

**Universidade Estadual de Campinas**

FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

DEPARTAMENTO DE TELEMÁTICA



# Contribuição ao Encaminhamento de Comprimentos de Onda em Redes Ópticas WDM

Tese apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

por

**Luis Guillermo Milla-León Genie**

Engenheiro Elétrico - Universidad Ricardo Palma

Lima, Perú

Orientador: **Prof. Dr. Ivanil Sebastião Bonatti** FEEC/UNICAMP

## **Banca Examinadora**

**Prof. Dr. Ivanil Sebastião Bonatti**

FEEC/UNICAMP (presidente)

**Prof. Dra. Eunice Luvizotto Medina Pissolato**

CPqD

**Prof. Dr. Akebo Yamakami**

FEEC/UNICAMP

**Prof. Dr. Edson Moschim**

FEEC/UNICAMP

FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DA ÁREA DE ENGENHARIA - BAE - UNICAMP

M611c Milla-Leon Genie, Luis Guillermo  
Contribuição ao encaminhamento de comprimentos  
de onda em redes ópticas WDM / Luis Guillermo  
Milla-Leon Genie.–Campinas,SP:[s.n.], 2003.

Orientador: Ivanil Sebastião Bonatti  
Dissertação (Mestrado) - Universidade Estadual de  
Campinas, Faculdade de Engenharia Elétrica e de  
Computação

1. Redes de computação - Protocolos. 2. Redes de  
Informação. I. Bonatti, Ivanil Sebastião. II.  
Universidade Estadual de Campinas. Faculdade de  
Engenharia Elétrica e de Computação. III. Título

# Resumo

O encaminhamento de conexões WDM nas redes ópticas transparentes é abordado nesta tese. A rede é modelada por um grafo no qual os nós representam os equipamentos de comutação óptica, com capacidade plena de conversão de comprimento de onda, e os ramos representam as fibras ópticas. O problema de estabelecimento de conexões ponto-a-ponto, com demanda conhecida *a priori* (caso estático) e com o objetivo de minimizar o número médio de pedidos de conexões não atendidos, foi formulado como um problema multifluxo em redes resultando em um problema de programação matemática linear inteira. Uma heurística, baseada em técnicas de caminho mínimo, foi proposta e apresentou resultados satisfatórios quando comparada com a solução ótima do problema multifluxo. Uma variante do algoritmo proposto foi aplicada ao caso dinâmico (conhecimento probabilístico das demandas) e avaliada através de simulação de eventos discretos. As estimativas das médias e dos intervalos de confiança foram obtidas com o método *bootstrap*. A técnica proposta apresentou desempenho satisfatório quando comparada com a técnica Max-Sum que se destaca na literatura especializada.

**Palavras-Chave:** Redes ópticas; Redes WDM; Encaminhamento de comprimentos de onda; Caminho Mínimo.

# Abstract

Connection routing in all-optical WDM networks is examined in this thesis. The network is modeled by a graph in which nodes represent the optical switching equipment, with full wavelength conversion capacity, and the links represent the optical fibers. The problem of establishing point-to-point connections with a given in advance demand (static case) and with the objective of minimizing the mean number of blocked connections was formulated as a multi-commodity flow problem resulting in an integer linear programming problem. A heuristic based in the shortest path technique was proposed and presented satisfactory results when compared with the optimal solution of the multi-commodity flow solution. A variation of the proposed algorithm was applied to the dynamic case (probabilistic knowledge of demands) and evaluated through discrete event simulation. Mean and confidence intervals estimates were obtained with the bootstrap method. The proposed technique presented a satisfactory performance when compared to the Max-Sum technique, which highlights in the specialized literature.

**Key-words:** Optical networks; WDM networks; Wavelength routing; Shortest path.

# Agradecimentos

- A Deus, pela oportunidade de aprendizado.
- A meu orientador, Prof. Dr. Ivanil S. Bonatti, por todo o que aprendi, pela sua dedicação e amizade.
- Ao Prof. Akebo e a Marcia, pela programação do modelo ILP.
- A meus amigos do Departamento de Telemática, em especial a Ricardo, Vinícius, Valter, José Maria, Yusef, Fabiano, Rubia, Tatiane, Edilson, Marcia e Wanessa.
- A meus pais pelo amor e apoio incondicional.

*O homem é sua própria estrela; e a alma que pode  
Fazer um homem honesto e perfeito,  
Comanda toda luz, toda influência, todo destino;  
Nada lhe advém cedo ou tarde demais.  
Nossos atos são nossos anjos, bons ou maus,  
Nossas sombras fatais que andam ao nosso lado, silentes.*

*Beaumont e Fletcher*

# Conteúdo

<b>Resumo e Abstract</b>	<b>ii</b>
<b>Agradecimentos</b>	<b>iii</b>
<b>Lista de Figuras</b>	<b>vii</b>
<b>Lista de Tabelas</b>	<b>viii</b>
<b>Acrônimos</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Considerações Gerais . . . . .	1
1.2 Multiplexação por Divisão no Comprimento de Onda. . . . .	2
1.3 Organização da Dissertação . . . . .	3
<b>2 Características e Elementos da Arquitetura de Redes WDM</b>	<b>4</b>
2.1 Características da Arquitetura de Redes WDM . . . . .	4
2.1.1 Reutilização de Comprimento de Onda . . . . .	4
2.1.2 Conversão de Comprimento de Onda . . . . .	4
2.1.3 Transparência . . . . .	5
2.1.4 Comutação de Circuitos . . . . .	5
2.1.5 Confiabilidade . . . . .	5
2.2 Elementos da Arquitetura de Redes WDM . . . . .	5
2.2.1 Terminais Ópticos de Linha . . . . .	6
2.2.2 Amplificadores Ópticos de Linha . . . . .	7
2.2.3 Multiplexadores Ópticos de Inserção e Retirada de Canais . . . . .	8
2.2.4 Comutadores Ópticos . . . . .	10
<b>3 Características das Redes Encaminhadas por Comprimento de Onda</b>	<b>12</b>
3.1 Considerações Gerais . . . . .	12
3.2 Encaminhamento e Atribuição de Comprimentos de Onda . . . . .	12
3.3 Métodos de Seleção de Caminho e de Comprimento de Onda . . . . .	13
3.3.1 Métodos de Seleção de Caminho . . . . .	14
3.3.2 Heurísticas de Atribuição de Comprimento de Onda. . . . .	15
3.4 Restrição de Continuidade do Comprimento de Onda. . . . .	17

---

3.5	Conversão de Comprimento de Onda. . . . .	17
3.5.1	Técnicas de Conversão de Comprimento de Onda. . . . .	19
3.6	Redes Multi-Fibra . . . . .	21
<b>4</b>	<b>Encaminhamento de Comprimentos de Onda em Redes WDM</b>	<b>22</b>
4.1	Introdução . . . . .	22
4.2	Classificação . . . . .	22
4.3	Caminho Mínimo . . . . .	24
4.3.1	Formulação do Modelo Dijkstra . . . . .	24
4.4	Encaminhamento Estático . . . . .	28
4.4.1	Formulação do Modelo de Caminho Mínimo . . . . .	29
4.4.2	Formulação do Modelo de Programação Linear Inteira . . . . .	30
4.4.3	Resultados Numéricos . . . . .	31
4.5	Encaminhamento Dinâmico . . . . .	35
4.5.1	Formulação do Modelo de Caminho Mínimo . . . . .	36
4.5.2	Formulação do Modelo Max-Sum. . . . .	37
4.5.3	Validação . . . . .	44
4.5.4	Resultados Numéricos . . . . .	50
<b>5</b>	<b>Conclusões e Perspectivas</b>	<b>52</b>
	<b>Apêndice A</b>	<b>54</b>
	<b>Apêndice B</b>	<b>56</b>
	<b>Bibliografia</b>	<b>86</b>

# Lista de Figuras

2.1	Diagrama em blocos de uma rede WDM típica. . . . .	6
2.2	Diagrama em blocos de um OLT. . . . .	7
2.3	Diagrama em blocos de um amplificador óptico de linha. . . . .	8
2.4	Arquitetura paralela do OADM. . . . .	9
2.5	Arquitetura modular paralela OADM. . . . .	9
2.6	Arquitetura serial OADM. . . . .	10
2.7	Arquitetura de retirada de banda OADM. . . . .	10
3.1	Exemplo de encaminhamento e alocação de comprimentos de onda para os casos (a) dinâmico e (b) estático. . . . .	13
3.2	Equivalência entre (a) uma rede multi-fibras e (b) uma rede mono-fibra com conversão de comprimento de onda. . . . .	21
4.1	Rede com 7 nós e 11 enlaces. . . . .	26
4.2	Nó 2 incluído na árvore mínima. . . . .	26
4.3	Nó 5 incluído na árvore mínima. . . . .	27
4.4	Nó 3 incluído na árvore mínima. . . . .	27
4.5	Nó 4 incluído na árvore mínima. . . . .	27
4.6	Nó 6 incluído na árvore mínima. . . . .	28
4.7	Solução de árvore mínima. . . . .	28
4.8	Rede de Seis nós com capacidades rotuladas nos enlaces. . . . .	32
4.9	Rede COST239EON. . . . .	33
4.10	Rede NSFNet. . . . .	34
4.11	Rede de 6 nós. . . . .	39
4.12	Capacidade residuais dos enlaces no estado $\psi_1$ . . . . .	40
4.13	Capacidades residuais dos enlaces no estado $\psi_2$ . . . . .	40
4.14	Capacidades residuais dos enlaces no estado $\psi_3$ . . . . .	41
4.15	Capacidades residuais dos enlaces no estado $\psi_4$ . . . . .	41
4.16	Cadeia de Markov de um enlace com $n$ circuitos. . . . .	45
4.17	Valores (pontos) e dados amostrais (asteriscos) gerados. . . . .	48
4.18	Intervalos de confiança para $t_f = 4.000$ (esquerda) e $t_f = 1.000$ (direita). . . . .	49

# Lista de Tabelas

4.1	Número de pedidos de conexão para a rede de seis nós. . . . .	32
4.2	Número de conexões atendidas e bloqueadas para a rede de seis nós. . . . .	32
4.3	Número de pedidos de conexão entre nós para a rede COST239EON. . . . .	33
4.4	Pedidos de conexão bloqueados para a rede COST239EON como função do número $n$ de comprimentos de onda por enlace. . . . .	34
4.5	Tempos computacionais (em segundos) para a rede COST239EON. . . . .	34
4.6	Número de pedidos de conexão entre nós da rede NSFNet. . . . .	35
4.7	Estatística da rede NSFNet para o modelo SP - caso estático. . . . .	35
4.8	Capacidades residuais $r(\psi'(\hat{p}), p)$ da rede no estado $\psi_0$ . . . . .	40
4.9	Capacidades residuais $r(\psi'(\hat{p}), p)$ da rede no estado $\psi_1$ . . . . .	40
4.10	Capacidades residuais $r(\psi'(\hat{p}), p)$ da rede no estado $\psi_2$ . . . . .	41
4.11	Capacidades residuais $r(\psi'(\hat{p}), p)$ da rede no estado $\psi_3$ . . . . .	41
4.12	Validação do modelo SP. . . . .	49
4.13	Validação do modelo $M\Sigma$ . . . . .	49
4.14	Estatística da rede de Seis Nós. . . . .	50
4.15	Estatística da rede COST239EON. . . . .	51
4.16	Estatística da rede NSFNet. . . . .	51
A-1	Diferentes bandas de comprimento de onda em fibra óptica. . . . .	54

# Acrônimos

DBR	Distributed Bragg Reflector
DEMUX	Demultiplexer
EDFA	Erbium-Doped Fiber Amplifier
EON	European Optical Network
FWM	Four-Wave Mixing
ILP	Integer Linear Programming
IP	Internet Protocol
ITU	International Telecommunication Union
MUX	Multiplexer
NP	Non-Polynomial
NSFNet	National Science Foundation Network
O/E/O	Optical-to-Electrical-to-Optical
OADM	Optical Add/Drop Multiplexer
OLA	Optical Line Amplifier
OLT	Optical Line Terminal
OSC	Optical Supervision Channel
OXC	Optical Crossconnect
RWA	Routing and Wavelength Assignment
SOA	Semiconductor Optical Amplifier
SONET	Synchronous Optical Network
SP	Shortest Path
STS	Synchronous Transport Signal
WAN	Wide Area Network
WCC	Wavelength Continuity Constraint
WDM	Wavelength Division Multiplexing
XGM	Cross-Gain Modulation
XPM	Cross-Phase Modulation

# Capítulo 1

## Introdução

### 1.1 Considerações Gerais

Até uns poucos anos atrás, as características de projeto das redes de telecomunicações e seus custos, eram determinados pelo tráfego de voz. Hoje, o tráfego de dados influencia significativamente a forma de projetos de redes. Dados estatísticos justificam este fenômeno: uma chamada telefônica de voz dura em média cerca de 3 minutos; em contraste, os usuários que se conectam à Internet permanecem conectados em média cerca de 20 minutos. As conexões à Internet introduzem cerca de seis vezes mais tráfego na rede que as chamadas telefônicas, e o tráfego na Internet duplica-se a cada 6 meses, e esta tendência aparentemente continuará com esse ritmo por algum tempo mais [RS02].

A contínua demanda por maior largura de banda está fortemente correlacionada ao custo da mesma. Os avanços tecnológicos tem tido sucesso em diminuir o custo da largura de banda, e este custo reduzido por sua vez, impulsiona o desenvolvimento de novas aplicações que fazem uso de maior largura de banda.

Os fatores mencionados têm impulsionado o desenvolvimento das redes ópticas de alta capacidade e sua transição marcadamente rápida dos laboratórios de pesquisa para a instalação comercial. Em particular, a tecnologia de multiplexação por divisão de comprimento de onda *Wavelength Division Multiplexing* (WDM) mostra-se promissora para satisfazer altas demandas de largura de banda.

## 1.2 Multiplexação por Divisão no Comprimento de Onda.

A fibra óptica tem uma largura de banda extremamente alta (ao redor de 25 THz, na banda de 1.550 nm). Contudo, apenas taxas de transmissão de uns poucos gigabits por segundo são utilizadas, dado que a taxa na qual um usuário pode acessar a rede é limitada pela velocidade de processamento dos equipamentos eletrônicos [MG02]. Três novas técnicas ampliam a banda de frequências usadas nas fibras ópticas

- WDM, técnica pela qual pode-se transmitir, através de uma fibra óptica, vários feixes de luz de diferentes comprimentos de onda simultaneamente;
- Amplificador de fibra com dopagem de Érbio (*Erbium-Doped Fiber Amplifier* EDFA), que amplifica os sinais de diferentes comprimentos de onda qualquer que seja o esquema de modulação ou a taxa de transmissão;
- Conversão de Comprimentos de Onda, que possibilita que todos os comprimentos de onda das fibras em cascata na rede sejam plenamente utilizados, ou seja libera a restrição de continuidade de comprimento de onda em uma conexão ponto-a-ponto [VSK00].

O WDM é conceitualmente similar à multiplexação por divisão de frequência. Múltiplos sinais modulam sinais ópticos de diferentes comprimentos de onda, e os sinais resultantes são combinados e transmitidos simultaneamente sobre a mesma fibra óptica.

O principal atrativo do WDM é que pode-se obter um incremento significativo da largura de banda disponível sem o enorme investimento necessário para se instalar fibras ópticas adicionais. Por exemplo, para transmitir 40 Gb/s sobre 600 km utilizando um sistema tradicional, requerem-se 16 pares de fibra óptica, cada um operando a 2.5 Gb/s, e 224 ( $[(600/40 - 1) * 16]$ ) regeneradores, dado que estes são instalados a cada 40 km. Por outro lado, um sistema WDM de 16 canais requer só um par de fibra óptica, e 4 ( $(600/120 - 1)$ ) amplificadores ópticos, dado que os amplificadores são instalados a cada 120 km. Sistemas WDM que usam 32 comprimentos de onda a OC-192 para proporcionar taxas de até 320 Gb/s, encontram-se disponíveis no mercado. A técnica WDM densa *Dense Wavelength Division Multiplexing* (DWDM) incrementa o número total de canais em uma fibra utilizando

canais com espaçamento muito estreito. Os espaçamentos típicos variam de 0,4 nm a 4 nm (vide Apêndice A)

Os amplificadores EDFA permitiram o desenvolvimento comercial dos sistemas WDM, ao proporcionarem um método de amplificar todos os comprimentos de onda de forma simultânea, sem se importar com as taxas de transmissão individuais, o esquema de modulação, ou os níveis de potência dos sinais. Anteriormente à invenção do EDFA, os efeitos das perdas ópticas eram compensadas utilizando-se regeneradores eletrônicos, que requeriam que os sinais ópticos fossem convertidos em sinais elétricos e novamente em sinais ópticos. Os regeneradores eletrônicos operam com apenas uma taxa de transmissão e um único comprimento de onda. O EDFA contém vários metros de fibra de silício dopada com íons de Érbio. Um laser óptico de potência é usado para energizar os íons de Érbio, os quais amplificam os sinais ópticos. Por uma feliz coincidência, esta banda de amplificação (1,53 - 1,56 microns), com uma faixa de 0,03-0,04 microns, coincide com a banda de 1,55 microns (de baixa atenuação) das fibras ópticas.

### **1.3 Organização da Dissertação**

A apresentação deste trabalho está organizada como segue. No Capítulo 2, apresenta-se as redes WDM descrevendo as características e os elementos que compõem uma rede óptica WDM típica. No Capítulo 3, aborda-se as características das redes ópticas WDM encaminhadas por comprimento de onda, descrevendo-se os tipos de encaminhamento existentes e as heurísticas de atribuição de comprimentos de onda, assim como as técnicas existentes para a conversão de comprimentos de onda. No Capítulo 4, aborda-se o encaminhamento de comprimentos de onda em redes WDM, apresentando-se duas novas propostas de encaminhamento, uma para o caso estático, comparada com a solução multi-fluxo, e uma para o caso dinâmico, comparada com a heurística Max-Sum [ZJM00]. Finalmente no Capítulo 5, apresentam-se as conclusões e possíveis extensões para este trabalho.

## Capítulo 2

# Características e Elementos da Arquitetura de Redes WDM

São apresentados neste capítulo as características e os elementos básicos de uma rede WDM. Uma abordagem mais detalhada pode ser encontrada em [Rib03]

### 2.1 Características da Arquitetura de Redes WDM

As características mais importantes da arquitetura de redes WDM são:

#### 2.1.1 Reutilização de Comprimento de Onda

Um caminho óptico é uma rota estabelecida entre dois nós (fonte e destino da informação), gerada pela alocação de um comprimento de onda em cada um dos enlaces que compõem o caminho. Múltiplos caminhos podem usar o mesmo comprimento de onda, desde que eles não se sobreponham numa mesma fibra. Esta capacidade espacial de reutilização permite à rede ter um número grande de caminhos utilizando um número limitado de comprimentos de onda.

#### 2.1.2 Conversão de Comprimento de Onda

Os comutadores ópticos podem conter conversores de comprimento de onda . A conversão pode aumentar a utilização dos comprimentos de onda na rede e diminuir a probabilidade de bloqueio na mesma. A conversão de comprimentos de onda também é necessária nas fronteiras da rede para adaptar os sinais exteriores à rede a um comprimento de onda adequado à

rede.

### 2.1.3 Transparência

Dado que múltiplos canais WDM podem ser operados independentemente em uma mesma fibra, estes podem transportar dados em uma variedade de taxas de bit e formatos, incluindo alguns canais analógicos e outros digitais, se desejado. Os protocolos que controlam as transferências de dados em diferentes canais podem ser diferentes, de forma que pode-se estabelecer subredes independentes operando em diferentes conjuntos de canais WDM sobre a mesma estrutura de fibra óptica [MBRM96].

