

AUGUSTO CASTELAN CARLSON

**ROTEAMENTO BASEADO EM CAMINHOS COM
PERFIS VARIANTES NO TEMPO**

FLORIANÓPOLIS

2007

UNIVERSIDADE FEDERAL DE SANTA CATARINA

**PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA**

**ROTEAMENTO BASEADO EM CAMINHOS COM
PERFIS VARIANTES NO TEMPO**

Dissertação submetida à
Universidade Federal de Santa Catarina
como parte dos requisitos para a
obtenção do grau de Mestre em Engenharia Elétrica.

AUGUSTO CASTELAN CARLSON

Florianópolis, abril de 2007.

ROTEAMENTO BASEADO EM CAMINHOS COM PERFIS VARIANTES NO TEMPO

Augusto Castelan Carlson

‘Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica, Área de Concentração em *Automação e Sistemas*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.’

Jean-Marie Farines, Dr.
Orientador

Nelson Sadowski, Dr.
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Banca Examinadora:

Jean-Marie Farines, Dr.
Presidente

Eduardo Camponogara, Ph.D.
Co-orientador

Roberto Willrich, Dr.

Roberto Alexandre Dias, Dr.

Edison Tadeu Lopes Melo, Msc.

Joni da Silva Fraga, Dr.

Ao meu pai e minha mãe (Renato e Maria Helena).

AGRADECIMENTOS

Aos orientadores Jean-Marie Farines e Eduardo Camponogara pelas orientações e apoio na realização deste trabalho.

Ao pessoal do LCMI que esteve sempre presente nos momentos de concentração, frustração e diversão.

A minha namorada Emanuela, que me apoiou durante todo o mestrado e compreendeu momentos de pouco atenção dedicados a ela e pela sua revisão em busca de erros de digitação e de português que auxiliaram no melhor entendimento do texto.

Aos meus familiares pelo incentivo em continuar estudando e estar sempre em busca de novos conhecimentos.

Aos membros da banca examinadora pelas sugestões e contribuições para melhoria deste trabalho.

Ao PGEEL que me acolheu como aluno do curso e ao CNPq pelo financiamento com uma bolsa de mestrado.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para obtenção do grau de Mestre em Engenharia Elétrica.

ROTEAMENTO BASEADO EM CAMINHOS COM PERFIS VARIANTES NO TEMPO

Augusto Castelan Carlson

Abril/2007

Orientador: Jean-Marie Farines, Dr.

Área de Concentração: Automação e Sistemas

Palavras-chave: Roteamento, Qualidade de Serviço, Redes de Computadores, Restrições Aditivas, Múltiplas Restrições e Otimização

Número de Páginas: xii + 72

Este trabalho trata do problema de otimização do uso global de recursos em uma rede intra-domínio para situações de demandas de fluxos com perfis predefinidos de banda variantes no tempo. Uma abordagem baseada em caminhos é proposta como alternativa à abordagem baseada em fluxos encontrada na literatura. Utilizou-se a relaxação de um modelo em programação inteira para o caso off-line e proposta uma heurística para o caso off-line e on-line. O modelo e a heurística são baseados em caminhos e utilizam o pré-cômputo desses para obtenção dos resultados. Realiza-se uma comparação entre o modelo baseado em caminhos e um modelo de fluxo em redes. Os desempenhos do roteamento fixo e variável utilizando perfis de banda variantes no tempo também são comparados.

Inicialmente, a formulação e a heurística utilizam como critério apenas a largura de banda. O uso de restrições aditivas no modelo baseado em caminhos é apresentado como uma vantagem em relação ao modelo baseado em fluxo em redes. Em seguida, é apresentada uma extensão para tratar múltiplas restrições de QoS sem efetuar alterações nos algoritmos e heurísticas desenvolvidos. Em contraposição, a abordagem baseada em fluxo necessita de alterações na sua implementação para considerar restrições adicionais, o que torna o problema NP-Difícil.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

PATH-BASED ROUTING WITH TIME-VARYING PROFILES

Augusto Castelan Carlson

April/2007

Advisor: Jean-Marie Farines, Dr.

Area of Concentration: Automation and Systems Engineering

Key words: Routing, Quality of Service, Computer Networks, Additive Constraint, Multiple Constraints and Optimization

Number of Pages: xii + 72

This work is concerned with the optimization of global resource usage in an intra-domain computer network, where flow-request profiles are time-varying and known in advance. A path-based model is proposed as an alternative to an existing network-flow-based model. The off-line routing problem is formulated as an interger linear program and a heuristic is proposed for off-line and on-line applications. Numerical comparisons between the path-based and network-flow-based models are performed. The trade-off between fixed and variable routing using time-varying bandwidth profiles is investigated.

Initially, the formulation and heuristic deal with only bandwidth usage, but later they are augmented to treat additional resources and QoS metrics. The use of additive constraints in the path-based model is presented as an advantage to the network-flow-based model. The proposed path-based approach can handle multiple constraints without modifying the algorithms and heuristics developed to optimize bandwidth usage. On the other hand, the flow-based approaches need profound changes in the procedures with the use of constrained shortest-path algorithms (NP-Hard).

Sumário

1	Introdução	1
1.1	Motivação	2
1.2	Objetivos	2
1.3	Organização do Trabalho	3
2	Roteamento com Qualidade de Serviço	4
2.1	Qualidade de Serviço e Roteamento em Redes de Computadores	4
2.2	A Rede MPLS e a Engenharia de Tráfego	7
2.2.1	A Rede MPLS	7
2.2.2	A Engenharia de Tráfego	8
2.2.3	Modelos e Algoritmos de Roteamento	9
2.3	Roteamento em Redes MPLS com Perfis de Banda Variantes no Tempo	10
2.3.1	Modelo de Fluxo em Redes para Roteamento com Perfis	12
2.3.2	Modelos em Programação Matemática para Roteamento Off-line	14
2.3.3	Heurística para o Roteamento On-line	18
2.3.4	Comentários	19
2.4	Conclusões	20

3	Roteamento Baseado em Caminhos	21
3.1	Algoritmos Baseados em Caminhos	22
3.2	Modelo Proposto em Programação Matemática	22
3.2.1	Modelo para Roteamento Fixo	23
3.2.2	Modelo para Roteamento Variável	24
3.2.3	Heurística	25
3.3	Cômputo de Caminhos	25
3.3.1	Caminhos Arcos-Disjuntos	26
3.3.2	Heurística de Geração de Caminhos Aleatória	27
3.3.3	Algoritmo de Geração de Todos os Caminhos	27
3.4	Conclusões	29
4	Estudo Comparativo Experimental	31
4.1	Condições Experimentais	31
4.2	Influência da Ponderação entre c_{max} e c_{mean}	32
4.3	Roteamento Fixo <i>versus</i> Variável	35
4.4	Impacto da Granularidade do Período de Tempo na Aceitação de Demandas	37
4.5	Análise Experimental de Roteamento Off-line	38
4.5.1	Cenário 1	38
4.5.2	Cenário 2	41
4.5.3	Influência de α no cenário 2	44
4.6	Análise Experimental de Roteamento On-line	45
4.6.1	Cenário 1	46
4.6.2	Cenário 2	49
4.7	Conclusões	50

5	Tratamento de Restrições no Problema de Roteamento	52
5.1	Revisão da Literatura	54
5.2	Modelos Matemáticos com Múltiplas Restrições	55
5.2.1	Modelo CFC Estendido	56
5.2.2	Modelo RM Estendido	58
5.3	Exemplo ilustrativo	60
5.3.1	Cenário	60
5.3.2	Resultados Numéricos	61
5.3.3	Discussão	62
5.4	Conclusões	63
6	Conclusão	65
6.1	Contribuições	66
6.2	Trabalhos Futuros	67

Lista de Figuras

2.1	Encaminhamento MPLS (“circuito”) e encaminhamento por pacotes	8
2.2	Alocação de demandas para perfis crescentes e decrescentes	11
3.1	Grafo utilizado para ilustrar o funcionamento do algoritmo que gera todos os caminhos	29
3.2	Iterações na geração dos caminhos	30
4.1	Rede de 10 nós utilizada para realização dos experimentos	32
4.2	Rede de 30 nós utilizada para realização dos experimentos	33
4.3	Capacidade dos arcos com variação do α	34
4.4	Relação entre c , c_{mean} e c_{max} com α variando	35
4.5	Relação entre c , c_{mean} e c_{max} com α variando - 10 requisições	35
4.6	Relação entre c , c_{mean} e c_{max} com α variando - 100 requisições	36
4.7	Comparativo: roteamento fixo e variável	37
4.8	Diferença percentual entre os modelos CFC e RM com variação do α	44
5.1	Grafo de 6 nós para ensaio com restrição de saltos	61
5.2	Ocupação dos arcos com restrição do número de saltos usando o modelo CFC.	63
5.3	Ocupação dos arcos usando o modelo RM	63

