, com taxas de perda inferiores em até 20% para o tráfego de voz, 22% para o tráfego de vídeo e 10% para o tráfego de dados.

Os resultados para a variação do retardo médio entre pacotes também se assemelharam aos resultados do modelo clássico. No entanto, os resultados do quarto modelo fuzzy, em geral, apresentaram valores para o retardo médio, 20% a 25% inferiores aos do modelo clássico.

Esses resultados, demonstram a capacidade do quarto modelo fuzzy em se adaptar as variações presentes no tráfego. Como este modelo tende a alocar mais banda passante por fluxo, ele expressa uma taxa de perdas menor do que o modelo clássico. Apesar de os três primeiros modelos fuzzy alocarem mais banda passante do que o quarto modelo, eles não garantem o balaceamento dos fluxos entre os enlaces, por isso sofrem taxas de perda maiores.



Figura 6.31: Variação da Taxa média de Perda de Pacotes e Retardo Médio para o tráfego de Voz, Vídeo e Dados, para o tráfego "onda quadrada".

Na Figura 6.32, pode-se atestar a característica do quarto modelo de otimização fuzzy, de distribuir as perdas entre todos os pares de comunicação, demonstrando sua capacidade em balancear o tráfego por toda a rede.



Figura 6.32: Taxa média de Perda de Pacotes por par origem-destino da rede, para o tráfego "onda quadrada".

6.3.3 Resumo dos Resultados Obtidos

Na Tabela 6.6, estão descritos os valores médios dos resultados obtidos com os experimentos de simulação, para o primeiro cenário de tráfego.

Modelo	Taxa Média	Taxa Média	Retardo
	de Utilização	de Perda	Médio
Clássico	76%		
Voz		0,01%	22,8 ms
Vídeo		2,6%	$29{,}4~\mathrm{ms}$
Dados		0%	$3,0 \mathrm{~ms}$
Fuzzy 1	87%		
Voz		89,3%	4,6 ms
Vídeo		$13,\!2\%$	32,5 ms
Dados		56,4%	$1,7 \mathrm{~ms}$
Fuzzy 2	94%		
Voz		83,0%	$1,5 \mathrm{ms}$
Vídeo		2,7%	$18{,}5~\mathrm{ms}$
Dados		60,1%	2,0 ms
Fuzzy 3	89%		
Voz		24,2%	$15,8 \mathrm{ms}$
Vídeo		1,7%	$16{,}2~\mathrm{ms}$
Dados		18,6%	$2,3 \mathrm{~ms}$
Fuzzy 4	85%		
Voz		0,01%	$17,8 \mathrm{ms}$
Vídeo		$1,\!2\%$	$19,8 \mathrm{\ ms}$
Dados		0%	2,7 ms

Tabela 6.6: Taxas médias de utilização, perda e retardo.

O modelo de otimização clássico, possui a característica de oferecer QoS para todas as classes de tráfego. Sua taxa de alocação da rede foi a mais baixa dentre todos os modelos (76%), o que o torna o algoritmo que permite uma melhor utilização da rede. A baixa utilização ocorre, devido ao fato de este modelo minimizar o custo total da rede. Nos experimentos de simulação, ocorreram perdas significativas apenas com o tráfego de vídeo (2,6%), já os valores para o retardo médio de chegada foram um pouco elevados, em relação aos modelos fuzzy. Isso se deve ao fato de alocação de banda passante, por fluxo, ser maior nos modelos fuzzy.

O primeiro modelo fuzzy, busca relaxar o objetivo do modelo clássico, para que se possa alocar mais banda passante por fluxo. O segundo modelo fuzzy acrescenta restrições para forçar o aumento dessa alocação. Todavia, os resultados dos experimentos de simulação para o tráfego de voz e dados, do primeiro e segundo modelo fuzzy, invalidam a aplicação destes algoritmos. Ambos convergiram para perdas massivas de pacotes de voz e dados, o que torna os valores para o retardo médio, presentes na Tabela 6.6, irrelevantes. O segundo modelo fuzzy apresentou bons resultados para o tráfego de vídeo, porém foi incapaz de garantir condições mínimas de QoS para as outras classes de tráfego. Ambos os modelos, concentraram o tráfego em poucos enlaces, causando as altas taxas de perdas vistas para o tráfego de voz e dados. Por demandar mais banda passante, os fluxos de vídeo foram privilegiados no arranjo geral da rede, não ocorrendo casos em que mais de dois fluxos eram alocados para um mesmo enlace.