### 2.1.4 Comutação de Circuitos

Os caminhos proporcionados pela camada óptica podem ser ativados ou desativados em função das necessidades impostas pela demanda. Isto é análogo ao estabelecimento ou terminação de circuitos em uma rede de comutação de circuitos, exceto pelo fato que a frequência de estabelecimento ou terminação tende a ser muito menor no caso da rede fotônica do que no caso da rede telefônica, dada a enorme diferença de largura de banda alocada.

### 2.1.5 Confiabilidade

A rede pode ser configurada para que, em caso de falhas nos enlaces, os caminhos já existentes possam ser re-roteados automaticamente por rotas alternativas [RS02]. Em geral, dado que uma rede WDM carrega uma enorme quantidade de tráfego, torna-se mandatório que a rede possua tempos de reconfiguração muito curtos, tipicamente da ordem de milisegundos.

## 2.2 Elementos da Arquitetura de Redes WDM

A arquitetura de uma rede típica WDM é mostrada na Figura 2.1. A rede é composta de *terminais ópticos de linha* (Optical Line Terminal), *multiplexadores ópticos para inserir/retirar* (Optical Add/Drop Multiplexer), e *comutadores ópticos* (Optical Cross Connect) interconectados através de enlaces ópticos. Não aparecem na figura os *amplificadores ópticos de linha* (Optical Line Amplifier), que são instalados ao longo dos enlaces para amplificar o

sinal óptico. Os OLTs, OADMs, e OXCs podem também incorporar amplificadores ópticos para compensar as perdas.

A arquitetura WDM suporta uma variedade de topologias, incluindo anel e malha. Os OLTs multiplexam múltiplos comprimentos de onda em uma única fibra e também demultiplexam um sinal composto WDM em comprimentos de onda individuais. Os OLT's são usados em nós extremos de um enlace ponto-a-ponto. Os OADMs são usados onde alguns comprimentos de onda precisam ser retirados localmente e outros precisam ser inseridos. Os OADMs são instalados tipicamente em topologias em anel. Os OXCs realizam uma função similar aos OADMs, mas em uma escala maior em termos de número de portas e comprimentos de onda envolvidos, e são instalados em topologias em malha ou para interconectar múltiplos anéis [RS02].

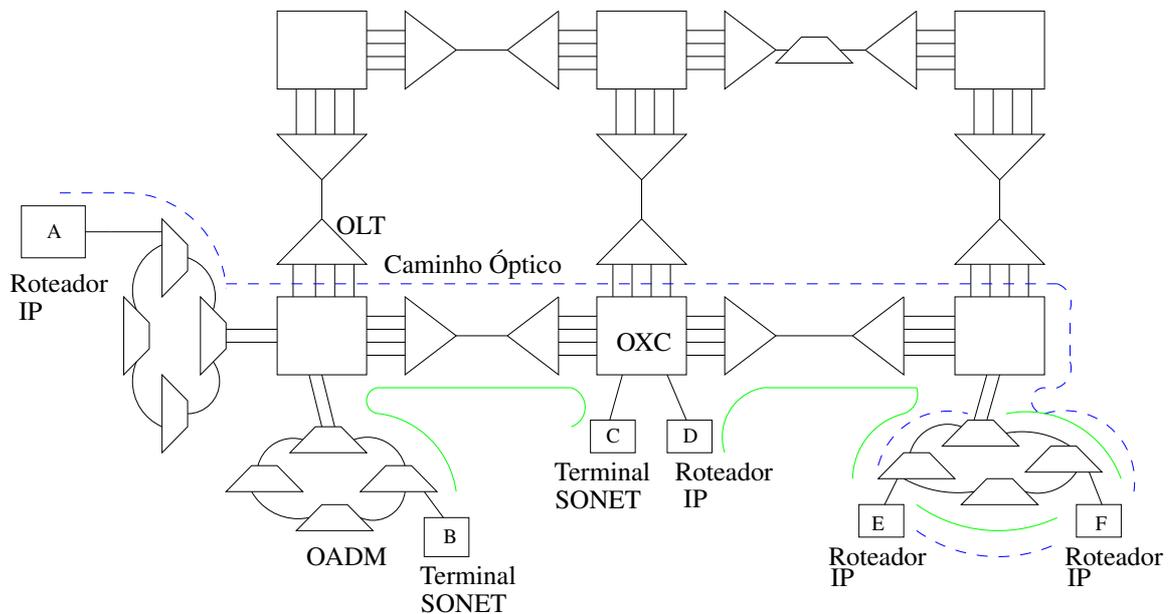


Figura 2.1: Diagrama em blocos de uma rede WDM típica.

### 2.2.1 Terminais Ópticos de Linha

Os *terminais ópticos de linha* (OLT) são elementos de rede relativamente simples sob o ponto de vista da arquitetura da rede. Os elementos funcionais de um OLT são mostrados na Figura 2.2: *conversores, multiplexadores de comprimento de onda*, e opcionalmente *amplificadores ópticos*.

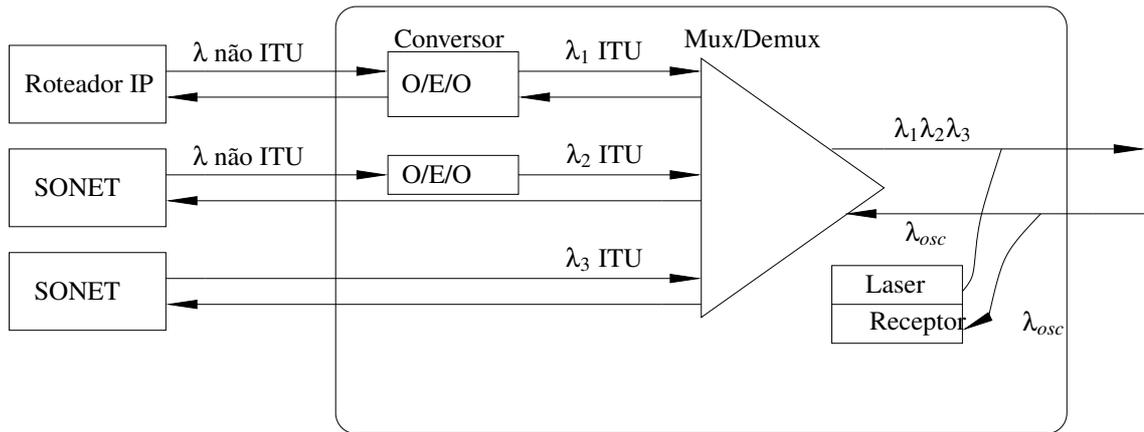


Figura 2.2: Diagrama em blocos de um OLT.

Um conversor adapta o sinal de entrada de um usuário fora da rede óptica em um sinal apropriado para uso dentro da rede óptica. Em forma análoga, no sentido inverso, adapta o sinal da rede óptica em um sinal apropriado para o usuário fora de rede óptica. O sinal de saída do conversor é multiplexado com outros sinais em diferentes comprimentos de onda usando um multiplexador de comprimento de onda. Adicionalmente, pode ser usado um amplificador óptico na saída. Na outra direção, o sinal WDM é novamente amplificado, antes de ser enviado ao demultiplexador que extrai os comprimentos de onda individuais. Estes comprimentos de onda são novamente enviados a um conversor ou diretamente colocados no equipamento do usuário. O OLT também possui um *canal óptico de supervisão* (OSC). O OSC é transportado em um comprimento de onda separado, diferente dos comprimentos de onda que levam o tráfego do usuário. O OSC é usado para monitorar o desempenho dos amplificadores ao longo do enlace, assim como outras funções de gerenciamento [RS02].

### 2.2.2 Amplificadores Ópticos de Linha

Os *amplificadores ópticos de linha* (OLA) são instalados nos enlaces ópticos com espaçamentos típicos entre 80 e 120 km. Os elementos funcionais de um OLA são mostrados na Figura 2.3. O elemento principal é o *amplificador de fibra com dopagem de Érbio* (EDFA).

O OLA típico utiliza dois ou mais blocos de amplificação (EDFA) em cascata com elementos intermediários. Esses elementos incluem compensadores de dispersão, para contrabalançar a dispersão cromática acumulada ao longo do enlace, e também OADMs. Os

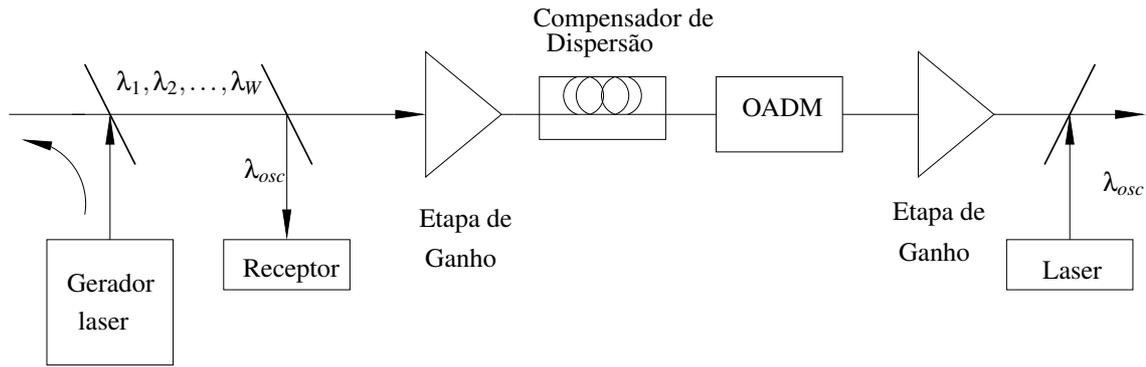


Figura 2.3: Diagrama em blocos de um amplificador óptico de linha.

amplificadores também incluem controle automático de ganho e monitor de desempenho do sinal.

### 2.2.3 Multiplexadores Ópticos de Inserção e Retirada de Canais

Os *multiplexadores ópticos de inserção e retirada de canais* (OADM) provêm um meio para administrar o tráfego intermediário em redes e podem ser usados para amplificação dos sinais. Três diferentes arquiteturas de OADM são mostrados nas Figuras 2.4 a 2.7.

Na arquitetura paralela mostrada na Figura 2.4 [RS02], os sinais de entrada são demultiplexados e alguns dos canais demultiplexados podem ser retirados localmente. Esta arquitetura impõe restrições mínimas ao planejamento de caminhos na rede e a perda através do OADM é fixa, independentemente de quantos canais são retirados e inseridos. Esta arquitetura não é adequada (custo/benefício) para manejar um número pequeno de canais a retirar, pois os canais precisam ser demultiplexados e multiplexados novamente. Como todos os canais são demultiplexados e multiplexados em todos os OADMs, cada caminho óptico passa por muitos filtros antes de chegar a seu destino, o que resulta em tolerâncias estreitas na variação permitida dos comprimentos de onda dos multiplexadores e dos lasers.

Pode-se conseguir algumas melhoras no custo dos OADMs utilizando-se arquitetura modular, como mostrado na Figura 2.5, em que a multiplexação e demultiplexação são feitas em duas etapas. A primeira etapa separa os comprimentos em faixas, e a segunda etapa separa as faixas em canais individuais. Além da economia de custo nos multiplexadores e demultiplexadores, o uso de faixas permite que os sinais sejam transferidos com menor e melhor

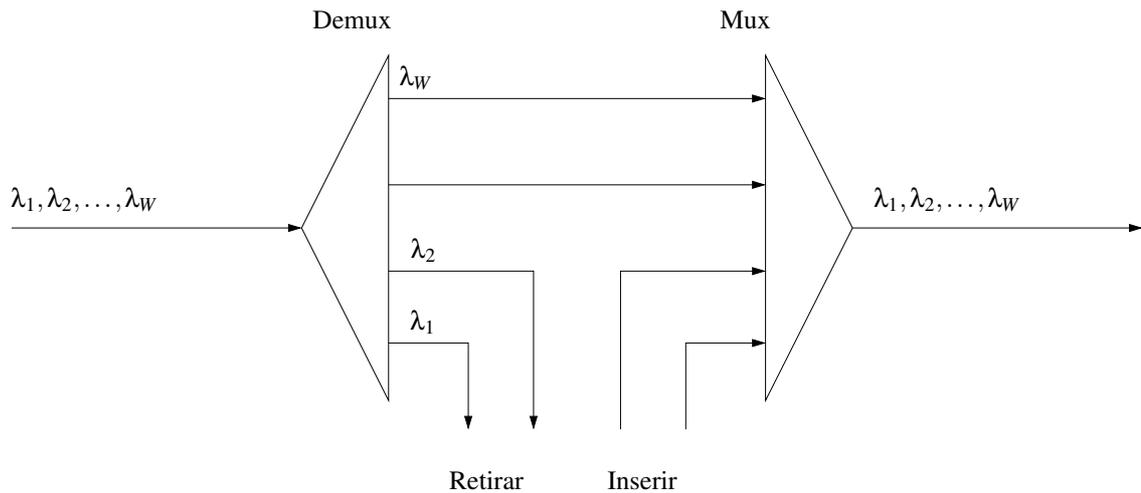


Figura 2.4: Arquitetura paralela do OADM.

uniformidade das perdas ópticas.

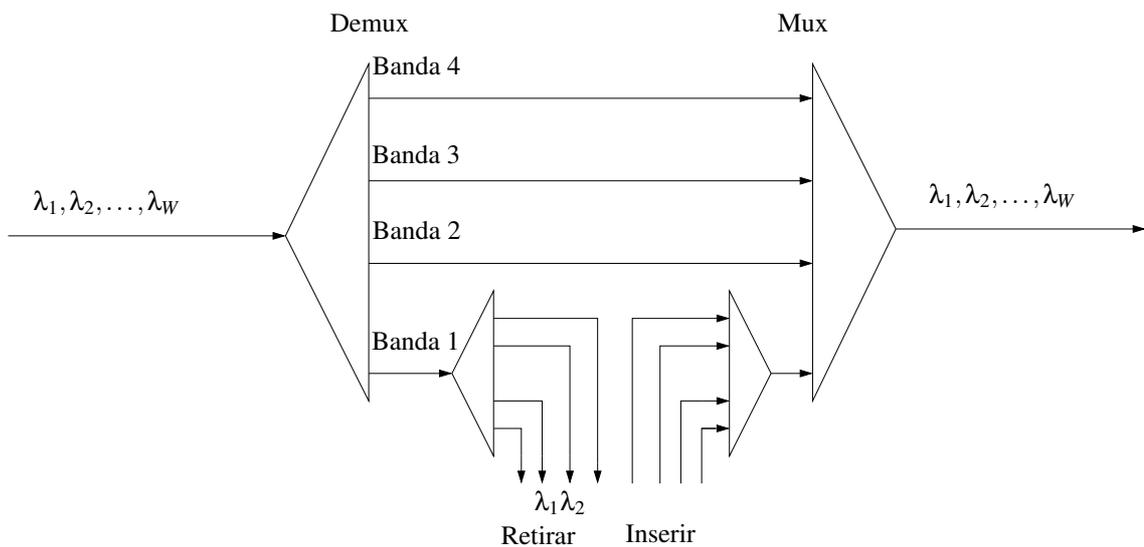


Figura 2.5: Arquitetura modular paralela OADM.

Na arquitetura serial mostrada na Figura 2.6, um único canal é retirado e inserido de um conjunto de canais de entrada, e para retirar e inserir múltiplos canais, colocam-se em cascata vários OADMs.

Retirar e inserir canais adicionais desordena os canais existentes. Portanto é desejável para este caso definir com antecedência qual será o conjunto de canais que precisa ser retirado em cada lugar, para minimizar tal desordem.

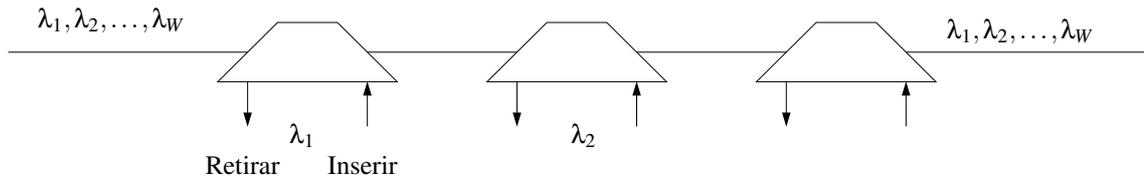


Figura 2.6: Arquitetura serial OADM.

Na arquitetura de retirada de banda, mostrada na Figura 2.7 um conjunto fixo de canais é retirado e inserido do conjunto agregado de canais. Os canais retirados passam através de um nível adicional de demultiplexação, para serem separados. Os canais inseridos são usualmente combinados entre si mediante acopladores simples.

Esta arquitetura tenta obter um balanço entre as arquiteturas paralela e serial. O número máximo de canais que podem ser retirados é determinado pelo tipo de filtro de banda utilizado. Inserir ou retirar canais adicionais na banda não afeta os demais caminhos, pois a perda de transmissão para os demais canais fora da banda é fixa.

Contudo, esta arquitetura complica o projeto de alocação de comprimentos de onda na rede, pois o mesmo conjunto de comprimentos de onda é retirado em cada lugar. Este problema pode ser solucionado utilizando diferentes tipos de OADMs, sendo que cada um retira um conjunto diferente de canais. Se a retirada dos comprimentos de onda pode ser planejada com antecedência e a rede permanece estática, então esta poderia ser uma opção viável. No entanto, em redes onde o tráfego muda com o tempo, esta é uma tarefa complexa.

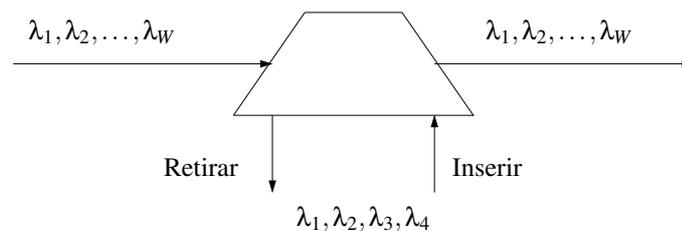


Figura 2.7: Arquitetura de retirada de banda OADM.

## 2.2.4 Comutadores Ópticos

Os OADMs são elementos de rede úteis para topologias de rede simples (anel, por exemplo) com um número pequeno de comprimentos de onda. Para topologias em malha com-

plexas e com um número grande de comprimentos de onda usa-se os *comutadores ópticos* (OXC), que podem usar chaves comutadoras ópticas ou elétricas.

O comutador óptico provê varias funções em redes de grande porte:

- ***Provisionamento de Serviços.***

O OXC pode ser usado para prover caminhos em grandes redes, de forma automatizada, sem ter que recorrer a conexões manuais, o que é importante quando os caminhos na rede precisam ser reconfigurados em resposta a mudanças no tráfego.

- ***Proteção.***

O OXC pode detectar falhas na rede, cortes de fibra e falhas nos equipamentos e reen-caminhar os caminhos automaticamente.

- ***Transparência a taxas de bit.***

O OXC é capaz de comutar sinais com taxas de bit arbitrárias e de formatos diferentes.

- ***Monitoramento do desempenho, acessibilidade a testes, e localização de falhas.***

O OXC provê visibilidade aos parâmetros de desempenho de um sinal em nós intermediários. Permite que os equipamentos sejam conectados a portas dedicadas, nas quais os sinais podem ser monitorados, sem interferir em seu desempenho.

- ***Conversão de comprimento de onda.***

Em adição à capacidade de chaveamento, o OXC pode incorporar capacidade de conversão de comprimento de onda.

- ***Multiplexação e adaptação.***

O OXC trata sinais de entrada e saída de taxas de bit típicas das linhas de transmissão ópticas. Contudo, o OXC pode incorporar capacidade de multiplexação e adaptação para comutar tráfego em granularidades mais finas, como por exemplo a STS-1 (51 Mb/s). Esta multiplexação é feita no domínio elétrico.

## **Capítulo 3**

# **Características das Redes Encaminhadas por Comprimento de Onda**

As características do encaminhamento e atribuição de comprimentos de onda em redes WDM são apresentadas neste capítulo.