Lista de Tabelas

3.1	Nodo (chave) com arco ligado diretamente a outro nodo	29
4.1	Comparativo: roteamento fixo e variável	37
4.2	Comparação do número de requisições alocadas	38
4.3	Valor do critério c - 15 requisições - cenário 1 off-line	40
4.4	Valor do critério c - 100 requisições - cenário 1 off-line	40
4.5	Valor do critério c - 400 requisições - cenário 1 off-line	40
4.6	Valor do critério c - 600 requisições - cenário 1 off-line	41
4.7	Valor do critério c - 15 requisições - cenário 2 off-line	42
4.8	Valor do critério c - 100 requisições - cenário 2 off-line	42
4.9	Valor do critério c - 400 requisições - cenário 2 off-line	43
4.10	Valor do critério c - 600 requisições - cenário 2 off-line	43
4.11	Valor do critério c - 100 requisições - $\alpha = 0,0$	45
4.12	Valor do critério c - 100 requisições - $\alpha = 0,25$	45
4.13	Valor do critério c - 100 requisições - $\alpha = 0,75$	46
4.14	Valor do critério c - 100 requisições - $\alpha = 1,0$	46
4.15	Valor do critério c - 20 requisições - cenário 1 on-line	48
4.16	Valor do critério c - 100 requisições - cenário 1 on-line	48
4.17	Valor do critério c - 400 requisições - cenário 1 on-line	48

4.18	Valor do critério c - 600 requisições - cenário 1 on-line	49
4.19	Valor do critério c - 20 requisições - cenário 2 on-line	50
4.20	Valor do critério c - 100 requisições - cenário 2 on-line	50
4.21	Valor do critério c - 400 requisições - cenário 2 on-line	51
4.22	Valor do critério c - 600 requisições - cenário 2 on-line	51
5.1	Caminhos utilizados para solução dos modelos	62

Capítulo 1

Introdução

Para prover Qualidade de Serviço (QoS) na camada de rede, faz-se necessário permitir o roteamento de fluxos que levam em conta a QoS. No roteamento com QoS, um roteador considera diferentes métricas durante a seleção dos caminhos para depois realizar o encaminhamento dos fluxos [28].

A questão de encaminhamento de fluxos continua sendo um tema bastante pesquisado, principalmente quando relacionado à oferta de QoS. O presente trabalho considera o problema de roteamento de fluxos quando estes apresentam perfis variantes no tempo. Este problema é restrito a um mesmo domínio de uma rede MPLS (*Multiprotocol Label Switching*) [2]. No assunto a ser tratado, cada fluxo a ser encaminhado em um caminho virtual LSP (*Label Switched Path*) da rede MPLS necessita de uma reserva de banda que varia durante o dia, segundo um perfil previamente conhecido. Estes perfis são, eventualmente, fornecidos ao provedor pelo usuário (SLA: *Service Level Agreement*) a partir da previsão ou da estimativa das suas necessidades durante o dia.

A solução proposta parte do pressuposto que, durante o dia, para cada período, o roteamento é feito com LSPs fixos e que a demanda de banda varia no tempo a cada período segundo perfis estimados previamente. Um algoritmo simples realiza a seleção ótima de LSPs. Esse algoritmo foi formulado a partir de um modelo em Programação Linear Inteira (PLI) baseado em caminhos.

Um modelo PLI com base nas equações de fluxo em redes foi proposto por Ricciato e Monaco (RM) [30]. Além de uma heurística inspirada no algoritmo de *Dijkstra*. A forma como esse modelo é proposto impossibilita o tratamento de restrições adicionais, tais como atraso de transmissão e número de saltos. A natureza do modelo torna inviável a solução do modelo inteiro, mesmo para pequenas instâncias.

Propõe-se um modelo em programação matemática para roteamento baseado em conjuntos de caminhos, entre cada par origem-destino, tratando, de forma implícita, de restrições aditivas e até mesmo estocásticas, enquanto permite controlar a qualidade da solução e o tempo computacional. O cômputo prévio de caminhos é um dos fatores que permitem obter um ganho no tempo computacional. Apesar da solução ótima de instâncias de maior porte, o tempo computacional ainda é excessivo para roteamento em tempo-real. Neste sentido, propõe-se uma heurística que preserva as propriedades de tratamento de restrições adicionais. Apresentam-se resultados de experimentos computacionais.

1.1 Motivação

A solução do roteamento para um problema com diversas restrições é definida como NP-Completo para o pior caso [24]. Diferentes heurísticas propostas na literatura não são capazes de tratar de problemas com diversas restrições ou conseguem tratar somente de um caso específico [23].

Ricciato e Monaco [30] propuseram um modelo em Programação Linear Inteira (PLI) alicerçado nas equações de fluxo em redes para tratar o problema de roteamento com perfis de banda variantes no tempo em redes com suporte a MPLS. Neste modelo, consideram-se todos os arcos da rede para a obtenção da solução. A forma como o modelo é proposto impossibilita o tratamento de restrições adicionais, como atraso de transmissão e número máximo de saltos. Apesar dos resultados obtidos, a adição de restrições adicionais tornaria o problema ainda mais complexo para ser tratado com este modelo.

A proposição de um modelo que trata do mesmo tipo de problema, mas baseado em caminhos, possibilita a adição de restrições adicionais sem aumentar a complexidade do modelo. Como o tratamento das restrições adicionais é realizado no cômputo dos caminhos, o modelo para solucionar o encaminhamento de fluxos não necessita ser modificado, à medida que são introduzidas novas restrições. Além disso, o modelo baseado em caminhos possibilita uma redução no custo e possui um maior controle sobre o tempo computacional necessário para obter a solução do problema.

1.2 Objetivos

Alocar, durante o dia, as demandas de fluxo com perfis variantes no tempo previstas *a priori* pelos usuários, tendo como critério a economia de banda dos enlaces da rede em contraposição a uma situação de reserva para estes, baseada no pico de banda consiste no principal objetivo deste trabalho.

Além disso, desenvolver modelos abrangentes (off-line e on-line) para encaminhamento de fluxos em redes de computadores que contemplem perfis de demanda variantes no tempo, restrições aditivas e cujo tempo computacional e qualidade da solução possam ser controlados. Este controle se torna viável devido à possibilidade de definir o número de caminhos adicionados ao modelo.

Dentre os objetivos específicos deste trabalho podemos citar:

- Apresentação do problema de roteamento em rede MPLS
- Desenvolvimento de modelos em programação matemática;
- Projeto e implementação de heurísticas para a solução do problema proposto; e
- Análise numérica.

1.3 Organização do Trabalho

No capítulo 2, apresentam-se informações sobre roteamento e QoS em redes de computadores, uma visão geral de MPLS e engenharia de tráfego. Em seguida, trabalhos que tratam do encaminhamento de fluxos são introduzidos. Posteriormente, demonstram-se trabalhos que consideram perfis de banda seguidos do trabalho de Ricciato e Monaco [30], utilizado como base na realização desta dissertação.

O capítulo 3 apresenta a proposta deste trabalho, que é o encaminhamento de fluxos com perfis variantes no tempo baseado em caminhos pré-computados. Também inclui a formulação matemática, heurística e informações sobre o cômputo de caminhos.

No capítulo 4, expõem-se os cenários e condições utilizadas nos experimentos, relação entre os objetivos da formulação matemática e um exemplo que compara o roteamento fixo *versus* o variável. Em seguida, um estudo dos limiares de alocação de demandas e o impacto da granularidade do período de tempo. Finalmente, os cenários e a análise numérica dos experimentos para o roteamento off-line e on-line.

Uma revisão da literatura sobre tratamento de múltiplas restrições e uma exemplificação de sua utilização no modelo proposto neste trabalho são dispostos no capítulo 5.

Por fim, no capítulo 6, apresentam-se a conclusão e sugestões de trabalhos futuros.

Capítulo 2

Roteamento com Qualidade de Serviço

Neste capítulo, faz-se uma revisão de literatura, incluindo tópicos como QoS, roteamento em redes de computadores e engenharia de tráfego, além de questões relacionadas a roteamento. Em seguida, será abordado o roteamento com perfis variantes no tempo e uma solução existente na literatura que servirá de base de comparação para a proposta feita neste trabalho.

2.1 Qualidade de Serviço e Roteamento em Redes de Computadores

Para que dois *hosts* em redes distintas se comuniquem, necessita-se obter um caminho para que os dados sejam encaminhados. Os algoritmos de roteamento têm a função de determinar estes caminhos e encaminhar os pacotes para o destino correto. Estes algoritmos são implementados nos roteadores [19].

O cálculo de rotas segue duas abordagens que definem dois tipos de problema: on-line e off-line [30]. Quando as demandas são admitidas pelo roteador sem conhecimento prévio das requisições seguintes, podendo ou não serem novamente alocadas, caracteriza-se um problema on-line. Estes cálculos são realizados de forma off-line quando o conjunto de demandas é conhecido previamente e todas as rotas são calculadas conjuntamente.

A forma como as redes são estruturadas faz com que o roteamento seja dividido em intra-domínio e inter-domínio. Em um mesmo domínio administrativo, tem-se o roteamento intra-domínio, enquanto o roteamento que envolve vários domínios administrativos (ou sistema autônomo) é chamado de inter-domínio.