O terceiro modelo fuzzy, busca resolver o problema de balanceamento de tráfego existente nos modelos fuzzy anteriores. De fato, isto ocorre, porém, o problema não é resolvido por completo, acarretanto na ocorrência de perdas massivas em dois enlaces. Todavia, o terceiro modelo propicia uma melhoria geral entre todas as classes de tráfego, obtendo valores de perdas bastante inferiores para o tráfego de voz e dados (aproximadamente 70%), e uma melhoria de 50% para o tráfego de vídeo. Os valores para o retardo médio, foram os mais baixo dentre todos os modelos, ficando muito abaixo dos valores apresentados pelo modelo clássico.

O quarto modelo de otimização fuzzy é equivalente ao terceiro, porém introduz uma relaxação no cálculo da função objetivo, permitindo que os conjuntos de restrições (adicionados pelos outros modelos) possam ser maximizadas ou minimizadas independentemente. Esta relaxação do objetivo, implicou em resultados que superam todos os outros modelos avaliados. Sua taxa de utilização média da rede (85%) foi a mais baixa entre os modelos fuzzy, porém superior a taxa do modelo clássico (76%). Seus valores para a taxa média de perda igualaram com os resultados do modelo clássico para o tráfego de voz e dados (quase nulos), e foram inferiores, em mais de 50%, para o tráfego de vídeo. Os valores para o retardo médio ficaram abaixo de todos os modelos, com exceção do terceiro modelo fuzzy. A relaxação do objetivo, no quarto modelo, possibilitou uma ação mais eficaz no balanceamento do tráfego pela rede, o que implicou na diminuição da quantidade média de banda passante alocada por fluxo. No entanto, por alocar mais banda passante do que o modelo clássico, pode-se obter valores menores de retardo para o tráfego de voz, vídeo e dados.

Considerando as estatísticas apresentadas na Tabela 6.7, fica evidenciada a superioridade do quarto modelo fuzzy em relação ao modelo clássico. Além de obter taxas de perda inferiores e retardos menores, o desvio padrão entre os pares foi sempre menor ou igual aos do modelo clássico, indicando um melhor equilíbrio entre as estatísticas. Analisando o 90° percentil, o quarto modelo fuzzy é superior ao modelo *clássico*, com destaque para as estatísticas do tráfego de vídeo. Pode-se observar que 90% dos valores da taxa de perda ficaram abaixo de 1,7%, e no caso do retardo, abaixo 31,8 milisegundos. Resultados estes, muito superiores aos obtidos com o modelo clássico.

	Desvio Padrão	Desvio Padrão	90^{o} Percentil	90^{o} Percentil
Modelo	Taxa de Perda	Retardo	Taxa de Perda	Retardo
Clássico				
Voz	$0,\!01\%$	13,42 ms	$0,\!02\%$	32,5 ms
Vídeo	$2,\!69\%$	21,41 ms	$5{,}2\%$	$55,9 \mathrm{\ ms}$
Dados	0%	$1{,}84~\mathrm{ms}$	0%	$4,3 \mathrm{~ms}$
Fuzzy 4				
Voz	$0,\!01\%$	8,19 ms	0,02%	$26,5 \mathrm{ms}$
Vídeo	$0,\!87\%$	$10{,}09~\mathrm{ms}$	1,7%	$31,8 \mathrm{\ ms}$
Dados	0%	$1{,}16~{\rm ms}$	0%	$4,0 \mathrm{~ms}$

Tabela 6.7: Desvio padrão e 90^o percentil entre os valores da taxa média de perda e de retardo dos pares origem-destino.

Capítulo 7 Conclusões

Nesta dissertação, apresentou-se o *framework* de Redes Auto-Ajustáveis, como um conjunto de técnicas capazes de prover Qualidade de Serviço por fluxo, em redes multiserviço de alta velocidade.