### **3.1 Considerações Gerais**

Em uma rede encaminhada por comprimento de onda, uma mensagem é enviada de um nó a outro utilizando um caminho contínuo em comprimento de onda, sem requerer conversão óptica-elétrica-óptica nem armazenamento nos nós intermediários. Os nós extremo do caminho acessam-o utilizando transmissores/receptores que estão sintonizados ao comprimento de onda no qual o caminho opera.

Algumas das características mais importantes das redes encaminhadas por comprimento de onda são: encaminhamento e atribuição de comprimentos de onda, os métodos de seleção de caminho e de comprimento de onda, a restrição de continuidade do comprimento de onda, a conversão de comprimento de onda, e as redes multi-fibras.

### **3.2 Encaminhamento e Atribuição de Comprimentos de Onda**

Nas redes WDM encaminhadas por comprimento de onda, uma conexão é estabelecida através de um caminho. Nas redes sem conversão de comprimento de onda, uma conexão entre um par de nós origem-destino, usa um caminho com comprimento de onda contínuo.

Os algoritmos usados para selecionar comprimentos de onda e estabelecer caminhos são conhecidos como algoritmos de *encaminhamento e atribuição de comprimento de onda* (RWA). A demanda de tráfego pode ser estática (determinística) ou dinâmica (aleatória). No caso de uma demanda de tráfego estática, os pedidos de conexão são conhecidos com precedência. No caso de demanda de tráfego dinâmica, os pedidos de conexão chegam e saem da rede, um por vez, de uma forma aleatória.

Considere a rede em anel com quatro nós e dois comprimentos de onda  $w_0$  e  $w_1$  por fibra, como mostrado na Figura 3.1 [MG02], na qual solicita-se as conexões entre os pares de nós  $\langle 0, 1 \rangle$ ,  $\langle 2, 3 \rangle$ ,  $\langle 1, 3 \rangle$ , e  $\langle 0, 2 \rangle$ . No caso dinâmico, os pedidos de conexão chegam na ordem citada e um algoritmo RWA dinâmico, que estabeleceria os caminhos  $p_0$ ,  $p_1$ , e  $p_2$  aos três primeiros pedidos, usaria o primeiro comprimento de onda livre para o caminho escolhido. Os caminhos  $p_0$  e  $p_1$  usam o comprimento de onda  $w_0$  e  $p_2$  usa  $w_1$ . Não é possível satisfazer o pedido de conexão  $\langle 0, 2 \rangle$ . Um algoritmo RWA estático poderia estabelecer os caminhos  $q_0$ ,  $q_1$ ,  $q_2$  e  $q_3$  para os quatro pares de nós, como mostrado na Figura 3.1(b).

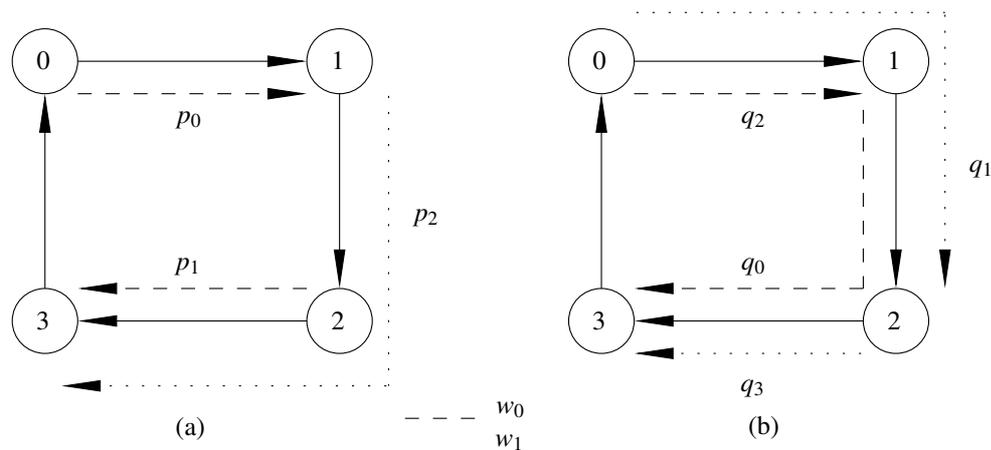


Figura 3.1: Exemplo de encaminhamento e alocação de comprimentos de onda para os casos (a) dinâmico e (b) estático.

### 3.3 Métodos de Seleção de Caminho e de Comprimento de Onda

Embora seja possível tratar o encaminhamento e atribuição de comprimentos de onda como um único problema, este pode ser simplificado decompondo-o em dois subproblemas:

um subproblema de encaminhamento e um outro subproblema de atribuição de comprimentos de onda. Nesta seção, detalha-se primeiro os métodos existentes na literatura para resolver o problema de encaminhamento e, a seguir, os métodos para resolver o problema de atribuição de comprimentos de onda.

### 3.3.1 Métodos de Seleção de Caminho

Os mais importantes métodos de encaminhamento existentes na literatura são *encaminhamento fixo*, *encaminhamento fixo-alternativo* e *encaminhamento adaptativo*.

- ***Encaminhamento Fixo.***

O encaminhamento fixo escolhe sempre o mesmo caminho para um determinado par origem-destino. Estes caminhos fixos são predeterminados.

- ***Encaminhamento Fixo-Alternativo.***

No encaminhamento fixo-alternativo, cada nó mantém uma tabela de encaminhamento que contém uma lista ordenada de um número de caminhos fixos para cada nó destino. Por exemplo, esta tabela pode incluir o caminho mínimo, o segundo caminho mínimo, e assim por diante. Um *caminho primário* entre os nós origem-destino é definido como o primeiro caminho da lista. Um *caminho alternativo* é qualquer outro caminho que não compartilhe enlaces com o primeiro caminho da lista.

Quando chega um pedido de conexão, o nó origem tenta estabelecer a conexão com o nó destino por cada um dos caminhos da tabela de encaminhamento, de forma seqüencial, começando pelo caminho primário, até que um caminho com comprimento de onda válido é encontrado. Se não é possível encontrar um caminho disponível na lista de caminhos alternativos, o pedido de conexão é bloqueado. O encaminhamento fixo-alternativo provê um controle simples para o estabelecimento e desconexão de caminhos, e também provê algum grau de tolerância contra falhas. O encaminhamento fixo-alternativo reduz a probabilidade de bloqueio quando comparado a encaminhamento fixo.

- ***Encaminhamento Adaptativo.***

No encaminhamento adaptativo, o caminho entre um nó origem e um nó destino é escolhido de forma dinâmica, dependendo do estado da rede. O estado da rede é determinado pelo conhecimento dos caminhos utilizados por todas as conexões atendidas num instante de tempo. O encaminhamento adaptativo de menor custo adapta-se às redes com conversão de comprimento de onda. Quando um pedido de conexão chega a um nó de encaminhamento, o caminho de menor custo entre origem e destino é determinado. Se existem múltiplos caminhos com o mesmo menor custo, um deles pode ser escolhido aleatoriamente. O encaminhamento adaptativo requer um grande suporte dos protocolos de controle e gerenciamento, para atualizar continuamente as tabelas de encaminhamento nos nós. Os encaminhamentos adaptativos resultam em uma menor probabilidade de bloqueio que os encaminhamentos fixos ou fixo-alternativos.

### 3.3.2 Heurísticas de Atribuição de Comprimento de Onda.

Para uma demanda de tráfego dinâmica, em que os pedidos de conexão chegam à rede um por vez, em geral, usam-se métodos heurísticos para atribuir comprimentos de onda aos caminhos.

- ***Aleatória.***

Dentre o conjunto de comprimentos de onda disponíveis escolhe-se um de forma aleatória, usualmente com uma distribuição de probabilidade uniforme.

- ***Primeiro-em-Encaixar.***

Os comprimentos de onda são enumerados e o comprimento de onda disponível com menor número é selecionado.

- ***Menos-Usado.***

Seleciona o comprimento de onda com menor uso na rede, procurando balancear a carga entre todos os comprimentos de onda existentes. Os pedidos de conexão que percorrem um pequeno número de enlaces tendem a ser mais satisfeitos que os demais.

- ***Mais-Usado.***

Tenta primeiro o comprimento de onda com maior uso na rede. O seu desempenho é bem melhor que o de Menos Usado [SB97] e um pouco melhor que o de Primeiro em Encaixar.

- ***Mínimo-Produto.***

É usado em redes de múltiplas fibras [JA96] e, nas redes de uma única fibra, torna-se o primeiro em encaixar. O seu objetivo é empacotar comprimentos de onda nas fibras, minimizando o número requerido destas na rede. Não tem desempenho melhor que a versão multi-fibra do primeiro em encaixar, no qual as fibras assim como os comprimentos de onda, estão ordenados.

- ***Menos Carregado.***

Seleciona o comprimento de onda na fibra que tem a maior capacidade residual no enlace mais carregado ao longo de um caminho [KA]. Quando usada em redes mono-fibra, a capacidade residual é 1 ou 0, e portanto a heurística seleciona o comprimento de onda com menor índice, reduzindo-se à técnica *primeiro em encaixar*.

- ***Max-Sum.***

A heurística Max-Sum foi proposta para redes multi-fibras e considera todos os possíveis caminhos disponíveis na rede, tentando maximizar as capacidades nos caminhos não usados depois do estabelecimento da conexão. [BS97] [SB97]

- ***Perda de Capacidade Relativa.***

É uma variante da técnica Max-Sum. Considera cada comprimento de onda disponível e escolhe o comprimento de onda que minimiza a soma das perdas de capacidade relativa para todos os caminhos disponíveis.

- ***Reservação de Comprimento de Onda.***

Um comprimento de onda determinado para um enlace específico é reservado para um fluxo de tráfego, usualmente um tráfego multi-enlace. Reduz a probabilidade de bloqueio para tráfegos multi-enlaces e aumenta-a para pedidos de conexão que percorrem somente um enlace.

- **Limiar de Proteção.**

Os pedidos de conexão de enlace único são atendidos somente se o número de comprimentos de onda não usados no enlace encontra-se em um valor acima de um determinado limiar [BK95].

### **3.4 Restrição de Continuidade do Comprimento de Onda.**

A restrição de continuidade no comprimento de onda implica que um caminho óptico deve usar o mesmo comprimento de onda ao longo de todos os enlaces que o compõem. A restrição refere-se às redes ópticas WDM que não dispõem de conversores de comprimento de onda.

Se um nó de comutação está equipado com capacidade de conversão de comprimento de onda, então a restrição de continuidade do comprimento de onda não se aplica e o problema de encaminhamento é o mesmo que em redes de comutação de circuitos, nas quais o único fator limitante é o número de canais disponíveis em cada enlace.

A restrição de continuidade do comprimento de onda leva a uma utilização ineficiente dos comprimentos de onda e tem como resultado uma probabilidade de bloqueio mais alta. Um pedido de conexão pode ser bloqueado, mesmo que haja comprimentos de onda disponíveis ao longo do caminho que liga os nós origem e destino da conexão solicitada.

### **3.5 Conversão de Comprimento de Onda.**

Em uma rede WDM encaminhada por comprimento de onda, a restrição de continuidade no comprimento de onda pode ser eliminada, utilizando-se conversores de comprimento de onda. Uma rede com capacidade plena de conversão é equivalente, sob o ponto de vista dos algoritmos de encaminhamento, a uma rede comutada de telefonia; portanto, somente o problema de encaminhamento precisa ser resolvido, e a atribuição de comprimentos de onda não é mais um problema a resolver.

A conversão de comprimento de onda pode melhorar a eficiência da rede, ao resolver os conflitos de alocação de comprimentos de onda nos caminhos. Em geral, o emprego de capacidade de conversão plena de comprimentos de onda provê um limite inferior para a

probabilidade de bloqueio dos esquemas de encaminhamento e atribuição de comprimento de onda.

Em muitos casos, a conversão plena de comprimentos de onda não é usada devido aos altos custos. É possível utilizar-se conversores de comprimento de onda para um subconjunto de nós, ou conversores de comprimento de onda compartilhados por mais de um enlace, ou utilizar conversores que somente fazem conversão para um número limitado de comprimentos de onda. Os problemas relativos ao projeto de uma rede WDM encaminhada por comprimento de onda, com uma limitada conversão de comprimento de onda são os seguintes [IM99]

- ***Locação Dispersa dos Conversores de Comprimento de Onda na Rede.***

Enquanto os conversores de comprimento de onda permanecem com um alto custo, não é viável equipar todos os nós de uma rede WDM com estes dispositivos. Os efeitos da conversão dispersa (ter conversão de comprimento de onda somente em alguns comutadores da rede) sobre a probabilidade de bloqueio foram estudados em [SAS96]. Uma questão interessante é *onde* colocar estes poucos conversores para se ter o melhor desempenho possível. Uma técnica heurística é apresentada em [Ine97] para a colocação destes conversores dispersos.

- ***Compartilhamento de Conversores.***

Pode não ser economicamente viável equipar todas as portas de saída dos comutadores com capacidade de conversão. Tem-se proposto arquiteturas de comutadores que permitem o compartilhamento de conversores entre os vários sinais de saída de um comutador. Foi demonstrado em [LL93] que o desempenho da rede satura quando o número de conversores em um comutador atinge um limiar. Um problema interessante é quantificar o efeito deste limiar no algoritmo de encaminhamento usado.

- ***Conversão limitada de Comprimento de Onda.***

Os conversores de comprimento de onda do tipo *mistura de quatro ondas* provêm somente uma capacidade limitada de conversão. Se a gama está limitada a  $k$  opções, um comprimento de onda de entrada  $\lambda_i$  somente pode ser convertido aos comprimentos de

onda  $\lambda_{\max(i-k,1)}$  a  $\lambda_{\min(i+k,w)}$ , onde  $w$  é o número de comprimentos de onda no sistema (indexados de 1 até  $W$ ). Contudo, os testes mostram que as redes que utilizam estes dispositivos tem melhor desempenho que aquelas que utilizam conversores com capacidade de gama completa, sob certas condições [IM99] [YLES96]. A conversão de comprimentos de onda limitada também pode ser provida nos nós usando técnicas de conversão opto-eletrônicas [RS97].

### 3.5.1 Técnicas de Conversão de Comprimento de Onda.

Os conversores de comprimento de onda devem ser capazes de operar a altas taxas de bits, assim como operar em cascata. Devem ter baixo número de componentes, baixa potência de consumo e custo reduzido. Devem ser rapidamente sintonizáveis em uma ampla e contínua gama de comprimentos de onda.

Um método de conseguir a conversão de comprimento de onda é usar um conversor eletro-óptico composto de um detector seguido de um regenerador elétrico, e um laser que retransmite o sinal de entrada em um novo comprimento de onda. Alternativamente, o sinal óptico de entrada é diretamente convertido a um outro comprimento de onda.

Para aplicações de comunicações ultra-rápidas, tem havido grande interesse em conseguir a conversão de comprimento de onda usando meios não lineares, tais como portas interferométricas de fibra óptica e espelhos de realimentação não linear. Embora estes sejam os mais bem sucedidos e permitam regeneração dos pulsos, normalmente requerem altos picos de potência para garantir uma operação não linear suficiente.

Para sistemas operando com códigos de *não retorno a zero*, a conversão de comprimento de onda usando diodos laser com elementos não lineares tem sido investigada. Por exemplo, tem-se conseguido conversão de comprimento de onda usando-se travamento de injeção, saturação de ganho, e absorção saturável em lasers com *reflectores de Bragg distribuído* (DBR). Estes dispositivos podem ser integrados em um único circuito integrado, mas têm limitações, seja por largura de banda ou por requererem altas potências ópticas para a conversão.

Recentemente, a maioria das demonstrações de conversão de comprimento de onda e regeneração têm sido feitas utilizando-se modulação óptica de ganho ou índice refrativo em *amplificadores ópticos a semiconductor* (SOAs). Três principais processos físicos têm sido

utilizados em SOAs para conseguir conversão de comprimento de onda: *modulação de ganho cruzado* (XGM), *modulação de fase cruzada* (XPM), e *mistura de quatro ondas* (FWM).

- ***Modulação de Ganho Cruzado.***

A modulação de ganho cruzado utiliza as interações entre dois sinais ópticos através da população dos portadores. O ganho em um SOA satura quando o nível de potência óptica aumenta. É possível modular o ganho do amplificador por um sinal de entrada e codificar a modulação do ganho em um novo comprimento de onda que atravessa o amplificador. Um sinal de luz de comprimento  $\lambda_1$  modula o ganho e o índice de refração do meio. A amplitude e a fase do feixe de luz contínuo, com comprimento de onda  $\lambda_2$  são mudadas pelo sinal de entrada. Um dos inconvenientes desta técnica é a deterioração da relação sinal-ruído devido ao nível de emissão espontânea. A qualidade do sinal é deteriorada por distorções na amplitude causadas pela modulação da portadora. Contudo, pode-se conseguir, por meio desta técnica, uma conversão de comprimento de onda em taxas elevadas (100 Gb/s). Esta forma de conversão de comprimento de onda é um dos mecanismos, inteiramente ópticos, mais simples disponíveis.

- ***Modulação de Fase Cruzada.***

Os sinais ópticos que atravessam o SOA são submetidos a uma modulação de fase associada à modulação de ganho cruzado. Este efeito de modulação de fase cruzada pode ser explorado em uma configuração de interferômetro de Mach-Zehnder, para conseguir a conversão de comprimento de onda. O interferômetro pode operar em modo não inversor, no qual um aumento na potência elétrica do amplificador produz um aumento na potência do feixe de luz. Comparada à modulação de ganho cruzado, o uso de um interferômetro melhora a qualidade da taxa de extinção do sinal convertido e pode também operar em altas taxas.

- ***Mistura de Quatro Ondas.***

A mistura de quatro ondas origina-se da resposta óptica não linear de um meio, quando estão presentes mais de uma onda. O resultado é a geração de um sinal em uma frequência superior, cuja intensidade é proporcional ao produto das intensidades das ondas in-

cidentes. A fase e a frequência da onda gerada é uma combinação linear das ondas de entrada. A técnica FWM preserva a fase, a amplitude e o formato do sinal, que contém a informação de entrada e permite a conversão simultânea de múltiplos comprimentos de onda de entrada a múltiplos comprimentos de onda de saída.

### 3.6 Redes Multi-Fibra

Em uma rede multi-fibra, um enlace entre dois nós contém um conjunto de fibras, em geral instalado em um único cabo de fibras. A probabilidade de encontrar-se um caminho com comprimento de onda contínuo é mais alta em redes multi-fibras que em redes mono-fibra.

Uma rede multi-fibras com  $F$  fibras por enlace e  $W$  comprimentos de onda por fibra é equivalente a uma rede de uma única fibra com  $FW$  comprimentos de onda com grau de conversão de comprimento de onda igual a  $F$ . A equivalência é mostrada na Figura 3.2.

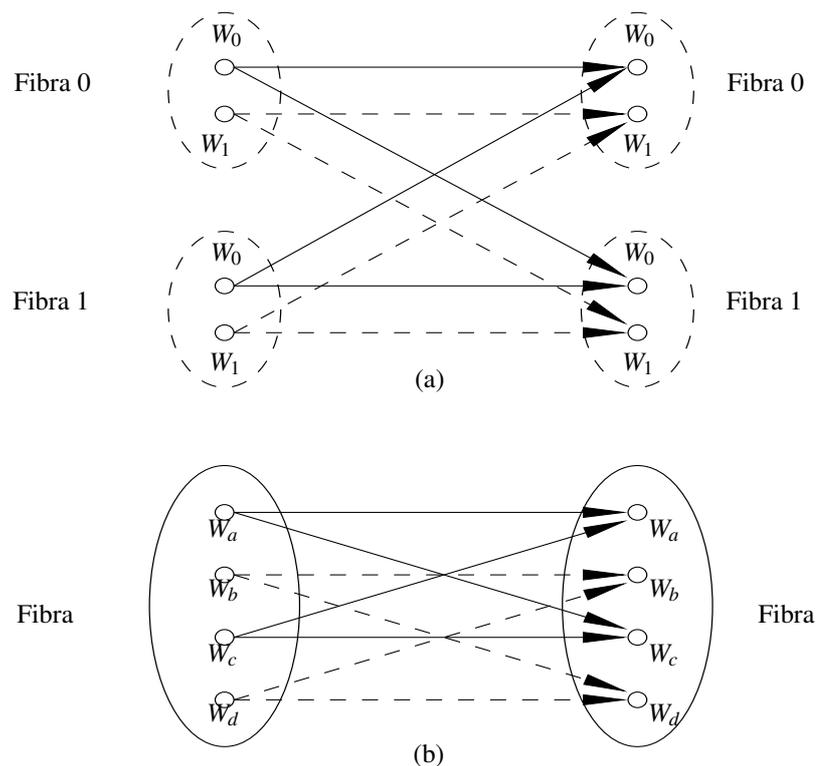


Figura 3.2: Equivalência entre (a) uma rede multi-fibras e (b) uma rede mono-fibra com conversão de comprimento de onda.