O protocolo intra-domínio pode ser do tipo *estado de enlace* ou *vetor de distância*. No tipo de protocolo *estado de enlace* todo roteador pertencente a uma mesma área possui um panorama da topologia de rede. Esses roteadores trocam informações periodicamente (que traz um custo devido ao uso de banda) a fim de manter as informações da topologia atualizadas e o processo de decisão é baseado no algoritmo de caminhos mínimos de *Dijkstra* [23]. O custo dos enlaces é utilizado como métrica de roteamento. Os protocolos do tipo *vetor de distância* não possuem conhecimento de toda a topologia da rede e a obtenção do melhor caminho é baseada no algoritmo de roteamento de Bellman-Ford, que consiste na escolha do caminho mais curto. O protocolo de roteamento mais utilizado, atualmente, para roteamento intra-domínio é o *Open Shortest Path First* (OSPF), que é do tipo *estado de enlace* [23], e tem sua decisão baseada no algoritmo de caminhos mínimos de *Dijkstra* [8]. O principal problema dos protocolos de roteamento intra-domínio é a inconsistência das informações sobre o estado da rede. Esta inconsistência está relacionada aos métodos utilizados para distribuição e agregação das informações de roteamento [23].

O protocolo *Border Gateway Protocol* (BGP)[14] é utilizado para o roteamento inter-domínios. Esse protocolo mantém uma tabela de redes IP que são alcançáveis através de sistemas autônomos. Para manter a tabela atualizada, há troca de informações entre os sistemas autônomos. Um sistema autônomo possui redes que pertencem ao mesmo domínio administrativo, por exemplo, um campus universitário. As decisões do BGP são baseadas nos caminhos e em políticas da rede. Entre os problemas do BGP está o tempo necessário para se recuperar de uma falha de uma rota ou enlace e não permitir um controle sobre o balanceamento do tráfego. Somente informa a melhor rota para certo destino fazendo com que outros caminhos alternativos deixem de ser conhecidos.

Nos últimos anos, com a exigência de desempenho por parte das aplicações, o atendimento à demanda por Qualidade de Serviço (QoS) se tornou uma necessidade [38]. QoS é a habilidade de dispositivos como roteadores, ou de aplicações, de oferecer garantia de que as exigências do serviço requisitado possam ser satisfeitas. Esta garantia pode ser fornecida através de mecanismos que permitem prover diferentes níveis de prioridade ou reserva de recurso para garantir certo nível de desempenho para os fluxos de dados de acordo com a necessidade da aplicação.

O Serviço Integrado é um sistema desenvolvido que fornece garantia de QoS para aplicações a partir da reserva de recurso e o conhecimento do estado ao longo da rede (nos roteadores, por exemplo) [19]. Serviços Integrados exigem que a reserva de recursos seja solicitada explicitamente através de uma sinalização para cada aplicação. Para realizar a sinalização, deve ser utilizado um protocolo como o RSVP [19]. Os roteadores mantêm as informações de uso dos recursos. Caso não haja recursos suficientes para atender uma solicitação, esta é recusada. Duas classes de serviço são

oferecidas pelo Serviço Integrado, o garantido e o de carga controlada. O primeiro garante limite de atraso nas filas dos roteadores e o segundo oferece um serviço similar ao oferecido por uma rede com pouca carga. Uma limitação deste tipo de serviço é a escalabilidade.

O modelo de Serviços Diferenciados surgiu com objetivo de oferecer uma arquitetura escalável e flexível com a habilidade de lidar com várias classes de serviço com diferentes níveis de qualidade. Este modelo de serviço possibilita definir a prioridade de um fluxo de dados baseado na marcação do campo DSCP do cabeçalho do datagrama IP, provendo QoS unidirecional. Os roteadores de borda marcam os valores no campo DSCP e os roteadores de núcleo encaminham os fluxos baseados no valor desse campo.

Atualmente, os algoritmos de roteamento buscam minimizar métricas como número de saltos entre origem-destino [9]. No que diz respeito ao roteamento, os protocolos tradicionais utilizam, geralmente, mecanismos de QoS baseados em uma única métrica, além de não permitir a definição de caminhos explicitamente e de realizar o cálculo das rotas em cada roteador. No entanto, para melhor suportar os requisitos exigidos para oferecer QoS, novos protocolos necessitariam de um mecanismo mais complexo que suportasse múltiplas restrições. Os protocolos utilizados atualmente não possuem suporte a múltiplas restrições e tornam limitado o roteamento para garantia de QoS, mesmo utilizando arquiteturas como Serviços Integrados [6] e Serviços Diferenciados [5].

Protocolos de roteamento podem ter um grande impacto na escalabilidade da rede devido ao processamento necessário para computar as rotas [36]. A frequência com que os caminhos são computados e a complexidade do algoritmo podem causar uma sobrecarga no processamento. Além disso, o aumento de restrições a serem satisfeitas pelo algoritmo faz com que este se torne mais complexo e necessite de mais recurso computacional.

O roteamento com QoS, diferentemente do roteamento convencional, considera várias métricas no cálculo de seleção da melhor rota [28]. As métricas mais comuns relacionadas à QoS são: atraso, variação do atraso, largura de banda, perda de pacotes e número de saltos. Os requisitos ainda podem ser especificados em termos da qualidade do vídeo ou áudio e tempo de resposta do usuário e mapeados em métricas de QoS [28]. Além de selecionar um caminho para cada conexão, o roteamento com QoS também tem como objetivo otimizar o uso da rede balanceando sua utilização [20]. Segundo Masip-Bruin [23], para oferecer roteamento com QoS, necessita-se implementar protocolos de roteamento baseados nas métricas de QoS que devem ser consideradas na obtenção dos caminhos.

A base da arquitetura do roteamento com QoS é formada pelo algoritmo de roteamento com QoS e pelo protocolo de roteamento. O protocolo de roteamento fornece para cada nodo da rede

informações sobre a topologia e em alguns casos informações sobre os recursos disponíveis nos enlaces. O algoritmo de roteamento considera o estado na rede em um determinado instante e computa o caminho entre uma origem e um destino sujeito a alguma restrição. Masip-Bruin [23] lista diversos pontos que ainda necessitam evoluir: roteamento hierárquico, gerenciamento da justiça do roteamento com QoS e necessidade de protótipos para ir além da simulação.

Como alternativa ao roteamento convencional tem-se o *Multiprotocol Label Switching* (MPLS), que permite a definição de caminhos explicitamente e que realiza o cálculo da rota uma única vez. Como a tomada de decisão envolvida não é baseada no caminho mais curto, ele permite uma melhor utilização dos recursos.

2.2 A Rede MPLS e a Engenharia de Tráfego

Com o crescimento da Internet, ampliou-se a importância do gerenciamento da rede para que haja uma melhor distribuição dos recursos, além do que, o aumento da largura de banda das redes não é uma solução aceitável para os problemas de congestionamento [3]. Com o objetivo de fornecer serviços em suas redes sem a necessidade de aumentar excessivamente a largura de banda, provedores passaram a se interessar pela Engenharia de Tráfego (ET). De acordo com Awduche [2], a Engenharia de Tráfego foi viabilizada pela aplicação das redes MPLS. A seção seguinte apresenta uma breve explicação sobre redes MPLS.

2.2.1 A Rede MPLS

A rede MPLS tem como objetivo possibilitar um roteamento escalável e flexível com desempenho, QoS e gerência do tráfego similar ao oferecido pela camada 2 (modelo de referência OSI) [35]. Esta arquitetura utiliza um esquema de encaminhamento no qual em cada pacote é encapsulado um rótulo entre os cabeçalhos da camada 2 e 3. Este rótulo indica qual o destino do pacote.

Baseado em rótulos, este protocolo possibilita a criação de um caminho virtual entre origem e destino chamado *Label Switched Path* (LSP). A criação dos LSPs é feita por um protocolo que também é responsável pela distribuição dos rótulos. Utilizando o LSP como uma conexão é possível usufruir das mesmas vantagens oferecidas por uma rede orientada a conexão.

Ao ingressar em um domínio MPLS, o rótulo é inserido no pacote pelo roteador de borda. Somente neste momento é feita uma análise do cabeçalho IP. Os roteadores de núcleo inspecionam o

rótulo do pacote, buscam na tabela de encaminhamento – *Label Information Base (LIB)* – o destino desse e encaminham-no.

O tráfego de aplicações pertencentes à mesma classe de serviço são agregados formando associação com uma *Forward Equivalence Class (FEC)*, sendo encaminhados em um mesmo LSP. Uma FEC pode ser definida de várias formas: fluxos com o mesmo destino, fluxo da mesma aplicação ou com o mesmo valor no campo *Type of Service (ToS)* do cabeçalho IP.