Nos dias de hoje, as redes de comunicação de núcleo que implementam a tecnologia ATM, são as mais indicadas a implantação deste *framework*. As organizações (privadas ou públicas) responsáveis por estas redes de comunicação, em geral, não utilizam qualquer método dinâmico para o ajuste de suas redes frente as variações do tráfego servido. Os ótimos índices de QoS atingidos por estas redes, dependem hoje, da baixa utilização de seus enlaces. Todavia, com a evolução das tecnologias utilizadas nas redes de acesso, o número de aplicações multimídia deve crescer exponencialmente nos próximos anos, aumentando a quantidade de banda passante requerida por cada usuário, e possibilitando a introdução de novas aplicações de banda larga, como vídeo sob demanda.

Tendo em vista o horizonte das novas tecnologias de acesso e das novas aplicações multimídia, as redes de núcleo do futuro deverão ser capazes de efetuar um uso mais racional de seus recursos, ajustando-se às necessidades temporais impostas pelos seus usuários. Fica evidenciado, então, a importância de técnicas de gerência de redes capazes de se adaptar em tempo-real as flutuações do tráfego servido.

O framework de Redes Auto-Ajustáveis inclui quatro etapas principais:

- 1. Coleta passiva das estatísticas dos fluxos da rede: que deve ser implementada por mecanismos de hardware dedicados, capazes de lidar com as altas taxas de transmissão existentes nos enlaces;
- 2. Cálculo da Banda Passante Equivalente: que deve levar em consideração as estatísticas dos fluxos e as características da rede, de forma a prover uma estimativa correta o suficiente, para que não haja desperdício de banda passante, nem o comprometimento dos requisitos de QoS;

- Solução do Problema de Particionamento Ótimo de Banda: que deve definir uma boa solução para o roteamento dos caminhos pela rede, e para a realocação de banda passante para todas as bandas virtuais, em todos os enlaces;
- 4. Utilização de uma técnica de controle para a configuração da rede: que deve utilizar uma técnica eficiente para transmitir a nova configuração para todos os nós da rede.

Neste trabalho, foram exploradas a segunda e a terceira etapa do *framework*. Alguns estimadores de banda passante equivalente, existentes na literatura, foram avaliados quanto a sua capacidade de garantir QoS aos fluxos de informação, e evitar a superalocação de banda passante. O estimador baseado no processo envelope do Movimento Browniano Fractal (FEP) demonstrou possuir grande robustez frente aos diferentes cenários de tráfego experimentados, apresentando ótimos resultados com o tráfego de voz, vídeo e dados. Por ser um estimador baseado em medidas das estatísticas do tráfego, o estimador FEP é capaz de calcular a banda passante equivalente para um fluxo, em tempo-real, o que o torna uma ótima ferramenta para a utilização no *framework* de redes auto-ajustáveis.

O estimador FEP foi então utilizado no estudo da terceira etapa do *framework*, na qual foram apresentados modelos matemáticos para a solução do Problema de Particionamento Ótimo de Banda. Considerando a imprecisão, inerente aos estimadores de banda passante, e a imprevisibilidade do trádego em geral, utilizaram-se técnicas fuzzy na construção de quatro modelos de otimização. Com estes modelos, foi possível relaxar a função objetivo, e acrescentar novas restrições fuzzy permitindo que novos objetivos fossem alcançados. As principais características dos modelos fuzzy implementados são:

- minimização do custo total na rede;
- maximização da quantidade de banda passante alocada, por fluxo, na rede;
- minimização do número de fluxos alocados, por enlace, na rede.

Estes modelos foram avaliados em um cenário simulado com condições reais de uma rede multisserviço de alta velocidade. O terceiro e o quarto modelos fuzzy propostos, apresentaram resultados animadores, superando, em muitos pontos, os índices de QoS prestados pelo modelo de otimização padrão, existente na literatura.

Os resultados obtidos, indicaram que a utilização de uma abordagem fuzzy, na solução do problema de particionamento de banda, pode trazer vantagens para o *framework* de Redes Auto-Ajustáveis.

Em resumo, esta dissertação apresenta como contribuição, o estudo dos estimadores de banda passante equivalente FEP e GA frente a diversos cenários de tráfego, sintético e real, para fluxos de voz, vídeo e dados. Apresenta, também, a configuração de um ambiente de simulação para Redes Auto-Ajustáveis, no qual se pode avaliar o comportamento de diversos modelos de otimização para o problema do Particionamento Ótimo de Banda. A terceira contribuição foi a introdução de modelos de otimização fuzzy, capazes de oferecer QoS para o tráfego servido, apresentando uma solução mais eficiente do que a do modelo de otimização presente na literatura.