## Capítulo 4

# Encaminhamento de Comprimentos de Onda em Redes WDM

O encaminhamento de comprimentos de onda em redes WDM é analisado neste capítulo. Um algoritmo baseado na determinação de caminhos mínimos é proposto e comparado com soluções da literatura.

### 4.1 Introdução

Nas redes ópticas transparentes, os usuários estão conectados entre si através de canais ópticos que não requerem conversão elétrica/óptica nos nós intermediários. Os algoritmos de encaminhamento adequam os comprimentos de onda que usam a banda de transmissão da fibra óptica, de modo que os usuários da rede possam obter a melhor resposta possível da multiplexação dos canais ópticos.

Em geral, seria desejável poder alocar um comprimento de onda diferente para cada pedido de conexão, porém, nas redes ópticas WDM, o número de comprimentos de onda disponíveis nos enlaces limita o número possível de conexões entre cada par de nós. As exigências de espaçamento dos canais de comprimento de onda e a capacidade de sintonia dos transceptores ópticos limitam o número de canais disponíveis.

### 4.2 Classificação

A tarefa de encaminhamento de comprimentos de onda pode ser definida como segue. Dado um conjunto de pedidos de conexão que precisam ser atendidos e dada uma restrição

no número de comprimentos de onda disponíveis, precisa-se determinar os caminhos que devem ser atribuídos a estes pedidos de conexão, de modo que se otimize algum parâmetro de desempenho da rede. O parâmetro a ser otimizado situa-se, em geral, em uma das seguintes três categorias:

- Maximizar o número de caminhos a serem estabelecidos;
- Minimizar o número de comprimentos de onda usados pela rede;
- Minimizar a probabilidade de bloqueio dos caminhos a serem estabelecidos.

Os algoritmos RWA da literatura diferem nas considerações de tráfego e nas métricas de desempenho utilizadas. As considerações do tipo de demanda do tráfego, em geral, estão em uma das seguintes duas categorias:

- ***Demanda Estática (ou Determinística)***, na qual um conjunto de pedidos de conexão para pares origem-destino é definida e conhecida *a priori*;
- ***Demanda Dinâmica (ou Aleatória)***, na qual os pedidos de conexão chegam à rede um a um. Em geral, dispõe-se de um conhecimento probabilístico da demanda.

As métricas de desempenho utilizadas, em geral, estão em uma das seguintes três categorias:

- Número de comprimentos de onda requeridos;
- Probabilidade de bloqueio da conexão, que é definida como a taxa entre o número de conexões bloqueadas e o número total de pedidos de conexão;
- Número de recursos disponíveis nos nós de roteamento, ou custo da fibra.

Para a classe de algoritmos com considerações de tráfego estáticas, o objetivo é minimizar o número de comprimentos de onda de forma a acomodar um determinado número de conexões, ou maximizar o número de conexões acomodadas, se o número de comprimentos de onda é limitado. Para a classe de algoritmos com considerações de tráfego dinâmicas, o objetivo é minimizar a probabilidade de bloqueio.

O estabelecimento estático de caminhos ópticos (*lightpaths*) foi analisado por [MBRM96] por uma métrica que não considera explicitamente a maximização do número de conexões ópticas estabelecidas. Através da combinação de comutação óptica (sem conversão) e comutação eletrônica, as conexões ponto-a-ponto na rede foram determinadas considerando-se a minimização do atraso médio dos pacotes de dados (encaminhados por uma rede SDH sobreposta sobre a rede de caminhos ópticos). A abordagem produziu dois resultados: Uma topologia (virtual) das conexões ópticas e uma estratégia de encaminhamento do tráfego nos pontos com comutação eletrônica. A topologia virtual que deve ser estabelecida sobre os enlaces ópticos, pode ser vista como uma demanda por conexões de caminhos ópticos sobre uma rede de fibras ópticas e constitui um exemplo do problema estático tratado neste capítulo.

### 4.3 Caminho Mínimo

O modelo de caminho mínimo (*Shortest Path SP*) é baseado no conceito de menor distância. Três principais algoritmos de caminho mínimo são conhecidos na literatura: Floyd, Bellman-Ford e Dijkstra [AMO93], com várias aplicações, principalmente em problemas de otimização de rotas de transporte. O algoritmo de Dijkstra foi escolhido por ser de fácil programação e de grande eficiência computacional. Calcula-se o caminho mínimo de um nó origem a todos os demais nós da rede resultando em uma árvore cuja raiz é o nó origem e cujas folhas são os nós destino. O algoritmo Dijkstra opera recursivamente em dois passos: atualiza o potencial (distância até a origem) dos nós fora da árvore mínima; e insere na árvore o nó de menor potencial.

#### 4.3.1 Formulação do Modelo Dijkstra

A construção do modelo necessita de duas definições prévias:

- Topologia Física, descrita por um grafo cujos nós consistem de comutadores ópticos, com capacidade plena de conversão de comprimento de onda, interconectados por fibras ópticas unidirecionais.
- Caminho Óptico (*Lightpath*) estabelecido pela alocação de um comprimento de onda em cada enlace interligando os nós do caminho entre a fonte e o destino da conexão.

**Algoritmo de Dijkstra**

Seja uma rede composta por  $n$  nós e  $l$  enlaces e uma topologia definida pela matriz de distâncias  $C$ . Deseja-se encontrar a árvore com as distâncias mínimas dos nós até o nó origem (escolhido arbitrariamente). O algoritmo de Dijkstra é descrito a seguir.

{Entradas}:

$V$ : {vetor de nós (contém os nós da topologia)};

$i$ : { nó origem};

$C$ : {matriz distância;  $c_{ik} > 0$ }

{Saídas}:

$v(k)$ : {vetor de distância mínima (potencial  $k$  até a origem)};

$p(k)$ : {vetor da árvore mínima};

{Variáveis internas:}

$T$ : {vetor dos nós contidos na árvore mínima};

$U$ : {vetor dos nós fora da árvore mínima};

{Inicialização}:

{insira  $i$  em  $T$  e compute  $U \leftarrow V - T$ };

$v(i) \leftarrow 0$

$p(i) \leftarrow i$ ; {indica que  $i$  é a origem do caminho}

$v(k) \leftarrow +\infty$  { $\forall k \in U$ };

$p(k) \leftarrow -1$  { $\forall k \in U$ };

**while**  $U \neq [ ]$  **do**

    {atualiza  $v(k)$  e  $p(k)$   $\forall k \in U$ };

**if**  $v(i) + d_{ik} < v(k)$  **then**

$v(k) \leftarrow v(i) + d_{ik}$

$p(k) \leftarrow i$

**end if**

    {procura novo nó  $i$  de menor potencial  $v(k)$  para  $k \in U$ . Se houver vários candidatos escolha o de menor  $k$ };

    {insira  $i$  em  $T$  e compute  $U \leftarrow V - T$ };

**end while**

### Exemplo

Considere a rede, mostrada na Figura 4.1, para exemplificar a aplicação do algoritmo de Dijkstra na qual 1 é nó origem. Inicialmente o nó  $i = 1$  recebe  $v(1) \leftarrow 0$  e  $p(1) \leftarrow 1$ , e os demais nós  $v(k) \leftarrow +\infty$  e  $p(k) \leftarrow -1, \forall k \in U$ .

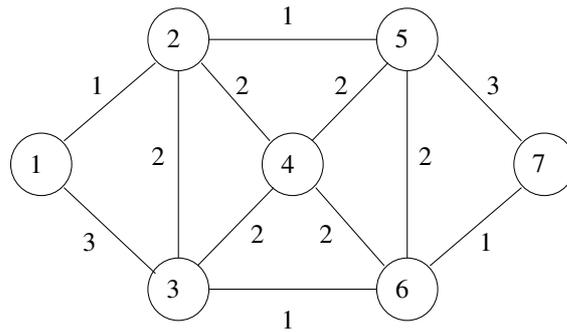


Figura 4.1: Rede com 7 nós e 11 enlaces.

O nó  $i = 1$  é então inserido no vetor  $T$ , fazendo parte da *árvore mínima* e o vetor  $U \leftarrow V - T = [2 \ 3 \ 4 \ 5 \ 6 \ 7]$  é computado. Os valores dos potenciais e dos nós precedentes são atualizados para os nós 2 e 3 e o novo nó escolhido de menor potencial e menor valor  $k$  é  $i \leftarrow 2$ , como mostrado na Figura 4.2.

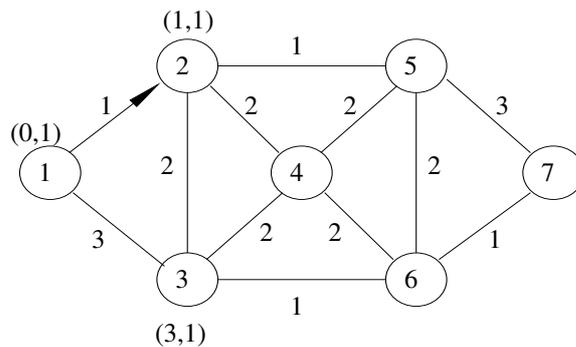


Figura 4.2: Nó 2 incluído na árvore mínima.

O nó 2 é então inserido no vetor  $T$  e os valores dos potenciais e dos precedentes dos nós 4 e 5 são atualizados. O novo nó escolhido, de menor potencial é  $i \leftarrow 5$ , como mostrado na Figura 4.3.

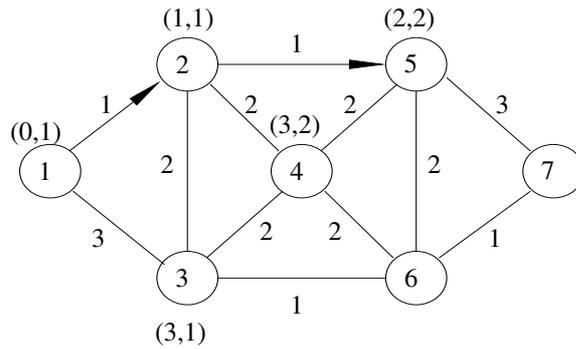


Figura 4.3: Nó 5 incluído na árvore mínima.

O nó 5 é inserido no vetor  $T$  e os valores dos potenciais e dos precedentes dos nós 6 e 7 são atualizados. O novo nó, de menor valor e potencial, escolhido é  $i \leftarrow 3$ , como mostrado na Figura 4.4.

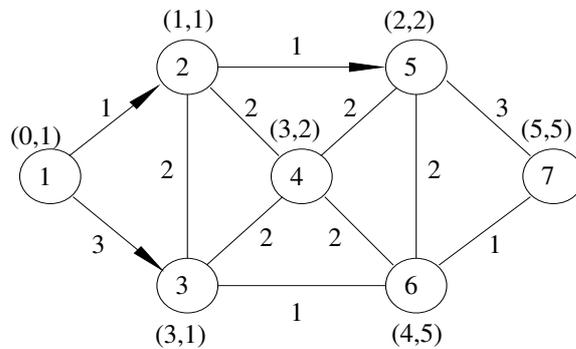


Figura 4.4: Nó 3 incluído na árvore mínima.

O nó 3 é inserido no vetor  $T$  e os valores dos potenciais e dos precedentes dos nós não são atualizados. O novo nó, de menor potencial, escolhido é  $i \leftarrow 4$  como mostrado na Figura 4.5.

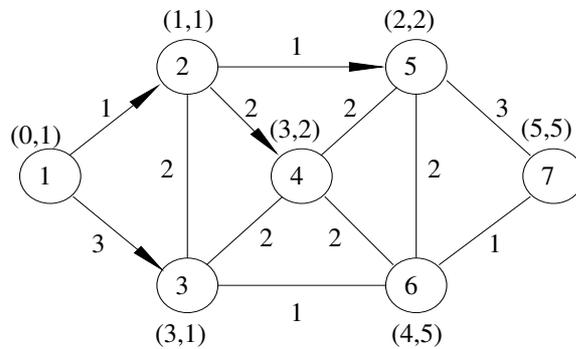


Figura 4.5: Nó 4 incluído na árvore mínima.

O nó 4 é inserido no vetor T e os valores dos potenciais e dos precedentes não são atualizados. O novo nó escolhido é  $i \leftarrow 6$ , como mostrado na Figura 4.6.

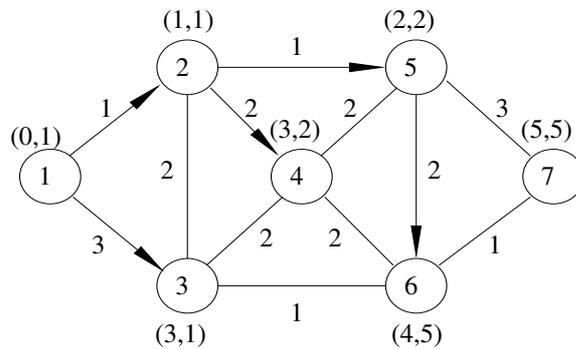


Figura 4.6: Nó 6 incluído na árvore mínima.

O nó 6 é inserido no vetor T e os valores dos potenciais e dos precedentes não são atualizados. O novo nó escolhido é  $i \leftarrow 7$ . O nó 7 é inserido no vetor T e o vetor  $U = [ ]$  (vazio) é computado, como mostrado na Figura 4.7.

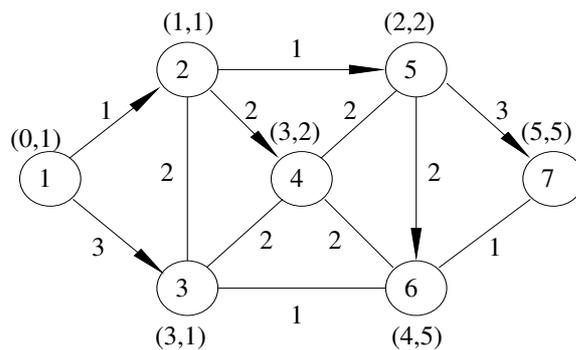


Figura 4.7: Solução de árvore mínima.

## 4.4 Encaminhamento Estático

No caso de demanda de tráfego estática (especificada em termos de pares origem-destino), os pedidos de conexão são conhecidos com precedência e correspondem a requisições cujo tempo de retenção é longo. O objetivo do encaminhamento é estabelecer conexões ponto-a-ponto para minimizar o número de pedidos de conexão bloqueados. Sob o ponto de vista da operadora da rede, esta estratégia corresponde à maximização da receita, pois o número de pedidos de conexão atendidos é maximizado. Para a determinação dos encaminhamentos, a

rede é modelada por um grafo no qual os nós representam os equipamentos de comutação óptica com capacidade plena de conversão de comprimento de onda e os ramos representam as fibras ópticas. Este problema de estabelecimento estático de caminhos é *não polinomial-completo*, portanto soluções ótimas requerem tempos computacionais exponenciais [CGK92] e heurísticas que produzem soluções sub-ótimas são utilizadas [MG02]. Várias heurísticas têm sido propostas na literatura.

Uma heurística baseada na aplicação seqüencial do algoritmo de caminho mínimo de Dijkstra, na qual as conexões com menores distâncias são alocadas prioritariamente, é proposta e comparada com uma formulação de multi-fluxo em redes que resulta em um problema de programação linear inteira (*Integer Linear Programming* ILP).

#### 4.4.1 Formulação do Modelo de Caminho Mínimo

O modelo SP, para o caso estático aqui proposto, segue dois critérios. O primeiro critério é encontrar o caminho mínimo para todo par de demanda (origem-destino) existente, e ordenar estes caminhos em função crescente do número de enlaces a percorrer. O segundo critério consiste em alocar o caminho mínimo para cada pedido processado. O primeiro critério tem como objetivo maximizar o número de pedidos de conexão atendidos. O segundo critério tem como objetivo minimizar o número de enlaces utilizados (e, portanto o número de comprimentos de onda utilizados) na rede. A seguir apresenta-se o algoritmo SP.

##### Algoritmo SP - Encaminhamento Estático

{Entradas}

$C$ : {Matriz de custo unitário }

$D$ : {Matriz de demandas (número de pedidos de conexão)}

$T$ : {Matriz com o número de comprimentos de onda disponíveis}

{Inicialização}

$\text{caminhos} \leftarrow \text{dijkstra\_todos}(C, D)$  {Determina, e ordena segundo o número de hops, o caminho mínimo para todas as demandas}

$k \leftarrow 1$

```

atendidos  $\leftarrow$  0
while  $k \leq \text{size}(\text{caminhos})$  do
  verifica  $\leftarrow$  existe_lambda(caminhos(k))
  if verifica then
    T  $\leftarrow$  atualiza_topologia(caminhos(k))
    D  $\leftarrow$  atualiza_demanda(caminhos(k)) {elimina a demanda atendida}
    atendidos  $\leftarrow$  atendidos + 1
    k  $\leftarrow$  k + 1
  else
    C  $\leftarrow$  atualiza_custos(T)
    caminhos  $\leftarrow$  dijkstra_todos(C,D) {demandas restantes}
    k  $\leftarrow$  1
    if size(caminhos) = 0 then
      break {Pára, se não existirem mais pedidos de conexão por atender}
    end if
  end if
end while

```

#### 4.4.2 Formulação do Modelo de Programação Linear Inteira

O modelo de *programação linear inteira* (ILP) provê uma descrição formal do encaminhamento da rede. Os modelos do problema RWA que utilizam ILP são muito complexos, para as ferramentas computacionais atualmente existentes. A maioria dos modelos ILP dividem o problema RWA em duas partes: um problema de encaminhamento e um outro problema de atribuição de comprimento de onda. Esta separação é feita de modo a encontrar uma solução em um tempo computacional razoável. No modelo ILP [TYM02] descrito, os problemas de encaminhamento e atribuição de comprimento de onda não são separados. A rede é expandida e modelada como um problema de fluxo de rede multiproduto. O problema ILP tem o objetivo de minimizar o número total de comprimentos de onda utilizados, dado um conjunto de pedidos de conexão a serem atendidos.

$$\text{Minimizar } F \quad (4.1)$$

sujeito a

$$F \geq \sum_{s,d,w} F_{ij}^{sdw} \quad (4.2)$$

$$\sum_i F_{ij}^{sd} - \sum_k F_{jk}^{sd} = \begin{cases} -\lambda_{sd} & \text{para } j = s \\ +\lambda_{sd} & \text{para } j = d \\ 0 & \text{para outro valor} \end{cases} \quad (4.3)$$

$$\sum_{sd} F_{ij}^{sdw} \leq n_{ij}^w \quad \forall w \text{ e } \forall (i, j) \quad (4.4)$$

$$F_{ij}^{sdw} \in \mathcal{N} \quad (4.5)$$

onde:

$F_{ij}^{sdw}$  denota o número de caminhos da origem  $s$  ao destino  $d$  no enlace  $ij$  e comprimento de onda  $w$ ;

$\lambda_{sd}$  denota o número de comprimentos de onda requeridos entre a origem  $s$  e o destino  $d$ ;

$n_{ij}^w$  denota o número de comprimentos de onda  $w$  disponíveis no enlace  $ij$ ;

$\mathcal{N}$  denota o conjunto de números inteiros não negativos.