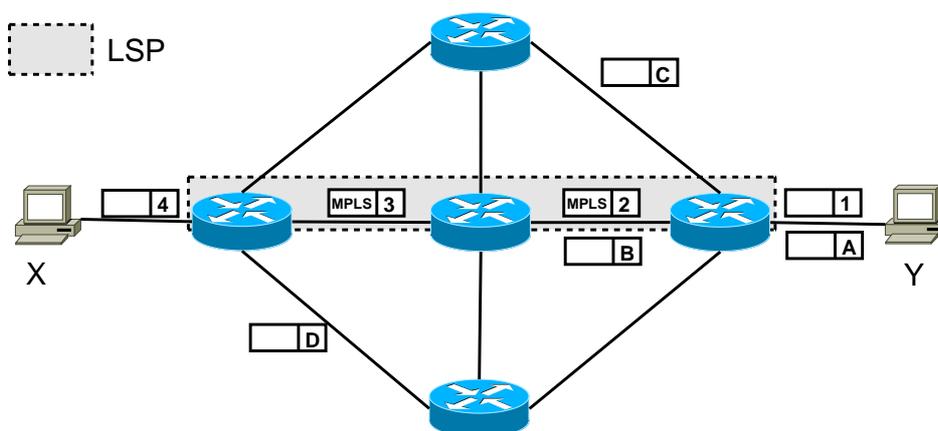


Figura 2.1: Encaminhação MPLS (“circuito”) e encaminhação por pacotes

A figura 2.1 ilustra a diferença dos encaminhamentos MPLS e por pacotes. Para o encaminhamento MPLS é criado o caminho virtual (LSP) no qual todos os pacotes seguem o mesmo caminho, como acontece na figura para os pacotes 1, 2, 3 e 4. Neste caso, os caminhos podem ser explícitos. No encaminhamento de pacotes usando roteamento tradicional, os pacotes podem seguir por caminhos diferentes, como é ilustrado para os pacotes etiquetados com as letras B, C e D, necessitando sequenciamento no destino.

2.2.2 A Engenharia de Tráfego

Protocolos de roteamento convencionais encaminham os fluxos pelo caminho de menor custo. No entanto, com o crescimento da Internet, surgiu a necessidade de melhor aproveitar os recursos da rede. Com a ação da Engenharia de Tráfego, possibilita-se utilizar caminhos diferentes do de menor custo selecionado pelo roteamento convencional para um fluxo e atender às demandas de forma a respeitar os requisitos exigidos.

Com a aplicação de engenharia de tráfego, pode-se reduzir o congestionamento através do balanceamento da carga dos enlaces e otimizar a utilização dos recursos da rede. Segundo Magalhães e Cardoso [22] “engenharia de tráfego (ET) é a tarefa de realizar o mapeamento dos fluxos de tráfego

em uma infra-estrutura física de transporte, de modo a atender os critérios definidos pela operação da rede”.

Os principais objetivos da engenharia de tráfego são [3]:

- Reduzir o congestionamento;
- Aprimorar a utilização da rede; e
- Satisfazer diferentes requisitos conduzindo a um crescimento do rendimento.

Problemas na oferta de QoS estão relacionados à carência e má distribuição dos recursos [22]. A ocorrência de congestionamento pela carência de recursos tem como única solução a atualização da infra-estrutura da rede com a adição de recursos. Entretanto, o caso de congestionamento por má distribuição de recursos caracteriza um problema de ET [22].

2.2.3 Modelos e Algoritmos de Roteamento

Diversos trabalhos apresentam soluções para roteamento baseados em heurísticas. Alguns deles são apresentados nesta seção.

Dias *et al.* [10] propuseram um modelo em programação inteira para implementação de Engenharia de Tráfego (ET) Dinâmica. Esta solução buscou tornar mais eficiente a utilização dos recursos da rede, gerando o menor impacto possível na infra-estrutura já existente. Através de relaxações, os problemas de ET dinâmica foram solucionados, servindo de suporte para o desenvolvimento de um algoritmo de roteamento heurístico. Este algoritmo possibilitou a implementação de ET dinâmica em redes IP sobre tecnologia MPLS. Além disso, um controle de admissão foi incluído no modelo, permitindo a implementação de uma estratégia de diferenciação de serviço. Também foi considerado no trabalho a capacidade de transmissão do canal e o atraso máximo fim-a-fim, mas se limitando à inclusão de novas métricas. Adicionalmente, a fim de selecionar as rotas para a definição dos LSPs utilizou-se o algoritmo de caminhos mínimos de *Dijkstra*.

Banerjee e Sidhu [3] consideraram em seus trabalhos o cômputo de caminhos através de ET para requisições tendo como requisitos largura de banda e atraso. Eles propuseram dois algoritmos on-line (TE-B e TE-DB) e compararam os resultados obtidos com soluções já existentes. Os algoritmos utilizados na comparação são MIRA [16], LCKS [3], WIDEST-SHORT[21], MIN-DELAY [37] e TAMCRA [26]. As duas heurísticas se aplicam em problemas de ET tendo como métrica a largura

de banda (TE-B) e, também, largura de banda e atraso (TE-DB), ambos com objetivo de aumentar o rendimento limitando custo e distribuindo a carga da rede. Consideraram a chegada aleatória de requisições realizando uma comparação estocástica. Em relação às outras soluções existentes, conseguem obter redução do bloqueio de requisições, custos de operação da rede e distribuição da carga. Da mesma forma que em [30] e [9], as heurísticas se limitam à largura de banda e do atraso, não sendo possível adicionar outras restrições.

Wang e Crowcroft [37] apresentam o algoritmo de roteamento MIN-DELAY, baseado em largura de banda e atraso, removendo os enlaces que não satisfazem os requisitos de largura de banda e em seguida buscando no grafo reduzido o menor caminho em relação ao atraso. Esta abordagem de eliminar os enlaces que não atendem aos requisitos de largura de banda também foi utilizada pela heurística de Ricciato e Monaco [30] que será apresentada na seção 2.3.

O leitor pode consultar Mieghem e Kuipers [24] que apresentam um algoritmo exato de roteamento para múltiplas restrições e Pragyansmita e Raghavan [28] para uma discussão de diversos algoritmos para roteamento.

O tema desta dissertação é o problema de roteamento com perfis de banda variantes no tempo e é tratado tendo como base o estudo de Ricciato e Monaco [30]. A seção seguinte apresenta o tema e os detalhes do trabalho de Ricciato e Monaco [30].

2.3 Roteamento em Redes MPLS com Perfis de Banda Variantes no Tempo

Quando o perfil de comportamento do tráfego é conhecido *a priori* ou não, e este varia em intervalos de tempo durante um determinado período, têm-se perfis de banda variantes no tempo. Conhecendo-se esse perfil de banda variante no tempo, pode-se planejar o encaminhamento de forma diferenciada.

Ricciato e Monaco [30] propuseram uma solução para o problema de garantia de roteamento de fluxos com perfis variantes no tempo em redes MPLS. Da mesma forma que Ricciato e Monaco [30], foi assumido neste trabalho que cada demanda é roteada em um LSP dedicado. Neste tipo de roteamento, a reserva de banda para cada LSP é realizada com base em um perfil previamente conhecido, ou com base em um contrato de acordo de nível de serviço (SLA). Este perfil varia com o tempo e a granularidade pode ser definida pelo usuário ou pelo administrador.

A figura 2.2 apresenta dois gráficos, cada um com um perfil de tráfego que corresponde a uma demanda para um mesmo enlace com capacidade de 6 unidades. O perfil de banda no gráfico da parte superior da imagem é crescente (requisição 1) e o da parte inferior e decrescente (requisição 2). Destes dois gráficos, conclui-se que o máximo de demanda para o dia todo para os 2 fluxos é de 6 unidades e que qualquer algoritmo que leve em conta apenas os picos de demanda de cada fluxo implicaria um desperdício de banda.

Ambas as requisições têm duração de vinte e quatro horas. Estas 24 horas foram divididas em seis intervalos de tempo de 4 horas. Pode-se observar que, em cada intervalo, fluxos com diferentes perfis são alocados para usufruir do recurso total do enlace que é de 6 unidades (somando o uso das duas requisições). Se a reserva de banda não for baseada no perfil variante no tempo, quando feita a reserva da requisição 1, por exemplo, baseada no pico (6 unidades), a requisição 2 não poderia ser alocada. Como o pico da requisição 1 é de 6 unidades, este valor seria alocado para todo o tempo de duração da requisição 1, mesmo sem utilizá-lo. Quando utilizada a abordagem baseada nos perfis variantes no tempo, ambas as requisições podem ser alocadas, pois a soma das demandas de seus perfis em cada período de tempo não ultrapassa o valor da capacidade do enlace em nenhum momento do dia.