7.1 Propostas de Trabalhos Futuros

Pode-se considerar esta dissertação, como uma introdução ao *framework* de Redes Auto-Ajustáveis, que estuda as principais etapas para a sua implementação. Como trabalho futuro, um estudo mais completo de todas as etapas é proposto. Para tal, faz-se necessário o aprofundamento dos estudos em várias áreas, que contemplem as seguintes questões:

- Estudo de técnicas de tempo-real, para a coleta de estatísticas dos fluxos na rede;
- Estudo de outros estimadores de banda passante equivalente presentes na literatura, buscando novas opções para o uso com diferentes classes de tráfego.
- Implementação de novos estimadores baseados em medidas, específicos para cada classe de tráfego;
- Adaptação de estimadores do parâmetro de *Hurst* para o *framework*, possibilitando que a estimativa de *H* ocorra paralelamente ao cálculo das outras estatísticas;
- Estudo de algorítmos aproximados para a solução do problema de particionamento ótimo de banda em tempo-real;
- Estudo de metodologias de controle distribuídas para a implementação em redes de grande porte.

Referências Bibliográficas

- P. Abry and D. Veitch. Wavelet analysis of long-range-dependent traffic. *IEEE Transactions on Information Theory*, 44(1):2–15, 1998.
- [2] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. Network Flows, theory, algorithms, and applications. Prentice-Hall, New Jersey, 1993.
- [3] A.Land and A. Doig. An automatic method of solving discrete programming problems. *Econometrica*, 28(3):497–520, 1960.
- [4] C. Barnhart, C. A. Hane, and P. H. Vance. Integer multicommodity flow problems. V Conference Integer Programming and Combinatorial Optimization, 1996.
- [5] Mokhtar S. Bazaraa, John J. Jarvis, and Hanif D. Sherali. *Linear Programming and Network Flows*. John Wiley, 2nd edition, 1990.
- [6] R. Bellman and L. A. Zadeh. Decision making in a fuzzy environment. *Management Sciences*, 17:141–164, 1970.
- [7] Lee Breslau, Sugih Jamin, and Scott Shenker. Comments on the performance of measurement-based admission control algorithms. In *INFOCOM (3)*, pages 1233– 1242, 2000.
- [8] N. L. S. Fonseca, G. S. Mayor, and C. A. V. Neto. On the equivalent bandwidth of self-similar sources. ACM Transactions on Modeling and Computer Simulation, 10(3):104–124, april 2000.
- [9] National Laboratory for Applied Network Research. http://pma.nlanr.net/pma/, 2004.
- [10] PMA Hardware National Laboratory for Applied Network Research. http://pma.nlanr.net/hardware/, 2004.
- [11] F.P.Kelly, S.Zachary, and I.B. Ziedins, editors. Notes on Effective Bandwidth, pages 141–168. Oxford University Press, 1996.