O modelamento considera uma demanda estática para uma rede multifibra. Os pedidos de conexão são conhecidos com precedência e o objetivo é determinar os caminhos e os comprimentos de onda, de forma a minimizar a função objetivo. O congestionamento, o número de comprimentos de onda requeridos e o custo são métricas típicas para os modelos de minimização. Alguns modelos permitem a conversão de comprimento de onda enquanto outros obrigam ao uso da restrição da continuidade do comprimento de onda [TYM02]. Zang [ZJM00] apresentou uma variedade de problemas de encaminhamento e atribuição de comprimentos de onda em redes WDM.

### 4.4.3 Resultados Numéricos

Nesta seção, os resultados numéricos são apresentados, de modo a comparar o desempenho dos dois modelos descritos acima, o modelo ILP e a proposta SP. Os dados de entrada são

as matrizes custo  $C$ , demanda  $D$  e topologia  $T$ . Os dados de saída são o número de pedidos de conexão atendidos, não atendidos, e os tempos computacionais.

### Rede de Seis Nós

Para a rede de seis nós da Figura 4.8, os números rotulados nos enlaces indicam o número de comprimentos de onda disponíveis. A Tabela 4.1 mostra os pedidos de conexão a serem atendidos. A Tabela 4.2 mostra que o número de conexões atendidas e bloqueadas para os dois modelos é o mesmo. As dez demandas atendidas têm os seguintes caminhos:  $ad$ ,  $ad$ ,  $ad$ ,  $abd$ ,  $abd$ ,  $abd$ ,  $abcf$ ,  $abcf$ ,  $abdcf$ , e  $abdcf$ .

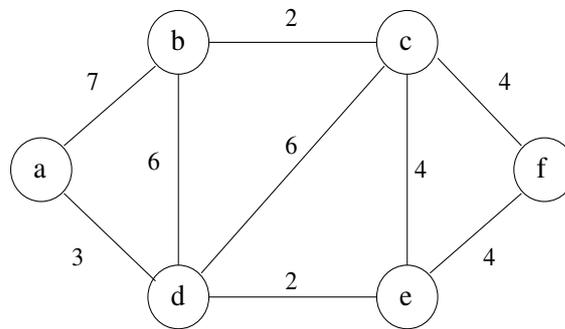


Figura 4.8: Rede de Seis nós com capacidades rotuladas nos enlaces.

Origem	Destino	Demandas
a	d	6
a	f	6

Tabela 4.1: Número de pedidos de conexão para a rede de seis nós.

Modelo	Atendidos	Bloqueados
ILP	10	2
SP	10	2

Tabela 4.2: Número de conexões atendidas e bloqueadas para a rede de seis nós.

### Rede COST239EON

Para a rede óptica europeia COST239EON mostrada na Figura 4.9 [TS95], a Tabela 4.3 [TYM02] indica os pedidos de conexão entre cidades a serem atendidos (total de 150 pedidos). A Tabela 4.4 mostra o número de pedidos de conexão bloqueados como função do

número  $n$  de comprimentos de onda disponíveis por enlace. A Tabela 4.5 mostra o tempo computacional (em segundos) usado pelos modelos, como função de  $(n)$ , para resolver o problema de encaminhamento.

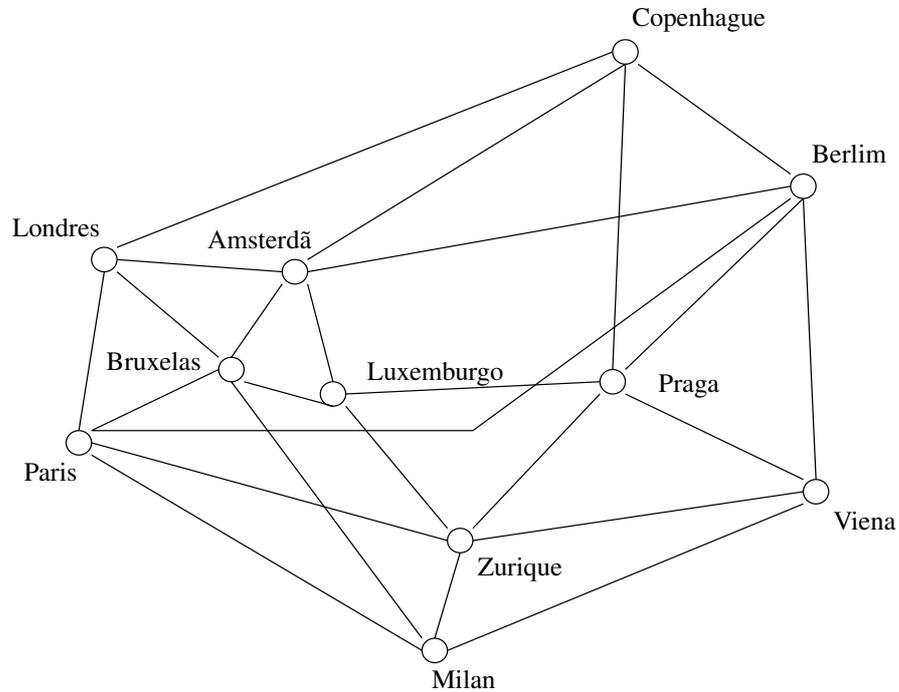


Figura 4.9: Rede COST239EON.

	1	2	3	4	5	6	7	8	9	10	11
1 - Copenhague	—	1	1	1	1	1	1	1	1	1	1
2 - Berlim	1	—	3	3	4	2	2	1	3	1	2
3 - Viena	1	3	—	1	1	1	1	1	1	1	1
4 - Milan	1	3	1	—	2	1	1	1	2	1	1
5 - Paris	1	4	1	2	—	3	2	1	2	1	2
6 - Londres	1	2	1	1	3	—	2	1	1	1	1
7 - Amsterdã	1	2	1	1	2	2	—	1	1	1	1
8 - Praga	1	1	1	1	1	1	1	—	1	1	1
9 - Zurique	1	3	1	2	2	1	1	1	—	1	1
10 - Luxemburgo	1	1	1	1	1	1	1	1	1	—	1
11 - Bruxelas	1	2	1	1	2	1	1	1	1	1	—

Tabela 4.3: Número de pedidos de conexão entre nós para a rede COST239EON.

n	1	2	3	4	5	6
ILP	100	66	40	18	2	0
SP	100	69	43	21	2	0

Tabela 4.4: Pedidos de conexão bloqueados para a rede COST239EON como função do número  $n$  de comprimentos de onda por enlace.

n	1	2	3	4	5	6
ILP	260	650	5.000	9.000	13.000	18.000
SP	36	23	18	9	7	4

Tabela 4.5: Tempos computacionais (em segundos) para a rede COST239EON.

### Rede NSFNet

Para a rede NSFNet mostrada na Figura 4.10 [MBRM96], a Tabela 4.6 [MBRM96] mostra os pedidos de conexão entre os nós (cidades) a serem atendidos. Observe na tabela que cada um dos nós está ligado a 4 outros nós formando uma topologia virtual de conexão. A Tabela 4.7 mostra o número de pedidos de conexão atendidos (*atend.*), o número de pedidos de conexão não atendidos (*n\_atend.*), a percentagem de atendimento (*% atend.*) e o tempo computacional em segundos  $t(s)$  como função do número ( $n$ ) de comprimentos de onda disponíveis.

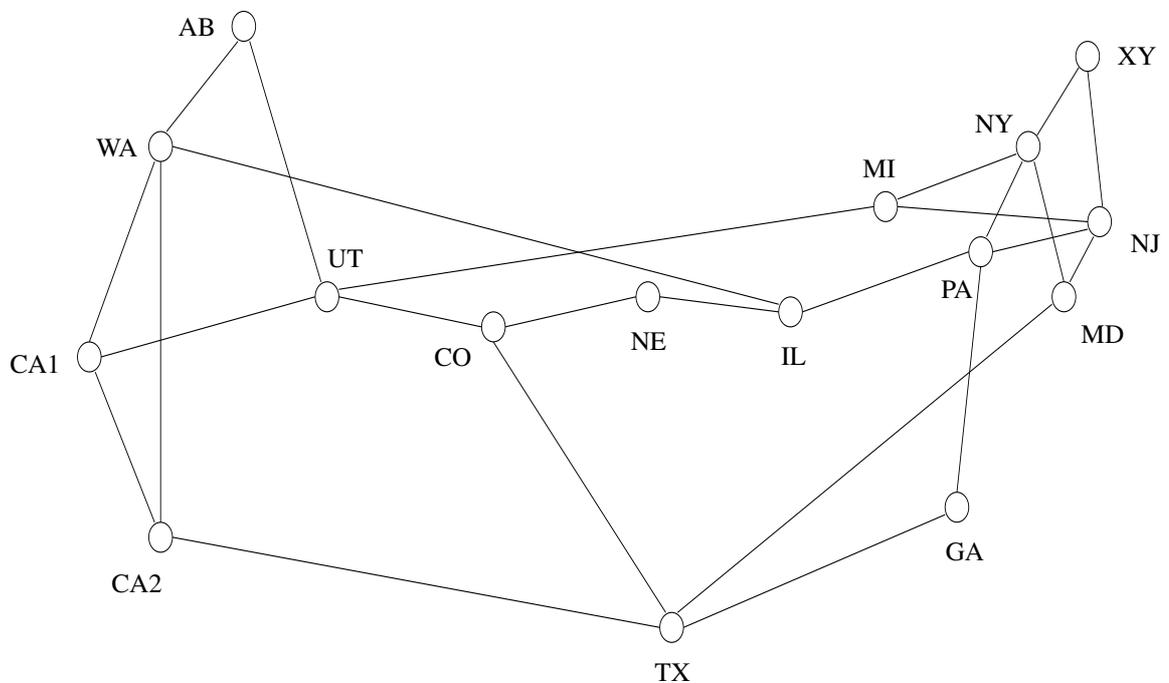


Figura 4.10: Rede NSFNet.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1 - WA	—	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0
2 - CA1	1	—	1	1	0	0	1	0	0	0	0	0	0	0	0	0
3 - CA2	0	1	—	0	0	0	0	0	0	1	0	1	0	0	0	1
4 - UT	0	1	0	—	1	1	0	0	0	0	0	0	0	0	0	1
5 - CO	0	0	0	1	—	0	1	0	0	0	1	0	0	1	0	0
6 - TX	1	0	0	1	0	—	0	0	0	0	0	0	0	1	1	0
7 - NE	0	1	0	0	1	0	—	1	0	1	0	0	0	0	0	0
8 - IL	1	0	0	0	0	0	1	—	1	0	0	0	0	1	0	0
9 - PA	0	0	0	0	0	0	0	1	—	1	0	1	1	0	0	0
10 - GA	0	0	1	0	0	0	1	0	1	—	1	0	0	0	0	0
11 - MI	0	0	0	0	1	0	0	0	0	1	—	0	1	0	0	1
12 - NY	1	0	1	0	0	0	0	0	1	0	0	—	0	0	1	0
13 - NJ	0	0	0	0	0	0	0	0	1	0	1	0	—	1	1	0
14 - MD	0	0	0	0	1	1	0	1	0	0	0	0	1	—	0	0
15 - AB	0	0	0	0	0	1	0	0	0	0	0	1	1	0	—	1
16 - XY	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	—

Tabela 4.6: Número de pedidos de conexão entre nós da rede NSFNet.

n	1	2	3	4	5	6
atend.	26	40	48	56	60	64
n_atend.	38	24	16	8	4	0
% atend.	41	63	75	88	94	100
t(s)	3	3	2	3	2	1

Tabela 4.7: Estatística da rede NSFNet para o modelo SP - caso estático.

## 4.5 Encaminhamento Dinâmico

No caso de demanda de tráfego dinâmica, os pedidos de conexão chegam à rede um por vez, de forma aleatória. Uma vez estabelecidos os caminhos, eles permanecem na rede por um tempo aleatório. A demanda de tráfego dinâmica modela várias situações nas redes de transporte. Pode ser necessário desfazer certos caminhos e estabelecer outros novos, em resposta a padrões de tráfego variáveis ou a partir de falhas nos componentes da rede. Qualquer solução do problema de encaminhamento dinâmico deve ser computacionalmente simples, dado que os pedidos de conexão devem ser processados em tempo real. Quando chega um novo pedido de conexão, precisa-se atribuir um caminho e um comprimento de onda ao pedido, com o objetivo de maximizar o número de pedidos de conexão atendidos (ou

equivalentemente, minimizar o número de pedidos de conexão bloqueados). Os algoritmos dinâmicos têm desempenho pior que os algoritmos RWA estáticos, já que um algoritmo RWA dinâmico não tem conhecimento sobre os pedidos de conexão futuros, enquanto que no caso estático todos os pedidos de conexão são conhecidos com precedência.

Nesta seção, avalia-se comparativamente dois modelos, a proposta denominada modelo SP, baseada no conceito do caminho mínimo, e o modelo  $M\Sigma$  que utiliza a heurística Max-Sum mencionada no Capítulo 3. O desempenho comparativo entre os algoritmos analisados nesta seção é avaliado através de *simulação de eventos discretos*. Os pedidos de conexão são gerados aleatoriamente, segundo a distribuição de Poisson, e os tempos de retenção são gerados aleatoriamente, segundo a distribuição exponencial negativa. Quando não há recurso disponível na rede, a chamada (pedido de conexão) é bloqueada e eliminada do sistema. A diferença entre os algoritmos analisados se dá apenas na forma de encontrar o caminho para atender o pedido de conexão.

#### 4.5.1 Formulação do Modelo de Caminho Mínimo

A seguir mostra-se o algoritmo de caminho mínimo proposto, para uma demanda de tráfego dinâmica.  $1/\mu$  é o tempo médio de retenção da conexão,  $D$  é a matriz de demanda expressa em Erlangs,  $cal$  é o calendário,  $o$  é o nó origem,  $d$  é o nó destino,  $C$  é a matriz de distância entre os nós e  $T$  é a matriz topologia da rede.

##### Algoritmo SP - Encaminhamento Dinâmico

```

 $\Lambda \leftarrow D\mu$ 
 $cal \leftarrow func\_expo(\Lambda)$  {inicializa o calendário com uma chegada para cada par  $od$  com demanda}
 $atendidos \leftarrow 0$ 
 $n\_atendidos \leftarrow 0$ 
 $t \leftarrow 0$ 
while  $t < t_f$  do
   $[t \ o \ d \ f] \leftarrow get(cal)$  {retira o primeiro registro do calendário}
  if  $f = 1$  then
     $\Delta\lambda \leftarrow func\_expo(\Lambda(o,d))$  {f=1, pedido de conexão}

```

```

cal ← put[t + Δλ o d 1] {insere evento no calendário em ordem cronologica}
caminho ← dijkstra(C,o,d)
if existe caminho then
    T ← atualiza_topologia(caminho)
    C ← atualiza_custos(T)
    atendidos ← atendidos + 1
    Δμ ← func_expo(μ(o,d))
    cal ← put[t + Δμ o d 0]
    G ← put[t + Δμ caminho] {insere caminho em ordem cronológica na matriz G}
else
    n_atendidos ← n_atendidos + 1
end if
else if f = 0 then
    caminho ← get(G) {f=0, fim de conexão}
    T ← atualiza_topologia(caminho)
    C ← atualiza_custos(T)
end if
end while

```

#### 4.5.2 Formulação do Modelo Max-Sum.

Para descrever a heurística de alocação de comprimentos de onda do modelo Max-Sum [SB97] introduz-se a seguinte notação. Seja  $\psi$  o estado da rede que especifica os caminhos existentes (caminhos e atribuição de comprimentos de onda) na rede. Seja  $c(\psi, l, w)$  o número de fibras no enlace  $l$  nas quais o comprimento de onda  $w$  está disponível. A capacidade  $r(\psi, p, w)$  do caminho  $p$ , no comprimento de onda  $w$ , é o número de fibras com comprimento de onda  $w$  disponível no enlace com menor capacidade residual ao longo de  $p$ , ou seja,

$$r(\psi, p, w) = \min_{l \in p} c(\psi, l, w). \quad (4.6)$$

A capacidade  $r(\psi, p)$  do caminho  $p$  no estado  $\psi$  é a soma das capacidades em todos os comprimentos de onda, ou seja,

$$r(\Psi, p) = \sum_{w=1}^W r(\Psi, p, w). \quad (4.7)$$

Seja

- $\hat{p}$  o caminho previamente escolhido para estabelecer a conexão em questão.
- $\Psi'(\hat{w}, \hat{p})$  o próximo estado da rede, se o comprimento de onda  $\hat{w}$  for alocado em  $\hat{p}$ .

Max-Sum escolhe o comprimento de onda  $\hat{w}$  no caminho  $\hat{p}$  que maximiza a quantidade

$$\hat{w}^* = \arg \max_{\hat{w} \in \Omega(\Psi, \hat{p})} \sum_{p \in P} r(\Psi'(\hat{w}, \hat{p}), p). \quad (4.8)$$

onde

- $\Omega(\Psi, \hat{p})$  é o conjunto de todos os comprimentos de onda disponíveis no caminho  $\hat{p}$ .
- $P$  é o conjunto de todos os caminhos da rede. Após o comprimento de onda ter sido estabelecido, o estado da rede é atualizado e o seguinte pedido de conexão pode ser processado.

#### Algoritmo $M\Sigma$ Proposto

Propõe-se uma adaptação do algoritmo Max-Sum de alocação de comprimento de onda em redes multi-fibra sem conversão para o problema de encaminhamento em redes com conversão. Considere um pedido de conexão entre os nós  $i$  (nó origem) e  $j$  (nó destino), e seja

- $\Psi$  o estado da rede (o conjunto de encaminhamentos prévios na rede).
- $c(\Psi, l)$  o número de comprimentos de onda disponíveis no enlace  $l$  (em uma ou várias fibras ópticas).
- $p$  um caminho na rede (seqüência de enlaces entre um nó fonte e um nó destino).
- $P$  um sub-conjunto, determinado *a priori*, de todos os caminhos factíveis da rede.
- $\hat{p}$  o caminho factível (com pelo menos um comprimento de onda disponível em cada um dos enlaces que compõem  $\hat{p}$ ) entre os nós  $i$  e  $j$ .

- $\hat{P}_{ij}$  o conjunto de caminhos factíveis entre os nós  $i$  e  $j$ .

A capacidade residual do caminho  $p$  é

$$r(\Psi, p) = \min_{l \in p} c(\Psi, l) \quad (4.9)$$

e, o caminho escolhido  $\hat{p}^*$  é

$$\hat{p}^* = \arg \max_{\hat{p} \in \hat{P}_{ij}} \sum_{p \in P} r(\Psi'(\hat{p}), p). \quad (4.10)$$

onde

$\Psi'(\hat{p})$  é o próximo estado da rede se uma conexão for estabelecida no caminho  $\hat{p}$ ;

Se houver várias soluções toma-se, por exemplo, um caminho entre os de menor distância.

### Exemplo

Considere a rede mostrada na Figura 4.11, e o conjunto de caminhos  $P = \{p_1, p_2, p_3, p_4, p_5\}$ , onde  $p_1 = ad$ ,  $p_2 = abd$ ,  $p_3 = abcf$ ,  $p_4 = adcf$ , e  $p_5 = adef$ .