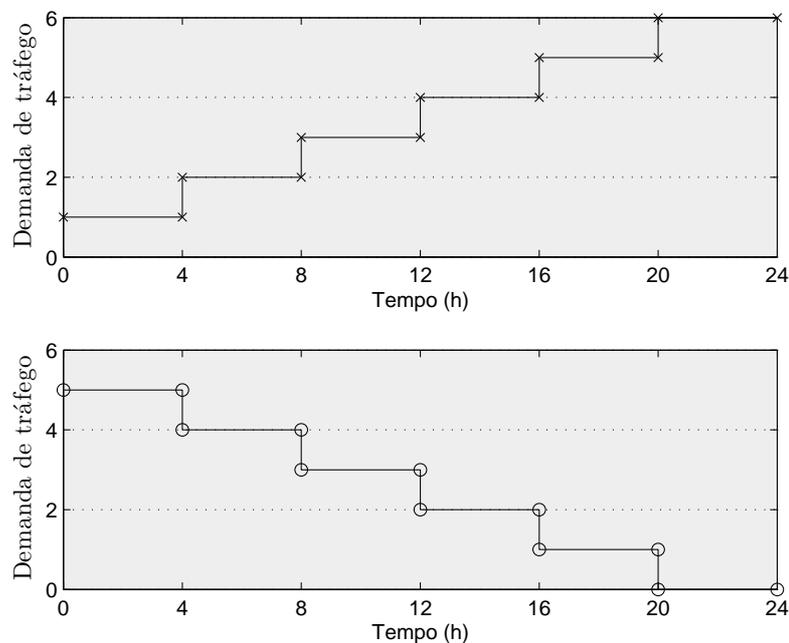


Figura 2.2: Alocação de demandas para perfis crescentes e decrescentes

Ricciato *et al.* [31] apresentam uma formulação em programação linear inteira mista para solucionar o problema de seleção de LSPs baseado no conhecimento prévio da dinâmica do tráfego

em uma rede nos vários períodos do dia. Este conhecimento prévio é obtido através de medições realizadas no passado. Como a solução para redes com certo número de nós e requisições se torna inviável devido ao tempo necessário para se obter o resultado, propuseram uma solução baseada em heurística. Consideraram o caminho mais curto como base para definir o roteamento on-line através desta heurística.

Suri *et al.* [34] apresentam um algoritmo on-line para roteamento com garantia de banda. O algoritmo realiza um pré-processamento para definir a alocação de banda para os enlaces da rede. Em seguida, na fase on-line utiliza um algoritmo para obtenção do menor caminho junto às informações fornecidas pelo pré-processamento. Baseia-se em perfis mensurados no passado ou em um SLA. Apresentam-se algumas limitações de algoritmos já existentes.

2.3.1 Modelo de Fluxo em Redes para Roteamento com Perfis

Ricciato e Monaco [30] consideram o problema de encaminhamento de fluxos de dados em redes com arquitetura MPLS, sendo conhecidas *a priori* as demandas de banda durante os diferentes períodos do dia (perfis de banda). Os diversos perfis de banda podem ser explicitamente declarados pelos usuários e inseridos em acordos de níveis de serviço (SLAs). Apesar das dificuldades de estimação dos perfis por parte dos usuários, o provedor pode tirar vantagem destes acordos na proposição de tarifação diferenciada buscando economizar os recursos. Também é possível que o usuário informe o pico em vez de perfis, no entanto, sendo mais custoso como visto anteriormente.

No sentido de viabilizar esta proposta, os autores acima propuseram modelos em programação matemática inteira para encaminhamento de fluxos, desenvolveram heurísticas e realizaram análises numéricas. O objetivo visado pelos autores consiste em selecionar, de forma otimizada, os LSPs fixos, mantendo um uso equilibrado dos recursos de tal forma que possam ser alocados a um mesmo enlace fluxos com perfis crescentes e decrescentes de demanda de usuários para os mesmos intervalos de tempo.

A rede de comunicação é representada por um grafo direcionado $G = (V, E)$ cujos nós (V) correspondem aos roteadores e cujos arcos (E) representam os enlaces. As demandas são realizadas ciclicamente em intervalos de tempo predefinidos como, por exemplo, um dia. Cada dia é dividido em um conjunto $\Theta = \{1, \dots, \theta\}$ de intervalos de tempo como horas ou minutos.

Cada enlace (i, j) tem uma capacidade C_{ij} de transmissão de dados em Mbps. $\mathcal{K} = \{1, \dots, K\}$ é o conjunto de requisições de serviço admitidas pelo provedor, tendo cada requisição k um nó origem

s_k , um nó destino d_k e um perfil de banda $f^k(\tau)$ correspondendo à demanda de banda durante o período $\tau \in \Theta$.

Uma utilização mais eficiente dos recursos pode ser obtida a partir dos perfis, conforme é ilustrado a seguir. Considerando que $f_{ij}^k(\tau)$ é a demanda de banda da requisição k através do enlace (i, j) e durante o intervalo τ . Com o conhecimento dos perfis, uma capacidade equivalente a $\max_{\tau \in \Theta} \sum_{k \in \mathcal{X}} f_{ij}^k(\tau)$ é suficiente para atender ao conjunto das requisições. Por outro lado, sem informações sobre os perfis, prover recursos para atendimento dos picos de demanda requer a reserva de capacidade equivalente a $\sum_{k \in \mathcal{X}} \max_{\tau \in \Theta} f_{ij}^k(\tau)$.

Além da compensação em cada intervalo entre os perfis crescentes e decrescentes da largura de banda solicitada, a economia depende ainda da habilidade do algoritmo de roteamento em encaixar esses perfis de forma balanceada em um mesmo enlace. Na figura 2.2 o enlace com capacidade de 6 unidades é ocupado por completo por duas demandas de perfis diferentes. Uma vez que a demanda por banda se mantém constante para um intervalo de tempo, a discretização do tempo se torna benéfica para alocar de forma equilibrada as demandas em um mesmo enlace e assegurar uma reserva de largura de banda. Essa discretização permite que um operador defina intervalos comuns para todas as requisições. Durante um intervalo de tempo τ , uma requisição k é associada ao valor de largura de banda $f^k(\tau)$, ou seja, existe uma sincronia nas mudanças da utilização da largura de banda.

O potencial de economia de recursos pode ser quantificado como a diferença destas capacidades [30]:

$$\sum_{k \in \mathcal{X}} \max_{\tau \in \Theta} f_{ij}^k(\tau) - \max_{\tau \in \Theta} \sum_{k \in \mathcal{X}} f_{ij}^k(\tau) \quad (2.1)$$

Para ilustrar a economia de recursos pela diferença das capacidades apresentadas pela equação (2.1), utilizou-se os dados da figura 2.2. Com a primeira parcela da equação obtém-se o pico de banda do enlace e com a segunda parcela obtém-se a capacidade suficiente para atender às requisições. Para requisição 1 tem-se que o pico de banda para todo τ é de 6 unidades e para a requisição 2 o pico de banda para todo τ é de 5 unidades como mostra a equação: $\sum_{k \in \mathcal{X}} \max_{\tau \in \Theta} f_{ij}^k(\tau) = \max\{f_{ij}^1(\tau) : \tau \in \Theta\} + \max\{f_{ij}^2(\tau) : \tau \in \Theta\} = 6 + 5 = 11$.

Então, com a primeira parcela da equação (pico máximo) obtém-se o valor de 11 unidades. Para a segunda parcela da equação, o valor máximo da demanda para as requisições 1 e 2 é de 6 unidades como mostra a equação: $\max_{\tau \in \Theta} \sum_{k \in \mathcal{X}} f_{ij}^k(\tau) = \max_{\tau \in \Theta} \{f_{ij}^1(\tau) + f_{ij}^2(\tau)\} = \max\{f_{ij}^1(1) + f_{ij}^2(1), f_{ij}^1(2) + f_{ij}^2(2), f_{ij}^1(3) + f_{ij}^2(3), f_{ij}^1(4) + f_{ij}^2(4), f_{ij}^1(5) + f_{ij}^2(5), f_{ij}^1(6) + f_{ij}^2(6)\} = 6$. Portanto, o

potencial de economia usando perfis variantes no tempo é de 5 unidades. É importante notar ainda que utilizando o pico de banda como parâmetro para reserva do enlace, a requisição 2 não poderia ser alocada.

Diversos trabalhos que tratam de garantia de largura de banda para roteamento on-line são encontrados na literatura [13, 16, 29, 34], mas somente o trabalho de Ricciato e Monaco [30] considera o tráfego com perfis de banda variantes no tempo. Trabalhos que tratam do roteamento off-line com perfis variantes no tempo também são encontrados [11, 31]. Em [31], considera-se uma formulação em programação linear inteira mista, possibilitando o balanceamento dos recursos, além de propor heurísticas. Em [11], o foco é a otimização dos protocolos de roteamento OSPF e IS-IS (Intermediate System-Intermediate System) para tornar mais eficiente o uso dos recursos. Ambos os trabalhos citados acima consideram o conhecimento prévio das demandas de tráfego. O foco de Ricciato e Monaco [30], base deste trabalho, é na solução do problema on-line e na sua comparação com a solução do problema off-line.

2.3.2 Modelos em Programação Matemática para Roteamento Off-line

Para avaliar a qualidade da solução heurística, Ricciato e Monaco [30] propuseram uma formulação em Programação Linear Inteira (PLI) para encaminhamento de fluxos. A formulação PLI foi empregada no cômputo de limites inferiores a partir da solução da relaxação contínua da instância off-line.

O modelo proposto busca a minimização de uma função composta pela ponderação entre o pico de banda máximo e a média de reserva de banda sobre o intervalo definido e para todos os enlaces. Em uma rede de computadores, utilizar uma ponderação entre o pico máximo e a média de reserva significa balancear o uso dos recursos nos enlaces através da escolha de caminhos que atendam à reserva. Estes caminhos são alocados de forma a não sobrecarregar um enlace e evitar caminhos muito longos. Como mencionado anteriormente, os protocolos utilizados não aproveitam completamente os recursos da rede, como no caso do BGP, onde somente o caminho considerado como sendo o melhor para realizar o encaminhamento é proposto. Desta forma, o caminho proposto pode ficar congestionado e caminhos alternativos, que poderiam oferecer um serviço de qualidade, ficam desocupados.