- [12] M. R. Garey and D. S. Johnson. Computers and Intractability, A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, New York, 22 edition, 2000.
- [13] Robert Garfinkel and George L. Nemhauser. Integer Programming. John Wiley & Sons, Canada, 1972.
- [14] F. Glover and M. Laguna. Tabu Search. Kluwer Academic Publishers, Boston, 1997.
- [15] F. Glover, J. Xu, and S.Y. Chiu. Optimizing a ring-based private line telecommunication network using tabu search. *Management Science*, 45(3):330–345, 1999.
- [16] F. Haciomeroglu and M. Devetsikiotis. A dynamic time scale approach for on-line measurement-based capacity allocation. *Brazilian Journal on Telecommunications*, 18(2), 2003.
- [17] Q. Hao, S. Tartarelli, and M. Devetsikiotis. Self-sizing and optimization of high-speed multiservice networks. In Proc. of IEEE Globecom '00, 3:1818–1823, 2000.
- [18] F. Herrera, M. Lozano, and J. L. Verdegay. Applying genetic algorithms in fuzzy optimization problems. *Fuzzy Sets & Artificial Intelligence*, 3(1):39–52, 1994.
- [19] J. H. Holland. Adaptation in Natural and Artificial Systems. The University of Michigan Press, 1975.
- [20] S. Jamin and S. J. Shenker. Measurement-based admission control algorithms for controlled-load service: A structural examination. Technical Report CSE-TR-333-97, 1 1997.
- [21] Sugih Jamin, Scott Shenker, and Peter B. Danzig. Comparison of measurement-based call admission control algorithms for controlled-load service. In *Proc. of INFOCOM* (3), pages 973–980, 1997.
- [22] A. P. Castro Jr. Alocação de recursos em redes programáveis. Master's thesis, Intituto de Computação, UNICAMP, Campinas, SP, 2001.
- [23] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–681, 1983.
- [24] G. J. Klir, U. H. St.Clair, and B. Yuan. Fuzzy Set Theory Foundations and Applications. Prentice Hall, New Jersey, 1997.
- [25] O. Lamparter and B. Stauffer. Akqui: a network traffic measurement tool. In Proc. of 10th International Conference on Telecommunications, 2:1078–1083, 2003.

- [26] S. Molnár and T. Dinh Dang. Pitfalls in long range dependence testing and estimation. In Proc. of Global Telecommunications Conference, 2000. GLOBECOM '00. IEEE, 1:662–666, 2000.
- [27] MPEG-4 and H.263 Video Traces for Network Performance Evaluation. http://wwwtkn.ee.tu-berlin.de/research/trace/trace.html, 2004.
- [28] S. Nakagawa, Y. Kato, S. Nakai, K. Ogura, and H. Saito. Self-sizing network operation systems in atm networks. In Proc. of Network Operations and Management Symposium, 1996., IEEE, 2:563–572, 1996.
- [29] S. Nalatwad and M. Devetsikiotis. Self-sizing networks: local vs. global control. In Proc. of IEEE International Conference on Communications, 4:2163–2167, 2004.
- [30] J. A. Nelder and R. Mead. A simplex method for function minimization. Computer Journal, 7:308–313, 1965.
- [31] I. Norros. On the use of fractional brownian motion in the theory of connectionless networks. *IEEE Journal of Selected Areas in Communications*, 13(6):953–962, 1995.
- [32] Harvey M. Salkin, Kamlesh Marthur, and Robert Haas. Foundations of Integer Programming. Elsevier Science Publishers, Amsterdan, Netherlands, 1989.
- [33] Mischa Schwartz. Broadband Integrated Networks. Prentice-Hall, New Jersey, 1996.
- [34] Network Simulator. http://www.isi.edu/nsnam/ns/, 2005.
- [35] W. Stallings. High-Speed Networks, TCP/IP and ATM Design Principles. Prentice-Hall, New Jersey, 1 edition, 1998.
- [36] Xpress-MP Suite. http://www.dashoptimization.com/, 2004.
- [37] Cisco Systems. http://www.cisco.com/en/us/products/hw/switches/ ps708/products_white_paper09186a0080131086.shtml, 2004.
- [38] D. Veitch and P. Abry. A wavelet based joint estimator of the parameters of longrange dependence. *IEEE Transactions on Information Theory special issue on Multiscale Statistical Signal Analysis and its Applications*, 45(3):878–897, 1999.
- [39] D. Veitch, P. Abry, and M. S. Taqqu. On the automatic selection of the onset of scaling. *Fractals*, 11(4):377–390, 2003.

- [40] Walter Willinger, Murad S. Taqqu, Robert Sherman, and Daniel V. Wilson. Selfsimilarity through high-variability: statistical analysis of ethernet lan traffic at the source level. *IEEE/ACM Transactions on Networking*, 5(1):71–86, 1997.
- [41] J. Yan. Adaptive configuration of elastic high-speed multiclass neworks. IEEE Communications Magazine, pages 116–120, May 1998.
- [42] M. Zangrilli and B. B. Lowekamp. Comparing passive network monitoring of grid application traffic with active probes. In Proc. of Fourth International Workshop on Grid Computing, pages 84–91, 2003.
- [43] H.-J. Zimmermann. Description and optimization of fuzzy systems. International Journal of General Systems, 2:209–215, 1976.