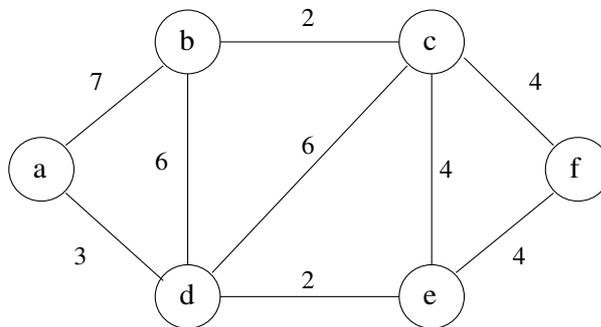


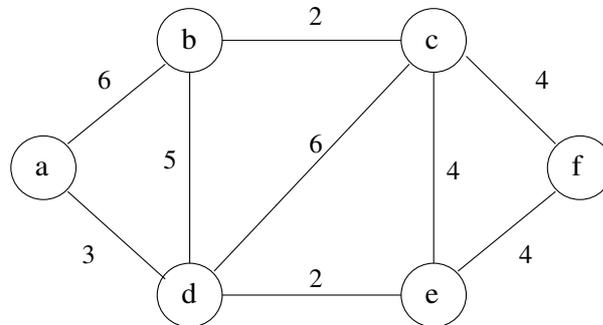
Figura 4.11: Rede de 6 nós.

Se a rede está vazia (sem conexões estabelecidas) então

$$\hat{P}_{ad} = \{p_1, p_2\} \text{ e } \hat{P}_{af} = \{p_3, p_4, p_5\}$$

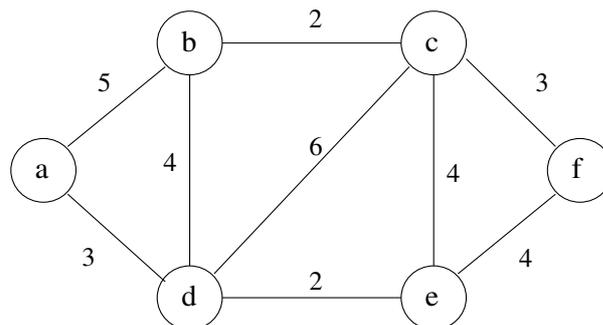
Considere a seguinte seqüência de pedidos de conexão:  $(a, d), (a, d), (a, f), (a, d)$ . As capacidades residuais  $r(\Psi'(\hat{p}), p)$  (rede vazia e pedido de conexão  $(a, d)$ ) são mostradas na Tabela 4.8, e indicam que o caminho  $p_2$  é escolhido, como mostrado na Figura 4.12.

$\hat{p}_{ad}$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$\Sigma$
$p_1$	2	6	2	2	2	14
$p_2$	3	5	2	3	2	15

Tabela 4.8: Capacidades residuais  $r(\psi'(\hat{p}), p)$  da rede no estado  $\psi_0$ .Figura 4.12: Capacidade residuais dos enlaces no estado  $\psi_1$ .

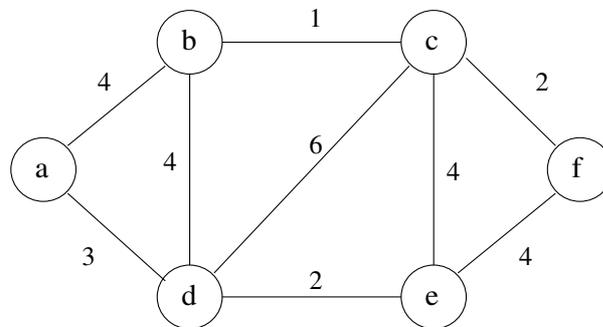
As capacidades residuais no estado  $\psi_1$  ( $p_2$  estabelecido e pedido de conexão  $(a, d)$ ) são mostradas na Tabela 4.9, e indicam que o caminho  $p_2$  é escolhido, como mostrado na Figura 4.13.

$\hat{p}_{ad}$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$\Sigma$
$p_1$	2	5	2	2	2	13
$p_2$	3	4	2	3	2	14

Tabela 4.9: Capacidades residuais  $r(\psi'(\hat{p}), p)$  da rede no estado  $\psi_1$ .Figura 4.13: Capacidades residuais dos enlaces no estado  $\psi_2$ .

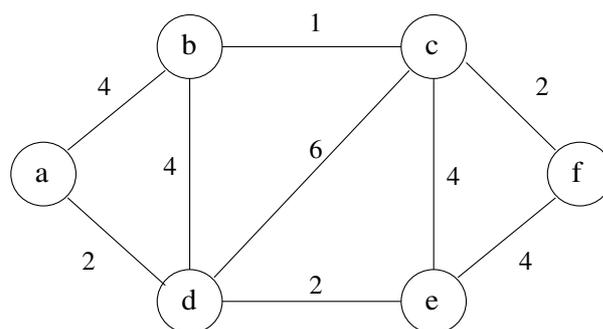
As capacidades residuais no estado  $\psi_2$  ( $p_1$  e  $p_2$  estabelecidos e pedido  $(a, f)$ ) são mostradas na Tabela 4.10, e indicam que o caminho  $p_3$  é escolhido (poderia ter sido  $p_4$ ) como mostrado na Figura 4.14.

$\hat{p}_{af}$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$\Sigma$
$p_3$	3	4	1	2	2	12
$p_4$	2	4	2	2	2	12
$p_5$	2	4	2	2	1	11

Tabela 4.10: Capacidades residuais  $r(\psi'(\hat{p}), p)$  da rede no estado  $\psi_2$ .Figura 4.14: Capacidades residuais dos enlaces no estado  $\psi_3$ .

As capacidades residuais no estado  $\psi_3$  ( $p_1$ ,  $p_2$  e  $p_3$  estabelecidos e pedido  $(a, d)$ ) são mostradas na Tabela 4.11, e indicam que o caminho  $p_1$  é escolhido (por ser o de menor distância), como mostrado na Figura 4.15.

$\hat{p}_{ad}$	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$\Sigma$
$p_1$	2	4	1	2	2	11
$p_2$	3	3	1	2	2	11

Tabela 4.11: Capacidades residuais  $r(\psi'(\hat{p}), p)$  da rede no estado  $\psi_3$ .Figura 4.15: Capacidades residuais dos enlaces no estado  $\psi_4$ .

O algoritmo  $M\Sigma$  proposto tem dois passos para resolver o problema de encaminhamento. O primeiro passo é a escolha dos caminhos, para o qual utiliza-se os seguintes critérios:

- Os caminhos são mínimos, isto é, têm o mesmo número de enlaces a percorrer.
- Os enlaces que compõem os caminhos mínimos são disjuntos.

No segundo passo, aplica-se a heurística Max-Sum, para encontrar o caminho que produz a máxima soma das capacidades residuais para os futuros pedidos de conexão. A seguir detalha-se o algoritmo  $M\Sigma$ .

A seguir mostra-se o algoritmo de caminho mínimo proposto, para uma demanda de tráfego dinâmica.  $1/\mu$  é o tempo médio de retenção da conexão,  $D$  é a matriz de demanda expressa em Erlangs,  $cal$  é o calendário,  $o$  é o nó origem,  $d$  é o nó destino,  $C$  é a matriz de distância entre os nós e  $T$  é a matriz topologia da rede.

#### Algoritmo $M\Sigma$ - Encaminhamento Dinâmico

```

 $\Lambda \leftarrow D\mu$ 
 $cal \leftarrow func\_expo(\Lambda)$  {inicializa o calendário com uma chegada para cada par  $od$  com demanda}
 $caminhos \leftarrow dijkstra\_3(C, D)$  {procura até 3 caminhos mínimos disjuntos}
 $atendidos \leftarrow 0$ 
 $n\_atendidos \leftarrow 0$ 
 $t \leftarrow 0$ 
while  $t < t_f$  do
   $[t \ o \ d \ f] \leftarrow get(cal)$  {retira o primeiro registro do calendário}
  if  $f = 1$  then
     $\Delta\lambda \leftarrow func\_expo(\Lambda(o, d))$  {f=1 pedido de conexão}
     $cal \leftarrow put[t + \Delta\lambda \ o \ d \ 1]$  {insere evento no calendario em ordem cronológica}
    if existe caminho then
       $\hat{P}_{od} \leftarrow compara(caminhos, o, d)$ 
      if  $\hat{P}_{od} \neq [ \ ]$  then
         $caminho\_escolhido \leftarrow max\_sum(\hat{P}_{od}, T, o, d)$ 
         $T \leftarrow atualiza\_topologia(caminho\_escolhido)$ 
         $[C, calc1] \leftarrow atualiza\_custos(T)$  {calc1 indica se um elemento de C foi zerado}
        if  $calc1 = 1$  then

```

```

    caminhos ← dijkstra_3(C,D)
end if
    atendidos ← atendidos + 1
    Δμ ← func_expo(μ)
    cal ← put[t + Δμ o d 0]
    G ← put[t + Δμ caminho_escolhido] {insere caminho em ordem cronológica na
    matriz G}
else
    n_atendidos ← n_atendidos + 1
end if
else
    n_atendidos ← n_atendidos + 1
end if
else if f = 0 then
    caminho ← get(G) {f=0 fim de conexão}
    T ← atualiza_topologia(caminho_escolhido)
    [C, calc0] ← atualiza_custos(T)
    if calc0 = 1 then
        caminho ← dijkstra_3(C,D)
    end if
end if
end while

```

Kodialan e Lakshman [KL00] propuseram um algoritmo para encaminhamento dinâmico bastante similar ao Max-Sum, que denominaram de MIRA (*Minimum Interference Routing Algorithm*). Assim como no Max-Sum, supõe-se que os pedidos de conexão são tratados um-a-um sem conhecimento sobre as demandas futuras, a não ser quanto aos possíveis pares origem-destino.

Na chegada de uma demanda para um dado par origem-destino, supondo-se conhecido o estado da rede (conexões estabelecidas), determina-se a capacidade residual de cada um

dos enlaces e resolve-se um problema de fluxo máximo para cada um dos demais pares de demanda. Determina-se, em seguida, a perda nos fluxos máximos devido à diminuição de uma unidade de capacidade em cada um dos enlaces da rede. A soma dessas perdas corresponde ao valor distância associado a cada um dos enlaces. O encaminhamento do par origem-destino, em questão é determinado pela resolução do problema de caminho mínimo associado. Devido à similaridade entre os algoritmos MIRA e Max-Sum e da maior complexidade do primeiro, decidiu-se comparar o algoritmo SP apenas com o Max-Sum.

### 4.5.3 Validação

#### O Modelo Markoviano

Supondo-se que os pedidos de conexão tenham distribuição poissoniana e que o tempo de retenção dos pedidos de conexão aceitos seja exponencialmente distribuído, a rede pode ser representada usando-se modelos de cadeia de Markov.

Em 1907, A. Markov publicou seu trabalho "Extensão dos teoremas de limites da teoria de probabilidades para uma soma de variáveis conectadas em cadeias", definindo o que é atualmente chamado de processo de Markov, uma forma de representar processos estocásticos. A cadeia de Markov é um caso particular, no qual os processos apresentam estados discretos. A transição entre estados pode ocorrer de forma contínua ou discreta no tempo [Kle75]. As cadeias de Markov são processos sem memória, ou seja, o comportamento do sistema futuro depende apenas de seu estado presente, sem relação com o tempo transcorrido ou com a seqüência de estados percorrida até alcançar o estado atual. O tempo de permanência num dado estado é uma variável aleatória independente do passado.

A fórmula de bloqueio de Erlang pode ser obtida pela aplicação das cadeias de Markov a um enlace de uma rede [Bea76]. O tráfego em uma rede de dados é descrito como um processo de Markov contínuo no tempo, no qual os estados representam as possíveis ocupações dos enlaces. As transições entre estados representam ou o início ou o término de um pedido de conexão aceito.

Suponha-se que a um grupo de  $n$  circuitos é oferecido um tráfego de intensidade  $a$ , definido por  $a = \lambda/\mu$ . A intensidade  $a$  pode ser entendida como o número médio de chegadas durante um tempo médio de retenção. O tempo entre pedidos de conexão sucessivos é distri-

buído exponencialmente com média  $1/\lambda$ . Se um pedido de conexão for iniciado e encontrar os  $n$  circuitos ocupados, é bloqueado e eliminado do sistema. O diagrama de estados de um grupo de  $n$  enlaces de dados é apresentado na Figura 4.16.

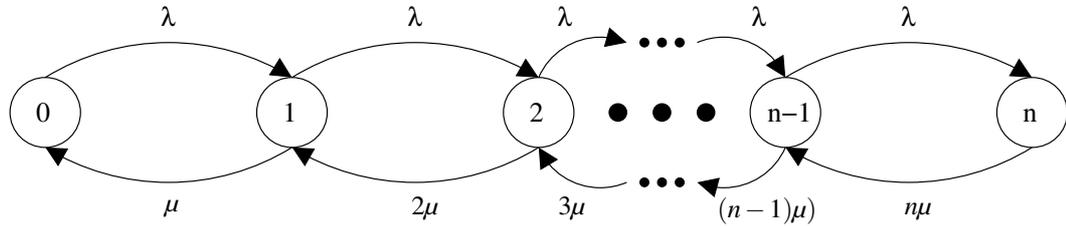


Figura 4.16: Cadeia de Markov de um enlace com  $n$  circuitos.

As condições de estacionaridade das probabilidades de que  $k$  enlaces estejam ocupados ( $P_k$ ) são mostradas nas equações 4.11.

$$\lambda P_0 = \mu P_1; \lambda P_1 = 2\mu P_2; \dots; \lambda P_{n-1} = n\mu P_n \quad (4.11)$$

Que resultam em

$$P_k = \frac{a^k}{k!} P_0 \quad ; k = 1, 2, \dots, n \quad (4.12)$$

Como

$$\sum_{k=0}^n P_k = 1 \quad (4.13)$$

Portanto

$$P_0 = \frac{1}{\sum_{k=0}^n \frac{a^k}{k!}} \quad (4.14)$$

e

$$P_k = \frac{a^k/k!}{\sum_{i=0}^n \frac{a^i}{i!}} \quad (4.15)$$

Define-se como probabilidade de bloqueio a razão entre o número de pedidos de conexão que não foram aceitos e o número total de pedidos de conexão.

$$B_n = P_n = E(a, n) = \frac{a^n / n!}{\sum_{k=0}^n \frac{a^k}{k!}} \quad (4.16)$$

A Equação 4.16 é a fórmula de bloqueio de Erlang. A probabilidade de bloqueio é igual à probabilidade de todos os  $n$  enlaces estarem ocupados, devido à independência entre os inícios dos pedidos de conexão e o estado do enlace. Assim, a probabilidade de um pedido de conexão encontrar o enlace no estado  $k$  é  $P_k$ . O mesmo não ocorre quando o processo de chegada de pedidos de conexão e o processo de ocupação são eventos correlacionados.

O número médio de conexões estabelecidas é

$$\bar{x} = a(1 - E(a, n)) \quad (4.17)$$

Para o caso de demanda de tráfego dinâmica, dado que os pedidos de conexão chegam à rede um por vez de forma aleatória, é preciso utilizar um método que permita estabelecer qual o tempo de simulação necessário para obter resultados confiáveis, ou seja, resultados com um intervalo de confiança apropriado. Para os modelos aqui apresentados, utilizou-se a técnica *bootstrap* [ET93] para a determinação do intervalo de confiança. A técnica utiliza um conjunto de amostras, geradas a partir dos valores de ocupação instantâneos da rede para um determinado tempo de simulação.

### A Técnica *Bootstrap*

A maioria das técnicas para computar a variância dos estimadores de parâmetros ou para estabelecer intervalos de confiança para os valores verdadeiros pressupõe que o tamanho do conjunto de valores da amostra disponível é suficientemente grande, de modo que podem-se aplicar resultados assintóticos. Contudo, na maioria dos problemas, esta suposição não pode ser feita, seja devido a restrições de tempo ou porque o processo é não-estacionário e só pequenas porções de dados estacionários são consideradas.

A técnica *bootstrap* foi introduzida, por B. Efron em 1979, para calcular os intervalos de confiança de parâmetros. Com o *bootstrap*, as observações são reatribuídas aleatoriamente, e as estimativas recalculadas. Estas reatribuições e recálculos são feitos muitas vezes e tratados como experimentos repetidos. A seguir mostra-se uma síntese do algoritmo *bootstrap*.

### Algoritmo *Bootstrap*

- Passo 0: Amostra original  $X = (x_1, x_2, \dots, x_m)$
- Passo 1: Gerar amostra aleatória de  $m$  valores para a média, com reposição, de  $X$ . Amostra *bootstrap*  $Y$ .
- Passo 2: Calcular a média de  $Y$ ,  $\mu_1$ .
- Passo 3: Repetir passos 1 e 2 para obter  $n$  estimadores *bootstrap*  $\mu_1, \mu_2, \dots, \mu_n$ .
- Passo 4: Ordenar os estimadores em forma crescente  $\mu(1), \dots, \mu(n)$ .
- Passo 5: O intervalo de confiança  $(1 - \alpha)100\%$  desejado é  $(\mu(q_1), \mu(q_2))$ , onde  $q_1 = \lfloor n\alpha/2 \rfloor$  e  $q_2 = n - q_1 + 1$ .

### Coleta de Amostras

A coleta de amostras é um processo que permite obter um conjunto de amostras representativas de um experimento. As amostras representam a ocupação média dos enlaces da rede avaliada em instantes de tempo pré-especificados, e são geradas via simulação dos algoritmos descritos.

Para determinar o conjunto de dados amostrais, considere-se um intervalo de tempo finito  $\Delta t$  e um determinado conjunto de valores gerados. Define-se a curva resultante de unir (mediante traços horizontais e verticais) os pontos que correspondem aos valores gerados. Assume-se uma partição de  $\Delta t$ , tal que todo sub-intervalo definido tenha a mesma duração. O conjunto de dados amostrais é a coleção dos valores médios de cada sub-intervalo. O valor médio de cada sub-intervalo é igual em valor à área contida embaixo da curva traçada, dividida pela duração do sub-intervalo.

O conjunto de valores médios obtidos compõe as amostras utilizadas pelo algoritmo *bootstrap* para calcular o intervalo de confiança da simulação. Na Figura 4.17, mostra-se a representação da ocupação de um enlace de uma rede (pontos) e os valores amostrais gerados (asteriscos) para um tempo de simulação de 40 segundos e sub-intervalos de 5 segundos.

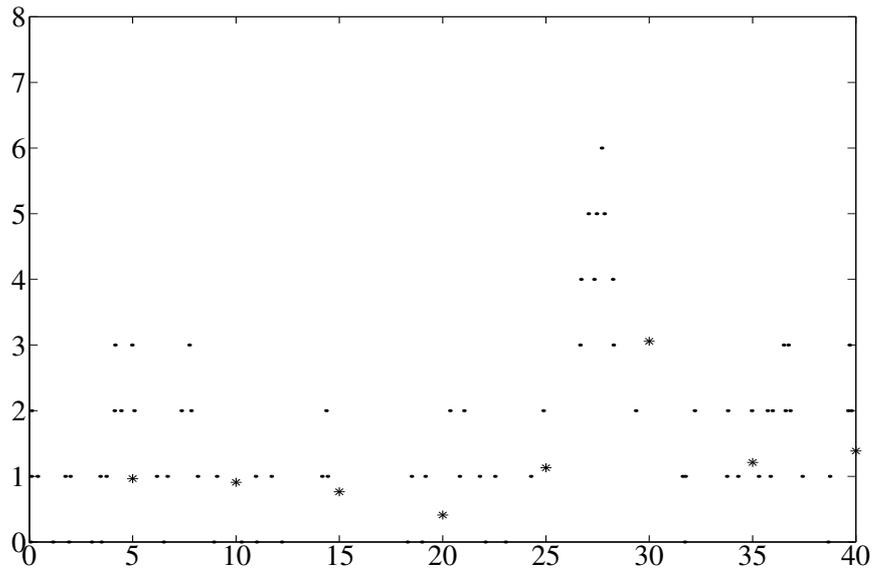


Figura 4.17: Valores (pontos) e dados amostrais (asteriscos) gerados.