A seguir, expõem-se dois modelos para o encaminhamento de fluxos: um modelo para roteamento fixo e outro para roteamento variável. Entende-se por roteamento fixo um roteamento que não varia com o tempo, ou seja, uma vez definida a rota para um fluxo, ela não será alterada no futuro. Enquanto, no roteamento variável, as rotas de um fluxo podem ser modificadas com o tempo.

Modelo para Roteamento Fixo

O problema é formulado como segue:

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (2.2a)$$

Sujeito a :

$$\sum_{j:(s_k,j) \in E} r_{s_k j}^k = 1, \quad k \in \mathcal{X} \quad (2.2b)$$

$$\sum_{i:(i,d_k) \in E} r_{i d_k}^k = 1, \quad k \in \mathcal{X} \quad (2.2c)$$

$$\sum_{j:(i,j) \in E} r_{i j}^k - \sum_{j:(j,i) \in E} r_{j i}^k = 0, \quad k \in \mathcal{X}, i \in V \setminus \{s_k, d_k\} \quad (2.2d)$$

$$u_{ij}(\tau) = \sum_{k \in \mathcal{K}} f^k(\tau) \cdot r_{ij}^k, \quad (i, j) \in E, \tau \in \Theta \quad (2.2e)$$

$$v_{ij} \geq u_{ij}(\tau), \quad (i, j) \in E, \tau \in \Theta \quad (2.2f)$$

$$v_{ij} \leq C_{ij}, \quad (i, j) \in E \quad (2.2g)$$

$$c_{max} \geq \frac{1}{C_{ij}} \cdot v_{ij}, \quad (i, j) \in E \quad (2.2h)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{C_{ij}} \cdot v_{ij}, \quad (2.2i)$$

$$r_{ij}^k \in \{0, 1\}, \quad k \in \mathcal{X}, (i, j) \in E \quad (2.2j)$$

onde:

- r_{ij}^k assume o valor 1 se a requisição k atravessa o enlace (i, j) e 0 caso contrário;
- $u_{ij}(\tau)$ é a reserva de banda no enlace (i, j) para o período τ ;
- v_{ij} é a máxima reserva de banda para o enlace (i, j) considerando todos os períodos de tempo;
- C_{ij} é a capacidade do enlace (i, j) ;
- c_{max} é a máxima reserva percentual de todos os picos de banda considerando todos os enlaces e períodos de tempo;
- c_{mean} é a média percentual dos picos de banda para todos os períodos e enlaces; e
- $|E|$ é a cardinalidade de E , ou seja, o número de enlaces.

A formulação PLI apresenta através da equação (2.2a) uma função multi-objetivo que estabelece uma relação de compromisso entre o pico médio e o pico máximo da largura de banda nos enlaces.

Minimizar o c_{mean} (pico médio) tende para a seleção de caminhos mais curtos, pois sua minimização acarreta na seleção de menos arcos. Já a minimização de c_{max} tende para um uso mais equilibrado dos recursos de cada enlace.

A ponderação entre os objetivos é estabelecida pelo parâmetro α ($0 \leq \alpha \leq 1$), conhecido como *trade-off*, que define a importância relativa entre os objetivos. O estudo da influência do valor de α na utilização dos recursos é apresentado na seção 4.2.

As restrições (2.2b) e (2.2c) asseguram que exatamente um caminho será estabelecido entre o nó ingresso s_k e egresso d_k para cada requisição k , enquanto a restrição (2.2d) se refere aos nós intermediários. Estas restrições são típicas do modelo de fluxo em redes [1].

A equação (2.2e) define a reserva de banda para cada fatia de tempo em cada enlace. A desigualdade (2.2f) define o pico de banda reservada para cada enlace e a desigualdade (2.2g) garante que a capacidade de cada enlace não será ultrapassada. As restrições (2.2h) e (2.2i) definem respectivamente o valor máximo e médio da carga do enlace.

Neste modelo, o número de variáveis do modelo é $(|\mathcal{X}| + |\Theta| + 1)|E| + 2$.

O modelo para o roteamento fixo apresentado exige que todas as requisições sejam encaminhadas. Caso os recursos sejam insuficientes para alocar todas as requisições, o modelo será infactível obrigando a rejeição de requisições.

Modelo para roteamento variável

O problema é formulado como segue:

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (2.3a)$$

Sujeito a :

$$\sum_{j:(s_k,j) \in E} r_{s_k,j}^k(\tau) = 1, \quad k \in \mathcal{X}, \tau \in \Theta \quad (2.3b)$$

$$\sum_{i:(i,d_k) \in E} r_{i,d_k}^k(\tau) = 1, \quad k \in \mathcal{X}, \tau \in \Theta \quad (2.3c)$$

$$\sum_{j:(i,j) \in E} r_{i,j}^k(\tau) - \sum_{j:(j,i) \in E} r_{j,i}^k(\tau) = 0, \quad k \in \mathcal{X}, i \in V \setminus \{s_k, d_k\}, \tau \in \Theta \quad (2.3d)$$

$$u_{ij}(\tau) = \sum_{k \in \mathcal{K}} f^k(\tau) \cdot r_{ij}^k(\tau), \quad (i, j) \in E, \tau \in \Theta \quad (2.3e)$$

$$v_{ij} \geq u_{ij}(\tau), \quad (i, j) \in E, \tau \in \Theta \quad (2.3f)$$

$$v_{ij} \leq C_{ij}, \quad (i, j) \in E \quad (2.3g)$$

$$c_{max} \geq \frac{1}{C_{ij}} \cdot v_{ij}, \quad (i, j) \in E \quad (2.3h)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{C_{ij}} \cdot v_{ij}, \quad (2.3i)$$

$$r_{ij}^k(\tau) \in \{0, 1\}, \quad k \in \mathcal{X}, (i, j) \in E, \tau \in \Theta \quad (2.3j)$$

onde:

- $r_{ij}^k(\tau)$ assume o valor 1 se a requisição k atravessa o enlace (i, j) no período de tempo τ e 0 caso contrário;
- $u_{ij}(\tau)$ é a reserva de banda no enlace (i, j) para o período τ ;
- v_{ij} é a máxima reserva de banda para o enlace (i, j) considerando todos os períodos de tempo;
- C_{ij} é a capacidade do enlace (i, j) ;
- c_{max} é a máxima reserva percentual dos picos de banda considerando todos os enlaces e períodos de tempo; e
- c_{mean} é a média percentual de todos os picos de banda para todos os períodos e enlaces.

Para obter a formulação para o roteamento variável, uma pequena modificação foi feita nas equações (2.2b), (2.2c), (2.2d) e (2.2e) da formulação apresentada na seção 2.3.2. Nestas equações a variável r_{ij}^k foi substituída pela variável $r_{ij}^k(\tau)$.

Apesar do número de restrições permanecerem o mesmo do modelo fixo, o número de variáveis para o modelo variável passa a ser $|\mathcal{K}||E||\Theta| + |\Theta||E| + |E| + 2$.

2.3.3 Heurística para o Roteamento On-line

O algoritmo de roteamento on-line proposto por Ricciato e Monaco [30] considera que as requisições chegam aleatoriamente na rede. Assim que uma requisição é admitida, enlaces que não possuem capacidade para atender à requisição naquele instante são removidos da topologia.

Em seguida, um custo é definido para cada enlace formando um grafo direcionado. O custo é definido pela expressão abaixo:

$$w_{ij} = \frac{C_{ij}}{C_{ij} - x_{ij}} + \varepsilon \quad (2.4a)$$

onde

$$x_{ij} = \max_{\tau \in \{1, 2, \dots, \theta\}} \{f^k(\tau) + u_{ij}(\tau)\}, \quad (2.4b)$$

Sendo:

- C_{ij} é a capacidade do enlace (i, j) ;
- w_{ij} é o custo associado ao enlace (i, j) ; e
- $0 < \varepsilon \ll 1$ é um parâmetro arbitrário.

O algoritmo de caminhos mínimos de *Dijkstra* é executado no grafo obtido para computar o caminho de menor custo que atenda a cada requisição k . Note que os custos dos enlaces são não negativos e, portanto, o algoritmo de *Dijkstra* está bem definido e pode ser aplicado ao grafo.

A heurística proposta por Ricciato e Monaco [30] atende a cada requisição k que chega à rede procurando alocá-la. Para cada nova requisição k , enlaces que não possuem capacidade suficiente para acomodá-la para algum τ são removidos da topologia (grafo). Em seguida, um custo w_{ij} é alocado para cada enlace (i, j) de acordo com a função (2.4a). Com base nesta equação, um novo grafo com pesos não negativos é produzido e o algoritmo de *Dijkstra* é executado para encontrar o caminho de menor custo entre os pares origem-destino da requisição k . Por fim, calcula-se o valor da função objetivo.

Ricciato e Monaco [30] apresentam as informações sobre a heurística proposta de forma textual e através de equações. No trabalho desenvolvido para esta dissertação, escreveu-se de forma estruturada o Algoritmo 1 baseado nas informações apresentadas em [30].

Algoritmo 1 Heurística proposta por Ricciato & Monaco [30]

procedure *heuristicRM*()

```

1: for all  $(i, j) \in E \wedge \tau \in \Theta$  do
2:    $u_{ij}(\tau) = 0$ 
3: end for
4: for each requisição  $k \in \mathcal{X}$  do
5:   for each enlace  $(i, j) \in E$  do
6:     if  $\max\{f^k(\tau) + u_{ij}(\tau) : \tau \in \Theta\} \geq C_{ij}$  then
7:       esconde enlace  $(i, j) \in E$ 
8:     end if
9:   end for
10:  for each enlace  $(i, j) \in E$  do
11:    define custo  $w_{ij}$  conforme equações (2.4a) – (2.4b)
12:  end for
13:  cômputo do caminho  $P^k$  de menor custo em  $G$  de  $s_k$  para  $d_k$  com algoritmo de Dijkstra
14:  if  $\nexists$  caminho  $P^k$  de  $s_k$  para  $d_k$  then
15:    indica rejeição
16:  else
17:    for all  $(i, j) \in P^k$  do
18:       $u_{ij}(\tau) = u_{ij}(\tau) + f^k(\tau), \tau = 1, \dots, \theta$ 
19:    end for
20:  end if
21:  restaura enlaces escondidos
22: end for
23: calcula função objetivo  $c$ 

```

A ordem de execução deste algoritmo é $O(|\mathcal{X}|(|E||\Theta| + |E|lg|V|))$, considerando que a fila de prioridades em *Dijkstra* foi implementada utilizando uma *heap* binária.

O modelo apresentado considera o volume de tráfego conhecido previamente durante os diferentes períodos de tempo permitindo determinar uma configuração estática da rede para acomodar este tráfego. Ricciato e Monaco [30] verificaram que não existe a necessidade de configurações variáveis da rede, pois as variações do tráfego são conhecidas previamente, ou podem ser previstas com medições. Além disso, concluem que este conhecimento prévio dos perfis das demandas de tráfego pode ser explorado para obter uma configuração estática eficiente.

2.3.4 Comentários

Ricciato e Monaco [30] apresentam uma solução para o problema de roteamento de fluxos com perfis variantes no tempo em uma rede MPLS. Propuseram um modelo em PLI para solução do problema de roteamento off-line e uma heurística simples para o problema on-line. A solução off-line

foi utilizada como limite de referência para comparação com a solução heurística on-line. A solução heurística para o roteamento fixo obteve valores muito próximos da solução off-line, sugerindo que a utilização do modelo variável não traz grandes vantagens.

Baseado nos resultados obtidos, os autores consideraram que o algoritmo tem boa eficiência, obtendo resultados muito próximos aos valores ótimos em termos de uso da banda podendo ser utilizado em cenários mais avançados. Este modelo pode, também, auxiliar um provedor a oferecer serviços mais flexíveis e que atendam às necessidades dos usuários. Além disso, os autores concluem que tendo o conhecimento prévio das demandas de banda, pode-se obter uma configuração estática ótima que não necessita ser aperfeiçoada com reconfigurações dinâmicas. Entretanto, a inclusão de restrições adicionais torna necessária uma nova formulação do problema e novas heurísticas. Este problema não foi abordado pelos autores em [30].

2.4 Conclusões

O objetivo deste capítulo foi apresentar as soluções utilizadas atualmente para roteamento em redes de computadores e a necessidade do roteamento com QoS, incluindo roteamento intra-domínio, inter-domínio e o uso de MPLS na engenharia de tráfego. Foram apresentados e discutidos diversos trabalhos que apresentam formulações matemáticas e algoritmos como solução para o problema de roteamento. Em seguida, apresentou-se o problema do roteamento com perfis variantes no tempo. Perfis de banda variantes no tempo apresentam a vantagem em relação à economia de banda se comparado à modelos que utilizam o pico máximo para fazer a reserva. O trabalho de Ricciato e Monaco [30] propôs uma solução baseada nos perfis de banda variantes no tempo e obteve bons resultados quando comparado com outros trabalhos.

Os problemas e conceitos apresentados dão base à contextualização deste trabalho. No capítulo 3, apresentar-se-á uma proposta de roteamento baseado em caminhos tendo como base o trabalho de Ricciato e Monaco [30].

Capítulo 3

Roteamento Baseado em Caminhos

A forma como o problema foi modelado por Ricciato e Monaco [30], apresentado na seção 2.3.1, utiliza todos os arcos do grafo para obter a solução. Além disso, faz com que a adição de novas restrições ao modelo o tornem mais complexo e isto acarreta o aumento do tempo computacional para obtenção da solução do problema.

Um novo modelo é proposto para lidar com o problema de roteamento com perfis variantes no tempo, de forma que seja possível ter um controle sobre a qualidade da solução e que possibilite a adição de novas restrições sem tornar o modelo mais complexo. Assim como em [30], considerou-se o problema de encaminhamento de fluxos de dados em redes com arquitetura MPLS, sendo conhecidas *a priori* as demandas de banda durante os diferentes períodos do dia. Os perfis de banda podem ser explicitamente declarados pelos usuários e inseridos em acordos de níveis de serviço (SLAs). Um modelo matemático e uma heurística são propostos.

O modelo proposto nesta seção denominado CFC, ao contrário do modelo apresentado em [30] denominado RM, utiliza caminhos pré-computados para obter a solução do problema. Um modelo baseado em caminhos possibilita controlar o número de caminhos que serão utilizados para obter a solução, o que influencia no tempo computacional. Os caminhos são selecionados previamente possibilitando incluir no modelo restrições adicionais sem aumentar a sua complexidade. Isto é possível porque as restrições podem ser aplicadas para seleção dos caminhos que serão adicionados ao modelo no momento da obtenção da solução. Este capítulo apresenta o modelo proposto em formulação matemática, a heurística e como foi realizado o cômputo de caminhos. Detalhes sobre o uso de restrições adicionais no modelo serão apresentados no capítulo 5.

3.1 Algoritmos Baseados em Caminhos

Os algoritmos apresentados anteriormente (seção 2.2.3), em geral selecionam um único caminho para o encaminhamento de uma requisição. Esta seleção é baseada no menor custo ou em alguma outra restrição, de forma que o caminho selecionado possa satisfazer as exigências de uma determinada requisição.

O pré-cômputo pode oferecer benefícios como elevar a escalabilidade, melhorar a capacidade de lidar com as falhas, melhorar o desempenho geral e balanceamento da rede através de roteamento utilizando rotas alternativas. Orda e Sprintson [27] apresentam um plano de pré-computo de parâmetros que podem ser utilizados para realização do roteamento.

O sistema de pré-cômputo de informações e uma implementação real da proposta para solução do problema de roteamento podem ser implementados em cada nó ou de forma centralizada em um único servidor de roteamento.

Neste trabalho e no de Orda e Sprintson [27], o pré-computo de caminhos é realizado. Diferentemente dos outros citados, utiliza-se o cômputo dos caminhos que satisfazem os requisitos de QoS. Em seguida, a solução é obtida utilizando os dados obtidos nessa etapa.

3.2 Modelo Proposto em Programação Matemática

Para o modelo proposto CFC, a formulação PLI foi empregada no cômputo da solução inteira (quando possível) e da relaxação contínua da instância off-line. Assim como em [30], o modelo busca a minimização ponderada entre o pico de banda máximo e a média de reserva de banda.

Do mesmo modo que na seção 2.3.2, apresentam-se dois modelos para o encaminhamento de fluxos: um para roteamento fixo e outro para roteamento variável.

3.2.1 Modelo para Roteamento Fixo

O problema é formulado como segue:

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (3.1a)$$

Sujeito a :

$$\sum_{l \in \mathcal{L}(k)} r_l^k = 1, \quad k \in \mathcal{X} \quad (3.1b)$$

$$u_{ij}(\tau) = \sum_{k \in \mathcal{X}} \sum_{l \in \mathcal{L}(k)} \{f^k(\tau) \cdot r_l^k \text{ se } (i, j) \in P_l^k\}, \quad (i, j) \in E, \tau \in \Theta \quad (3.1c)$$

$$v_{ij} \geq u_{ij}(\tau), \quad (i, j) \in E, \tau \in \Theta \quad (3.1d)$$

$$v_{ij} \leq C_{ij}, \quad (i, j) \in E \quad (3.1e)$$

$$c_{max} \geq \frac{1}{C_{ij}} \cdot v_{ij}, \quad (i, j) \in E \quad (3.1f)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{C_{ij}} \cdot v_{ij}, \quad (3.1g)$$

$$r_l^k \in \{0, 1\}, \quad k \in \mathcal{X}, l \in \mathcal{L}(k) \quad (3.1h)$$

onde:

- $\mathcal{L}(k) = \{1, 2, \dots, l(k)\}$ corresponde ao conjunto de índices dos caminhos tendo origem s_k e destino d_k e $l(k)$ é o número de caminhos disponíveis para a requisição k ; e
- P_l^k é o l -ésimo caminho disponível para a requisição k , o qual consiste dos enlaces atravessados pelo caminho desde o nó s_k até o d_k .