Os valores do intervalo de confiança em função da duração dos sub-intervalos para simulações de duração total ( $t_f$ ) de 4.000 segundos (Figura à esquerda) e 1.000 segundos (Figura à direita) são mostrados na Figura 4.18. Observe na Figura à esquerda que o intervalo de confiança é invariante com a duração do sub-intervalo e que em toda a Figura o número de amostras é maior que 40 ( $4000/100$ ). Na Figura à direita, observa-se que o intervalo de confiança apresenta problemas se o número de amostras é menor que 25 ( $1000/40$ ). Observe também que, como esperado, para um maior tempo de simulação tem-se um menor valor para o intervalo de confiança (compare a Figura da direita com a da esquerda).

As Tabelas 4.12 e 4.13 mostram a validação das simulações SP e  $M\Sigma$ , respectivamente, considerando um enlace com 10 comprimentos de onda e um tráfego oferecido  $d$  variando de 1 a 20 Erlangs. Pode-se ver que o valor médio analítico ( $m_{analit}$ ), calculado a partir da Equação 4.17, encontra-se sempre muito próximo ao valor medio experimental ( $medio$ ), entre os valores inferior ( $inf$ ) e superior ( $sup$ ) das simulações. Mostra-se também que para obter intervalos de confiança ( $ic$ ) abaixo de 5 % requerem-se tempos de simulação ( $t_f$ ) que variam entre 16.000 e 800 segundos, segundo o tráfego oferecido  $d$  existente na rede. Observe a diminuição do intervalo de confiança com o aumento do bloqueio, resultado da menor variação do estado de ocupação no enlace.

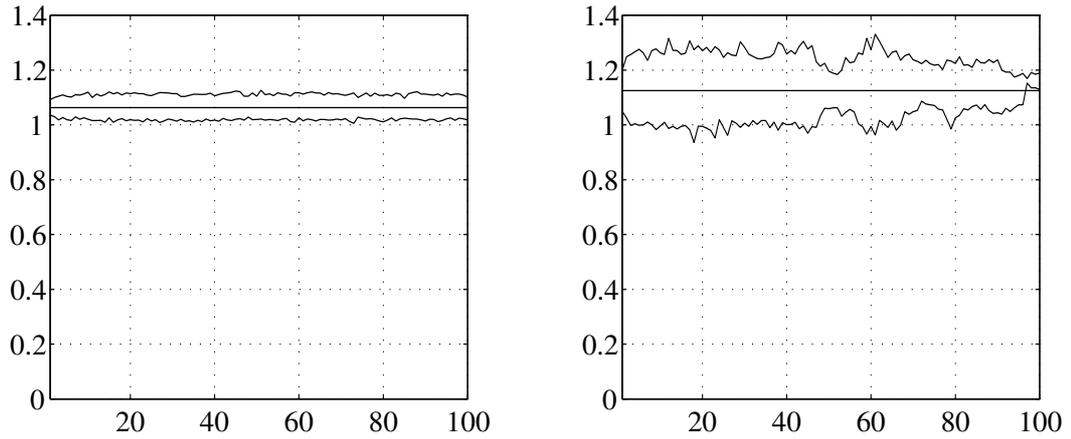


Figura 4.18: Intervalos de confiança para  $t_f = 4.000$  (esquerda) e  $t_f = 1.000$  (direita).

d	1	5	10	15	20
$t_f$	16.000	3.200	1.600	1.067	800
m.analit.	1,00	4,90	7,85	8,84	9,24
inf.	0,97	4,82	7,82	8,81	9,20
sup.	1,01	4,96	7,97	8,89	9,27
medio	0,99	4,89	7,90	8,85	9,23
ic (%)	4,69	2,94	1,93	0,96	0,78
delta	640	128	64	42.68	32
E(10) (%)	0	1,8	21,5	41,0	53,8

Tabela 4.12: Validação do modelo SP.

d	1	5	10	15	20
$t_f$	16.000	3.200	1.600	1.067	800
m.analit.	1,00	4,90	7,85	8,84	9,24
inf.	0,96	4,75	7,76	8,79	9,22
sup.	1,00	4,97	7,94	8,87	9,30
medio	0,98	4,86	7,84	8,84	9,26
ic (%)	4,05	4,53	2,21	0,79	0,85
delta	640	128	64	42.68	32
E(10) (%)	0	1,8	21,5	41,0	53,8

Tabela 4.13: Validação do modelo  $M\Sigma$ .

#### 4.5.4 Resultados Numéricos

Nesta seção mostram-se os resultados, de modo a comparar o desempenho dos modelos anteriormente descritos. Para os algoritmos SP e  $M\Sigma$  descritos, especifica-se os seguintes dados de entrada. Matrizes de custo ( $C$ ), demanda ( $D$ ) e topologia ( $T$ ), e o tempo de simulação ( $t_f$ ). Assume-se, para todas as redes, um tempo médio de permanência das conexões ( $t_m$ ) unitário. Os dados de saída são o número de pedidos de conexão atendidos, não atendidos e a percentagem de pedidos de conexão aceitos. Os pedidos de conexão foram gerados a partir das Tabelas de demandas mostradas para o caso de encaminhamento estático (ver seção 4.4).

##### Rede de Seis Nós

Para a rede de seis nós mostrada na Figura 4.8, utilizando-se o conjunto de pedidos de conexão mostrados na Tabela 4.1, obteve-se, para cada modelo, o número de pedidos de conexão atendidos ( $atend.$ ), o número de pedidos de conexão não atendidos ( $n\_atend.$ ), a percentagem de pedidos de conexão atendidos ( $\% \text{ atend.}$ ), o tempo simulado ( $t_c(s)$ ) em segundos, e a percentagem do intervalo de confiança ( $ic(\%)$ ) obtido pelo método *bootstrap*, como mostrado na Tabela 4.14. O tempo de simulação ( $t_f$ ) utilizado foi de 1.333,33 (16.000,00/12) segundos.

Modelo	SP	$M\Sigma$
atend.	10.932	11.003
n_atend.	5.180	5.167
% atend.	67,85	68,05
$t_c$ (s)	170,00	494,00
ic (%)	1.85	1.84

Tabela 4.14: Estatística da rede de Seis Nós.

##### Rede COST239EON

Para a rede *COST239EON* mostrada na Figura 4.9, utilizando-se cinco (5) comprimentos de onda por enlace, e os números de pedidos de conexão mostrados na Tabela 4.3, obteve-se, para cada modelo, o número de pedidos de conexão atendidos ( $atend.$ ), o número de pedidos de conexão não atendidos ( $n\_atend.$ ), a percentagem de atendimento ( $\% \text{ atend.}$ ), o

tempo simulado ( $t_c(s)$ ), em segundos, e o intervalo de confiança ( $ic(\%)$ ) obtido pelo método *bootstrap*, como mostrado na Tabela 4.15. O tempo de simulação ( $t_f$ ) utilizado foi de 107 (16.000,00/150) segundos.

Modelo	SP	$M\Sigma$
atend.	10.378	10.321
n_atend.	5.542	5.885
% atend.	65,18	63,68
$t_c$ (s)	214,00	14.700,00
ic (%)	2.64	2.06

Tabela 4.15: Estatística da rede COST239EON.

### Rede NSFNet

Para a rede *NSFNet* mostrada na Figura 4.10, utilizando-se cinco (5) comprimentos de onda por enlace, e os números de pedidos de conexão mostrados na Tabela 4.6, obteve-se, para cada modelo, o número de pedidos de conexão atendidos (*atend.*), o número de pedidos de conexão não atendidos (*n\_atend.*), a percentagem de atendimento (*% atend.*), o tempo simulado ( $t_c(s)$ ), em segundos, e o intervalo de confiança ( $ic(\%)$ ) obtido pelo método *bootstrap*, como mostrado na Tabela 4.16. O tempo de simulação ( $t_f$ ) utilizado foi de 250 (16.000,00/64) segundos.

Modelo	SP	$M\Sigma$
atend.	12.179	12.250
n_atend.	3.802	3.717
% atend.	76,2	76,7
$t_c$ (s)	302,00	3.800,00
ic (%)	2.24	2.46

Tabela 4.16: Estatística da rede NSFNet.

As simulações nas três redes mostraram que os algoritmos SP e  $M\Sigma$  têm desempenhos similares. Nas redes com conversão plena de comprimento de onda o aumento de complexidade do algoritmo  $M\Sigma$  em relação ao SP não se justifica. Outros trabalhos têm abordado o problema de encaminhamento de comprimentos de onda em redes ópticas WDM [Jun01] [Nas03].

# Capítulo 5

## Conclusões e Perspectivas

Neste trabalho apresentou-se um estudo sobre as redes ópticas WDM encaminhadas por comprimento de onda, descrevendo-se os principais elementos que as compõem, assim como suas características mais importantes. Apresentou-se também um estudo sobre o algoritmo de caminho mínimo de Dijkstra aplicado ao problema de encaminhamento de comprimentos de onda, e os resultados comparativos da implementação, tanto para demandas de tráfego estáticas, quanto para demandas de tráfego dinâmicas.

Entre as conclusões mais importantes deste trabalho, destaca-se as seguintes:

- Para uma demanda de tráfego estática, a técnica baseada no caminho mínimo (SP) foi validada, pois o número de pedidos de conexão bloqueados obtido foi, em todos os casos, um valor muito próximo do valor ótimo obtido pelo modelo de programação linear inteira (ILP).
- Quanto aos tempos computacionais obtidos, para uma demanda de tráfego estática, o modelo SP tem um desempenho muito melhor que o modelo ILP, devido à grande complexidade do algoritmo de otimização multi-produto usado pela técnica ILP.
- Para uma demanda de tráfego estática, cabe salientar que o modelo ILP possui algumas vantagens sobre o modelo SP. O modelo ILP tende a distribuir uniformemente na rede o número de comprimentos de onda por enlace e minimiza o número de conversores de comprimentos de onda requeridos nos nós. Estas vantagens não foram avaliadas, pois assumiu-se neste trabalho, conversão plena de comprimentos de onda em todos os nós de comutação das redes examinadas.

- Para uma demanda de tráfego dinâmica, o modelo SP teve um desempenho (bloqueio) similar ao modelo Max-Sum. O modelo Max-Sum teve desempenho pior que o modelo SP, em termos de tempos computacionais, devido à complexidade do algoritmo Max-Sum.
- Os resultados obtidos indicam que os parâmetros mais importantes que afetam o desempenho dos algoritmos avaliados são: o número de nós, grau de conectividade da rede (quão malhada é a rede), o número de comprimentos de onda disponíveis nos enlaces e o número de demandas da rede.
- A técnica de validação dos modelos, utilizada para o caso de demanda de tráfego dinâmica mostrou ser robusta, pois os intervalos de confiança obtidos avaliados pela técnica *bootstrap* foram praticamente invariantes com o tamanho dos sub-intervalos para a coleta de amostras.

As análises realizadas neste trabalho permitem sugerir o prosseguimento dos estudos do impacto do uso do algoritmo de caminho mínimo nas redes ópticas. Em particular, dois problemas destacam-se:

- Uso do algoritmo de caminho mínimo combinado com alocação restrita de conversores limitados de comprimento de onda nos nós de comutação;
- Uso do algoritmo de caminho mínimo tendo como métrica de distância, associada ao enlace, uma estimativa probabilística do bloqueio *a posteriori*.

Quanto à síntese das redes ópticas (determinação das capacidades para uma demanda especificada), o uso do algoritmo de caminho mínimo combinado com o bloqueio, computado a partir da fórmula de Erlang, permite vislumbrar o desenvolvimento de algoritmos de síntese (e aplicativos computacionais) relativamente simples e eficazes.

# Apêndice A

## Espectro de Freqüência e Separação de Canais

### A.1 Faixas de comprimento de onda para transmissão por fibra óptica

Banda	Descrição	Faixa de Comprimento de Onda (nm)
O	Original	1.260 a 1.360
E	Estendida	1.360 a 1.460
S	Curta	1.460 a 1.530
C	Convencional	1.530 a 1.560
L	Longa	1.565 a 1.625
U	Ultra-Longa	1.625 a 1.675

Tabela A-1: Diferentes bandas de comprimento de onda em fibra óptica.

A tabela A-1 [RS02] mostra as diferentes bandas disponíveis para transmissão em fibra óptica monomodo. Os primeiros sistemas WDM usavam a banda *C*, principalmente porque era a região de operação dos EDFAs. Hoje, tem-se EDFAs que trabalham na banda *L*, que permitem que os sistemas WDM usem as bandas *C* e *L*. A *amplificação Raman* complementa o uso dos EDFAs e poderá abrir futuramente outras bandas, tais como a *S* e a *U*, para aplicações WDM.

Na atualidade, os sistemas WDM utilizam a região em torno do comprimento de onda de  $1.55 \mu\text{m}$ , principalmente por dois motivos :

- As perdas intrínsecas da fibra óptica são as menores nessa região ;
- A disponibilidade de amplificadores ópticos de alto desempenho para essa região é alta.

## A.2 Separação de Canais

Um parâmetro de interesse nas redes ópticas encaminhadas por comprimento de onda é a separação de canais, que especifica a separação entre duas frequências ou comprimentos de onda. Esta separação pode ser medida em unidades de frequência ou de comprimento de onda.

Da relação

$$f = \frac{c}{\lambda} \quad (\text{A-1})$$

Tem-se que, diferenciando  $f$  com respeito a  $\lambda$ , e avaliando-se para um comprimento de onda  $\lambda_o$ , obtém-se

$$\Delta f = -\frac{c}{\lambda_o^2} \Delta \lambda \quad (\text{A-2})$$

A Equação (A-2) permite calcular o número de comprimentos de onda que é possível obter a partir de uma projetada separação de canais, conhecendo-se a largura de banda de operação.

Por exemplo, considerando-se uma separação de canais  $\Delta f = 100\text{GHz}$ , em torno de  $\lambda_o = 1.55\mu\text{m}$ , obtém-se uma separação no comprimento de onda  $\Delta \lambda = 0.8\text{nm}$ . No caso da Banda  $L$ , segundo a Tabela A-1, tem-se que para a largura de banda na faixa de (1.565 a 1.625)  $\text{nm}$ , o número de comprimentos  $n$  disponíveis é:

$$n = \frac{(1.625 - 1.565)}{0.8} = 75; \quad (\text{A-3})$$

# Apêndice B

## Programas Desenvolvidos

Na implementação do código fonte dos programas desenvolvidos, utilizou-se o *Matlab 6* [DB03] para ambiente Windows, como ferramenta de programação.

### B.1 Encaminhamento Estático

#### Modelo SP

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Programa Principal      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function resultado = Main(C,D,T);

to = clock;
N = size(C,1);
caminhos = dijkstra_todos(C,D);
caminhos = caminhos(find(caminhos(:,1)>0),:);
[a,b]=size(caminhos);
M=size(D,1);
atendidos= 0;
n_atendidos=0;
k = 1;
while k <= size(caminhos,1)
```

```

verifica = existe_lambda(caminhos(k,:),T);
if verifica == 1
    T=atualiza_topologia(caminhos(k,:),T);
    D=atualiza_demanda(caminhos(k,:),D);
    atendidos = atendidos + 1;
    k = k + 1;
elseif verifica == 0
    C=atualiza_custos(caminhos,C,T);
    caminhos = dijkstra_todos(C,D);
    caminhos = caminhos(find(caminhos(:,1)>0),:);
    if size(caminhos,1) == 0
        break;
    end
    k=1;
end
end
end
caminhos=dijkstra_todos(C,D);
n_atendidos = size(caminhos,1);
demanda_atualizada = D
topologia_atualizada = T
disp(['Atendidos      = ' num2str(atendidos)]);
disp(['Nao Atendidos = ' num2str(n_atendidos)]);
ro = atendidos/(atendidos+n_atendidos);
disp(['Percentagem de Atendimento = ' num2str(ro)]);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula o caminho mínimo para toda demanda %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function caminhos = dijkstra_todos(C,D)

```

```

N = size(C,1);
M =size(D,1);
w=1;
Rotas=[ ];
for i=1:M
    for j=1:M
        while D(i,j)>0
            caminho = dijkstra(C,i,j);
            D(i,j)=D(i,j)-1;
            for k=1:length(caminho)
                Rotas(w,k) = caminho(k);
            end
            w=w+1;
        end
    end
end
D;
caminhos = ordenar(Rotas);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula distância mínima e rota respectiva %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function caminho = dijkstra(C,origem,destino);

N = size(C);
for i=1:N
    for j=1:N
        if C(i,j) == 0

```

```
                C(i,j) = inf;
            end
        end
    end
end
Ni = origem;
for i = 1 : N,
    A(i) = i;
    Rot(i) = 0;
    if i == Ni
        d(i) = 0;
    else
        d(i) = inf;
    end
end
end
F = [ ];
V(1) = Ni;
for t = 1 : N,
    min = inf;
    for i = 1 : length(A),
        if d(A(i)) < min
            r = A(i);
            min = d(A(i));
        end
    end
end
F = [F r];
[a b] = find(A == r);
A = [A(1:b-1) A(b+1:end)];
G = [ ];
for i = 1 : N,
    if C(r, i) < inf
```

```
        G = [G i];
    end
end
V = intersect(A, G);
for i = 1 : length(V),
    x = d(V(i));
    y = d(r) + C(r,V(i));
    [p q] = min([x y]);
    if p < d(V(i))
        d(V(i)) = p;
        Rot(V(i)) = r;
    end
end
end
Nf = destino;
if d(Nf) == inf
    caminho = 0;
    return;
end
Path = [ ];
Path(1) = Nf;
no_at = Nf;    i=1;
while no_at ~= Ni,
    i = i+1;
    Path(i) = Rot(no_at);
    no_at = Rot(no_at);
end
output_rota = [ ];
for i = 1 : length(Path),
    output_rota = [output_rota Path(length(Path)-i+1)];
```

```
end
caminho = output_rota;
custo = num2str(d(Nf));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Ordena a matriz de rotas, em forma ascendente %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function resultado = ordenar(rotas)

N=size(rotas,1);
for i=1:N-1
    for j=i+1:N
        v1=length(rotas(i,find(rotas(i,:)>0)));
        v2=length(rotas(j,find(rotas(j,:)>0)));
        if (v1)>(v2)
            troca = rotas(i,:);
            rotas(i,:) = rotas(j,:);
            rotas(j,:) = troca;
        end
    end
end

resultado=rotas;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Verifica se existe canal (lambda) disponível. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function verifica = existe_lambda(caminhos,T)
```

```

conta=0;
v=caminhos(:,find(caminhos(:,>0)));
for j=1:length(v)-1
    if T(v(j),v(j+1))>0
        conta=conta+1;
    end
end
if conta==length(v)-1
    verifica=1;
else
    verifica = 0;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de topologias. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_topologia(caminhos,T)

v=caminhos(:,find(caminhos(:,>0)));
for j=1:length(v)-1
    T(v(j),v(j+1))=T(v(j),v(j+1))-1;
end
saida = T;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de demandas. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_demanda(caminhos,D)

```

```

M=size(D,1);
[a,b]=size(caminhos);
for k=1:a
    v=caminhos(k,find(caminhos(k,*)>0));
    if D(v(1),v(length(v)))>0
        D(v(1),v(length(v)))= D(v(1),v(length(v)))-1;
    end
end
caminhos;
saida=D;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de custos. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_custos(caminhos,C,T)

N=size(C,1);
a=size(caminhos,1);
M=size(T,1);
for k=1:a
    v=caminhos(k,find(caminhos(k,*)>0));
    for j=1:length(v)-1
        if T(v(j),v(j+1))==0
            C(v(j),v(j+1))=0;
        end
    end
end
saida=C;

```

## B.2 Encaminhamento Dinâmico

### B.2.1 Modelo SP

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Programa Principal- atendimento dinâmico.      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function resultado = Main(C,D,T,tfi)

ti=0;
tm=1;
mi=1/tm;
La=D*mi;
cal = inicial(C,D);
atendidos=0;
n_atendidos=0;
G=[ ];
N=size(C,1);
T_at=T;
C_at=C;
ci_counter=0;
while ti < tfi
    ti=cal(1,1);
    o=cal(1,2);
    d=cal(1,3);
    f=cal(1,4);
    ci_counter=ci_counter + 1;
    ci(ci_counter,1)=size(G,1);
    ci(ci_counter,2)=ti;
    if f==1

```

```

deltalamb=func_expo(La(o,d));
cal=[cal;[[ti+deltalamb] o d 1]];
cal=sortrows(cal,1);
caminho = dijkstra(C_at,o,d);
cal(1,:)=[ ];
if size(caminho,2) > 1
    T_at=atualiza_topologia1(caminho,T_at);
    C_at=atualiza_custos1(caminho,C_at,T_at);
    atendidos = atendidos + 1;
    deltami=func_expo(mi);
    cal=[cal;[[ti+deltami] o d 0]];
    cal=sortrows(cal,1);
    caminho1=completa(caminho,N);
    G=[G;[[ti+deltami] caminho1]];
    G=sortrows(G,1);
else
    n_atendidos = n_atendidos + 1;
end
elseif f==0
    G1=G(1,2:length(find(G(1,:)>0)));
    T_at=atualiza_topologia0(G1,T_at);
    C_at=atualiza_custos0(G1,C_at,T_at);
    G(1,:)=[ ];
    cal(1,:)=[ ];
end
end
topologia_atualizada = T_at
disp(['Atendidos      = ' num2str(atendidos)]);
at=atendidos;
disp(['Nao Atendidos = ' num2str(n_atendidos)]);

```



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function tempo=func_expo(lambda);
```

```
r = unifrnd(0,1);
```

```
tempo = -log(r)/lambda;
```

```
return;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% Calcula distância mínima e rota respectiva %
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function [caminho,custo] = dijkstra(C,origem,destino);
```

```
N = size(C,1);
```

```
for i=1:N
```

```
    for j=1:N
```

```
        if C(i,j) == 0
```

```
            C(i,j) = inf;
```

```
        end
```

```
    end
```

```
end
```

```
Ni = origem;
```

```
for i = 1 : N,
```

```
    A(i) = i;
```

```
    Rot(i) = 0;
```

```
    if i == Ni
```

```
        d(i) = 0;
```

```
    else
```

```
        d(i) = inf;
```

```
end
end
F = [ ];
V(1) = Ni;
for t = 1 : N,
    min = inf;
    for i = 1 : length(A),
        if d(A(i)) < min
            r = A(i);
            min = d(A(i));
        end
    end
    F = [F r];
    [a b] = find(A == r);
    A = [A(1:b-1) A(b+1:end)];
    G = [ ];
    for i = 1 : N,
        if C(r,i) < inf
            G = [G i];
        end
    end
    V = intersect(A, G);
    for i = 1 : length(V),
        x = d(V(i));
        y = d(r) + C(r,V(i));
        [p q] = min([x y]);
        if p < d(V(i))
            d(V(i)) = p;
            Rot(V(i)) = r;
        end
    end
```

```

        end
    end
    Nf = destino;
    if d(Nf) == inf
        caminho = 0;
        return;
    end
    Path = [ ];
    Path(1) = Nf;
    no_at = Nf;    i=1;
    while no_at ~= Ni,
        i = i+1;
        Path(i) = Rot(no_at);
        no_at = Rot(no_at);
    end
    output_rota = [ ];
    for i = 1 : length(Path),
        output_rota = [output_rota Path(length(Path)-i+1) ];
    end
    caminho = output_rota;
    custo = num2str(d(Nf));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz topologia %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_topologia1(caminho,T)

v=caminho;
for j=1:length(v)-1

```

```

    T(v(j),v(j+1))=T(v(j),v(j+1))-1;
end
saida = T;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de custos %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_custos1(caminho,C,T)

v=caminho;
for j=1:length(v)-1
    if T(v(j),v(j+1))==0
        C(v(j),v(j+1))=0;
    end;
end;
saida=C;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gera a matriz de caminhos Completa %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = completa(caminho,n)

saida = [ ];
naux=length(caminho);
if naux ~= n
    aux =zeros(1,n-naux);
    caminho=[caminho aux];
end

```

```
saida=caminho;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz topologia %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_topologia0(caminho,T)

v=caminho;
for j=1:length(v)-1
    T(v(j),v(j+1))=T(v(j),v(j+1))+1;
end
saida = T;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de custos %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_custos0(caminho,C,T)

v=caminho;
for j=1:length(v)-1
    if T(v(j),v(j+1))==1
        C(v(j),v(j+1))=1;
    end
end
saida=C;
```

## B.2.2 Modelo Max-Sum

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
% Principal - para atendimento dinamico. %  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function resultado = Main(C,D,T,tfi)
```

```
t_roda = clock;  
ti=0;  
tm=1;  
mi=1/tm;  
La=D*mi;  
cal = inicial(C,D);  
v = dijkstra_todos(C,D);  
atendidos=0;  
n_atendidos=0;  
G=[ ];  
N=size(C,1);  
T_at=T;  
C_at=C;  
ci_counter=0;  
while ti < tfi  
    ti=cal(1,1);  
    o=cal(1,2);  
    d=cal(1,3);  
    f=cal(1,4);  
    ci_counter=ci_counter + 1;  
    ci(ci_counter,1)=size(G,1);  
    ci(ci_counter,2)=ti;  
    if f==1
```

```
deltalamb=func_expo(La(o,d));
cal=[cal;[[ti+deltalamb] o d 1]];
cal=sortrows(cal,1);
cal(1,:)=[];
if size(v,2)>1
    [v1,valor]=compara(v,o,d);
    if valor == 0
        v2 = max_sum(v,v1,T_at,o,d);
        if size(v2,2) > 1
            T_at=atualiza_topologica1(v2,T_at);
            [C_at,calc1]=atualiza_custos1(v2,C_at,T_at);
            atendidos = atendidos + 1;
            deltami=func_expo(mi);
            cal=[cal;[[ti+deltami] o d 0]];
            cal=sortrows(cal,1);
            v21=completa(v2,N);
            G=[G;[[ti+deltami] v21]];
            G=sortrows(G,1);
            if calc1 == 1
                v = dijkstra_todos(C_at,D);
            end
        elseif size(v2,2) == 1
            n_atendidos = n_atendidos + 1;
        end
    else
        n_atendidos = n_atendidos + 1;
    end
else
    n_atendidos=n_atendidos + 1 ;
end
```

```

elseif f==0
    G1=G(1,2:length(find(G(1,:)>0)));
    T_at=atualiza_topologia0(G1,T_at);
    [C_at,calc0]=atualiza_custos0(G1,C_at,T_at);
    G(1,:)=[ ];
    cal(1,:)=[ ];
    if calc0 == 1
        v = dijkstra_todos(C_at,D);
    end
end
end

end

topologia_atualizada = T_at
disp(['Atendidos      = ' num2str(atendidos)]);
disp(['Nao Atendidos = ' num2str(n_atendidos)]);
ro = atendidos/(atendidos + n_atendidos);
disp(['Percentagem de Atendimento = ' num2str(ro)]);
disp('-----');
disp(['Tempo= ' num2str(etime(clock,t_roda)) ' segundos.']);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Inicializacao do calendario %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = inicial(C,D)

N=size(C,1);
cal=[ ];
tm=1;
mi=1/tm;
La=D*mi;

```

```

saida = [ ];
for i=1:N
    for j=1:N
        if i~=j
            if La(i,j)~=0
                delta = func_expo(La(i,j));
                cal=[cal;[delta i j 1]];
                cal = sortrows(cal,1);
            end
        end
    end
end
saida = cal ;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gera uma demanda aleatoria %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function tempo=func_expo(lambda);
r = unifrnd(0,1);
tempo = -log(r)/lambda;
return;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Encontra o(s) caminho(s) mínimo(s) (máximo 3) %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = dijkstra_todos(C,D)

M =size(D,1);

```

```
caminhos=[ ];
saida=[ ];
for i=1:M
    for j=1:M
        if D(i,j) > 0
            v = dijkstra(C,i,j);
            if size(v,2) > 1
                Caux=C;
                for k=1:length(v)-1
                    Caux(v(k),v(k+1))=0;
                end
                v1 = dijkstra(Caux,i,j);
                for k=1:length(v1)-1
                    Caux(v1(k),v1(k+1))=0;
                end
                v2=dijkstra(Caux,i,j);
                if length(v) == length(v1)
                    t = size(caminhos,1);
                    for m=1:size(v,2)
                        caminhos(t+1,m) = v(1,m);
                        caminhos(t+2,m) = v1(1,m);
                    end
                    if length(v1) == length(v2)
                        for m=1:size(v1,2)
                            caminhos(t+3,m) = v2(1,m);
                        end
                    end
                end
            else
                t = size(caminhos,1);
                for m=1:size(v,2)
```

```
        caminhos(t+1,m) = v(m);
    end
end
end
end
end
end
saida = caminhos;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calcula a distância mínima e rota respeitativa %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [rota,custo] = dijkstra(C,origem,destino);

N = size(C,1);
for i=1:N
    for j=1:N
        if C(i,j) == 0
            C(i,j) = inf;
        end
    end
end

Ni = origem;
for i = 1 : N,
    A(i) = i;
    Rot(i) = 0;
    if i == Ni
        d(i) = 0;
    else
```

```
        d(i) = inf;
    end
end
F = [ ];
V(1) = Ni;
for t = 1 : N,
    min = inf;
    for i = 1 : length(A),
        if d(A(i)) < min
            r = A(i);
            min = d(A(i));
        end
    end
    F = [F r];
    [a b] = find(A == r);
    A = [A(1:b-1) A(b+1:end)];
    G = [ ];
    for i = 1 : N,
        if C(r,i) < inf
            G = [G i];
        end
    end
    V = intersect(A, G);
    for i = 1 : length(V),
        x = d(V(i));
        y = d(r) + C(r,V(i));
        [p q] = min([x y]);
        if p < d(V(i))
            d(V(i)) = p;
            Rot(V(i)) = r;
        end
    end
end
```

```
        end
    end
end
Nf = destino;
if d(Nf) == inf
    rota = 0;
    return;
end
Path = [ ];
Path(1) = Nf;
no_at = Nf;    i=1;
while no_at ~= Ni,
    i = i+1;
    Path(i) = Rot(no_at);
    no_at = Rot(no_at);
end
output_route = [ ];
for i = 1 : length(Path),
    output_route = [output_route Path(length(Path)-i+1) ];
end
rota = output_route;
custo = num2str(d(Nf));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Encontra os possíveis "caminhos mínimos" %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [saida1,saida2] = compara(v,o,d)

saida1 = [ ];
```

```

a = size(v,1);
v1=[ ];
saida2=0;
for k=1:a
    if v(k,1) == o & v(k,length(find(v(k,:)))) == d
        v1=[v1;v(k,1:length(find(v(k,:))))];
    end
end
saida1 = v1;
if size(v1,1) == 0
    saida2 =1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Escolhe o caminho(s) "max-sum" %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = max_sum(v,v1,T,o,d)

a=size(v,1);
b=size(v1,1);
if size(v1,1) == 1
    saida = v1;
    return;
end
matriz = [ ];
saida = [ ];
matriz_s2 = [ ];
matriz_s1 = [ ];
for k=1:b

```

```

T_at2 = atualiza_topologia(v1(k,:),T);
matriz_aux = [ ];
for n=1:a
    if v(n,1) ~= 0 | v(n,length(find(v(n,:)))) ~= d
        lambdas = [ ];
        for j=1:length(find(v(n,:)))- 1
            valor = T_at2(v(n,j),v(n,j+1));
            lambdas = [lambdas valor];
        end
        matriz_aux = [matriz_aux min(lambdas)];
    end
end
for p=1:size(matriz_aux,2)
    matriz(k,p) = matriz_aux(1,p);
end
matriz(k,p+1) = sum(matriz_aux);
matriz_s1 = [matriz_s1; v1(k,:) sum(matriz_aux)];
end
matriz_s = sortrows(matriz_s1,size(matriz_s1,2));
caminho = matriz_s(size(matriz_s,1),1:length(find(matriz_s(2,:)))-1);
saida = caminho;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz topologia %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_topologia(caminho,T)

saida=[ ];
c=caminho;

```

```

T_at2 = T;
for j=1:length(c)-1
    T_at2(c(j),c(j+1))=T_at2(c(j),c(j+1))-1;
end
saida = T_at2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de custos      %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [saida,calcula] = atualiza_custos1(caminho,C_at,T_at)

saida=[ ];
c=caminho;
calcula=0;
for j=1:length(c)-1
    if T_at(c(j),c(j+1))==0
        C_at(c(j),c(j+1))=0;
        calcula=1;
    end
end
saida=C_at;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gera a matriz caminhos Completa  %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = completa(caminho,n)

saida = [ ];

```

```

naux=length(caminho);
if naux ~= n
    aux =zeros(1,n-naux);
    caminho=[caminho aux];
end
saida=caminho;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz topologia %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function saida = atualiza_topologia0(caminho,T_at)

c=caminho;
saida = [ ];
for j=1:length(c)-1
    T_at(c(j),c(j+1))=T_at(c(j),c(j+1))+1;
end
saida = T_at;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Atualiza a matriz de custos %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [saida,calcula] = atualiza_custos0(caminho,C_at,T_at)

saida=[ ];
c=caminho;
calcula=0;
for j=1:length(c)-1

```

```

    if T_at(c(j),c(j+1))==1
        C_at(c(j),c(j+1))=1;
        calcula=1;
    end
end
saida=C_at;

```

### B.2.3 Validação

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gera amostra a partir dos dados de simulação          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function x = amostra(delta, ci)

m = length(ci)
t = ci(1, 2);
ta = 0;
tf = ci(m, 2);
n = fix(tf / delta);
kk = 1;
x = zeros(n, 1);
k = 1;
while kk < n + 1
    while t < delta * kk
        x(kk) = x(kk) + (t - ta) * ci(k, 1);
        ta = t;
        k = k + 1;
        t = ci(k, 2);
    end
end

```

```

    x(kk) = (x(kk) + (kk * delta-ta) * ci(k, 1)) / delta;
    kk = kk + 1;
    ta=delta*(kk-1);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bootstrap. Para determinação do intervalo de confiança %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [cir5, media, cir196, delta] = bootstrap(ci)

m=length(ci);
for k=1:200
    in =fix(m*rand(m,1)+1);
    cir(k)=mean(ci(in));
end
cir = sort(cir);
cir5=cir(5); % valor inferior
media = mean(ci); % valor medio
cir196=cir(196); % valor superior
delta=100*(cir(196) - cir(5))/media; % intervalo de confianza

```

# Bibliografia

- [AMO93] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows*. Prentice Hall, 1993.
- [Bea76] D. Bear. *Principles of Telecommunication - Traffic Engineering*. Peter Peregrinus LTD, 1976.
- [BK95] A. Birman and K. Kershenbaum. Routing and Wavelength Assignment Methods in Single-Hop All-Optical Networks with Blocking. In *Proceedings of the IEEE INFOCOM '95, April, Boston, MA, vol. 2, pp. 432-438, 1995*.
- [BS97] R. A. Barry and S. Subramaniam. The MAX-SUM Wavelength Assignment Algorithm for WDM Ring Networks. In *Proceedings of the OFC '97, 1997*.
- [CGK92] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath Communications: An Approach to High Bandwidth Optical WANs. *IEEE Transactions on Communications, vol.40, no. 7, pp. 1171-1182, 1992*.
- [DB03] D.Hanselman and B.Littlefield. *Matlab 6 : Curso Completo*. Pearson Education do Brasil, Sao Paulo, 2003.
- [ET93] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, 1993.
- [IM99] J. Iness and B. Mukherjee. Sparse Wavelength Conversion in Wavelength-Routed WDM Networks. *Photonic Network Communications, 1999*.
- [Ine97] J. Iness. *Efficient Use of Optical Components in WDM-Based Optical Networks*. PhD thesis, University of California, Davis, 1997.

- [JA96] G. Jeong and E. Ayanoglu. Comparison of Wavelength-Interchanging and Wavelength-Selective Cross-Connects in Multiwavelength All-Optical Networks. In *Proceedings of the IEEE INFOCOM '96, March, San Francisco, CA, vol. 1, pp. 156-163, 1996.*
- [Jun01] R. C. A. A. Junior. Alocação de Rota e Comprimento de Onda em Anéis WDM. Master's thesis, Universidade Estadual de Campinas, 2001.
- [KA] E. Karasan and E. Ayanoglu. Effects of Wavelength Routing and Selection Algorithms on Wavelength Conversion Gain in WDM Optical Networks.
- [KL00] M. Kodialam and T. V. Lakshman. Minimum Interference Routing with Applications to MPLS Traffic Engineering. In *Proceedings of the IEEE INFOCOM '00, March, vol. 2, pp. 884-893, 2000.*
- [Kle75] L. Kleinrock. *Queueing Systems*. John Wiley & Sons, Vol. 1, 1975.
- [LL93] K. C. Lee and V. O. K. Li. A Wavelength-Convertible Optical Network. *IEEE/OSA Journal of Lightwave Technology, May-June, vol. 11, no. 5-6, pp. 962-970, 1993.*
- [MBRM96] B. Mukherjee, D. Banerjee, S. Ramamurthy, and A. Mukherjee. Some Principles for Designing a Wide-Area WDM Network. *IEEE/ACM Transactions on Networking, October, vol. 4, no. 5, 1996.*
- [MG02] C. Siva Ram Murthy and M. Gurusamy. *WDM Optical Networks*. Prentice Hall, New Jersey, 2002.
- [Nas03] L. H. B. Nascimento. Contribuição ao Estudo de Redes Fotônicas de Pacotes. Master's thesis, Universidade Estadual de Campinas, 2003.
- [Rib03] J. A. Justino Ribeiro. *Comunicações Ópticas*. Editora Érica, São Paulo, 2003.
- [RS97] R. Ramaswami and G. H. Sasaki. Multiwavelength Optical Networks with Limited Wavelength Conversion. In *Proceedings of the IEEE INFOCOM '97, Kobe, Japan, pp. 489-498, April, 1997.*

- [RS02] R. Ramaswami and K.N. Sivarajan. *Optical Networks A Practical Perspective*. Morgan Kauffmann Publishers, San Francisco, CA, 2002.
- [SAS96] S. Subramaniam, M. Azizoglu, and A. K. Somani. All-Optical Networks with Sparse Wavelength Conversion. *IEEE/ACM Transactions on Networking, August, vol. 4, pp. 544-557*, 1996.
- [SB97] S. Subramaniam and R. A. Barry. Wavelength Assignment in Fixed Routing WDM Networks. In *Proceedings of the ICC, Montreal, Canada, vol. 1, pp. 406-410*, 1997.
- [TS95] L. G. Tan and M. C. Sinclair. Wavelength Assignment between the Central Nodes of the COST 239 European Optical Network. In *Proceedings of the 11th UK Performance Engineering Workshop, Liverpool, September, pp. 235-247*, 1995.
- [TYM02] M.T. Takahashi, A. Yamakami, and E. Moschim. An algorithm to solve the static RWA problem with wavelength conversion using multicommodity optimization technique. In *Proceedings of the 2002 International Telecommunication Symposium, pp 1244-1247*, Natal, Brazil, 2002.
- [VSK00] K. R. Venugopal, M. Shivakumar, and P. S. Kumar. A Heuristic for Placement of Limited Range Wavelength Converters in All-optical Networks. In *Proceedings of the INFOCOM '99 18th Annual Joint Conference of the IEEE Computer and Communications Societies Networks, OPNET 2000, January, Paris, France, pp 1109-1115.*, 2000.
- [YLES96] J. Yates, J. Lacey, D. Everitt, and M. Summerfield. Limited-Range Wavelength Translation in All-Optical Networks. In *Proceedings of the IEEE INFOCOM '96, March, San Francisco, CA, vol. 3, pp. 954-961*, 1996.
- [ZJM00] H. Zang, J. P. Jue, and B. Mukherjee. A Review of Routing and Wavelength Assignment Approaches for Wavelength-Routed Optical WDM Networks. *Optical Networks Magazine, pp. 47-60*, 2000.