A função objetivo (3.1a) é a mesma apresentada na seção 2.3.2. A restrição (3.1b) assegura que cada requisição k será encaminhada por um único caminho entre o nó ingresso s_k e egresso d_k . A equação (3.1c) estabelece a reserva de banda para cada enlace (i, j) na fatia de tempo τ se o enlace (i, j) pertencer ao caminho P_l^k , e se este caminho for selecionado para o encaminhamento da requisição k .

Como na formulação da seção 2.3.2, a desigualdade (3.1d) define o pico de banda reservada para cada enlace e a desigualdade (3.1e) garante que a capacidade de cada enlace não será ultrapassada. De forma similar ainda à formulação anterior, as restrições (3.1f) e (3.1g) definem respectivamente o valor máximo e médio da carga do enlace.

Supondo por simplicidade $\mathcal{L}(k) = \mathcal{L}$ para todo $k \in \mathcal{X}$, então o número de variáveis do modelo é $|\mathcal{L}| |\mathcal{X}| + |E| |\Theta| + |E| + 2$. A diferença do número de variáveis entre o modelo RM e CFC para

roteamento fixo é $|\mathcal{X}|(|E| - |\mathcal{L}|)$. Esta diferença se torna significativa a favor do modelo CFC se com este modelo se consegue obter uma solução satisfatória utilizando poucos caminhos como, por exemplo, quando utilizados somente caminhos arcos-disjuntos.

Assim como mencionado na seção 2.3.2, o modelo para roteamento fixo apresentado exige que todas as requisições sejam encaminhadas e, havendo recursos insuficientes, o modelo será infactível obrigando a rejeição de requisições.

3.2.2 Modelo para Roteamento Variável

O problema é formulado como segue:

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (3.2a)$$

Sujeito a :

$$\sum_{l \in \mathcal{L}(k)} r_l^k(\tau) = 1, \quad k \in \mathcal{X}, \tau \in \Theta \quad (3.2b)$$

$$u_{ij}(\tau) = \sum_{k \in \mathcal{X}} \sum_{l \in \mathcal{L}(k)} \{f^k(\tau) \cdot r_l^k(\tau)\} \text{ se } (i, j) \in P_l^k, \quad (i, j) \in E, \tau \in \Theta \quad (3.2c)$$

$$v_{ij} \geq u_{ij}(\tau), \quad (i, j) \in E, \tau \in \Theta \quad (3.2d)$$

$$v_{ij} \leq C_{ij}, \quad (i, j) \in E \quad (3.2e)$$

$$c_{max} \geq \frac{1}{C_{ij}} \cdot v_{ij}, \quad (i, j) \in E \quad (3.2f)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{C_{ij}} \cdot v_{ij}, \quad (3.2g)$$

$$r_l^k(\tau) \in \{0, 1\}, \quad k \in \mathcal{X}, l \in \mathcal{L}(k), \tau \in \Theta \quad (3.2h)$$

Do mesmo modo que foi feito na seção 2.3.2, para obter a formulação para o roteamento variável segundo o modelo RM, foi feita uma pequena modificação nas equações (3.1b) e (3.1c) da formulação do modelo CFC apresentada na seção 3.2.1. Nestas equações, a variável r_l^k foi substituída pela $r_l^k(\tau)$.

Apesar do número de restrições permanecerem o mesmo do modelo fixo, o número de variáveis para o modelo variável passa a ser $|\mathcal{X}||\mathcal{L}||\Theta| + |\Theta||E| + |E| + 2$.

O roteamento variável permite que a rota alocada para cada demanda mude a cada intervalo de tempo permitindo que a configuração se adapte proporcionando uma economia de recursos. Apesar

de proporcionar uma economia de recursos, o modelo variável tem uma desvantagem em relação ao modelo fixo: o seu uso necessita de uma capacidade computacional superior dos dispositivos de rede, além de aumentar a complexidade dos protocolos de rede. A seção 4.3 apresenta uma comparação dos dois modelos CFC (fixo e variável) e permite tirar conclusões sobre o modelo de maior interesse.

3.2.3 Heurística

A heurística implementada utiliza caminhos pré-computados com o objetivo de minimizar a reserva de banda, buscando encaminhar os fluxos de forma que se respeite aos requisitos exigidos pelas requisições. Pode-se optar por caminhos arcos-disjuntos definidos tanto pelo provedor como pelo usuário, buscando atender às restrições. Estes caminhos podem ser obtidos através de procedimentos sistemáticos de enumeração ou produzidos por um método estocástico. Detalhes sobre a obtenção dos caminhos são apresentados na seção 3.3. A seguir, descreve-se o princípio da heurística e apresenta-se o algoritmo.

Para cada requisição k , os caminhos obtidos são adicionados ao conjunto P_l^k . Uma vez que os caminhos foram obtidos, para cada requisição uma busca é realizada no conjunto correspondente. Caminhos que não possuem recursos suficientes para alocar as requisições são eliminados. Para cada requisição, dentre os caminhos disponíveis, aquele que minimizar o valor da função objetivo corresponderá ao caminho \hat{P}_{min}^k . O caminho selecionado para encaminhar a requisição é aquele que fornece o menor valor para a função objetivo. Para o roteamento off-line, a heurística tem conhecimento de todas as requisições a fim de realizar o encaminhamento dos fluxos. No roteamento on-line, cada requisição é alocada de acordo com a chegada na rede. Os valores finais de c_{mean} , c_{max} e da função objetivo c são calculados no encerramento da execução da heurística de forma a obter o valor ótimo. O pseudocódigo da heurística é apresentado pelo Algoritmo 2. A ordem de execução deste algoritmo é $O(|\mathcal{X}| \{|\mathcal{L}||\Theta| + |E||\Theta|\})$.

3.3 Cômputo de Caminhos

Segundo Orda e Sprintson [27], o uso de métodos baseados no pré-cômputo de informações é uma abordagem necessária para controle e gerenciamento eficiente da rede, principalmente quando associado ao roteamento com QoS. Neste trabalho, utilizou-se o pré-cômputo de caminhos, que pode ser realizado em momentos de ociosidade do equipamento que efetua o cômputo de caminhos ou em processos de baixa prioridade em “plano de fundo” (*background*).

Algoritmo 2 Heurística baseada em caminhos**procedure** *heuristicCFC*()

```

1: for all  $(i, j) \in E \wedge \tau \in \Theta$  do
2:    $u_{ij}(\tau) = 0$ 
3: end for
4: for each requisição  $k \in \mathcal{X}$  do
5:    $\hat{c}_{min} = +\infty$ 
6:    $\hat{P}_{min}^k = \emptyset$ 
7:   for all  $l \in \mathcal{L}(k)$  do
8:     if  $\forall (i, j) \in P_l^k, f^k(\tau) + u_{ij}(\tau) < C_{ij}$  then
9:       calcula  $\hat{c}_{mean}, \hat{c}_{max} \wedge \hat{c} = \alpha \cdot \hat{c}_{max} + (1 - \alpha) \cdot \hat{c}_{mean}$  se requisição  $k$  for encaminhada ao longo do
       caminho de  $P_l^k$ 
10:    end if
11:    if  $\hat{c} < \hat{c}_{min}$  then
12:       $\hat{c}_{min} \leftarrow \hat{c}, \hat{P}_{min}^k \leftarrow P_l^k$ 
13:    end if
14:  end for
15:  if  $\hat{c}_{min} = +\infty$  then
16:    fluxo rejeitado
17:  else
18:    encaminha requisição  $k$  ao logo do caminho  $\hat{P}_{min}^k$ 
19:    for all  $(i, j) \in \hat{P}_{min}^k \wedge \tau \in \Theta$  do
20:       $u_{ij}(\tau) = u_{ij}(\tau) + f^k(\tau)$ 
21:    end for
22:  end if
23: end for

```

Três modos de obtenção de caminhos (arcos-disjuntos, caminhos aleatórios, todos os caminhos) foram utilizados com o objetivo de obter um número diferenciado de caminhos e garantir boa distribuição dos caminhos pela rede apresentar-se-ão a seguir.

3.3.1 Caminhos Arcos-Disjuntos

O Conjunto de caminhos arcos-disjuntos é formado por caminhos que podem conter o mesmo nó, mas não o mesmo arco. Os caminhos arcos-disjuntos foram obtidos através da implementação do algoritmo de fluxo máximo *Ford-Fulkerson* [1, 8]. As capacidades dos arcos foram definidas com o valor um, desta forma cada arco somente pode ser utilizado para encaminhar uma unidade. Uma vez que o algoritmo de fluxo máximo tenha sido computado entre um nó origem s e um nó destino t , um outro algoritmo utiliza os nós de origem e de destino para gerar os caminhos a partir do vetor que guarda os predecessores de cada nodo.

Este algoritmo é executado na fase de cômputo de caminhos. Deve-se notar que caminhos arcos-disjuntos não são necessariamente nós-disjuntos. Caso se tenha interesse por caminhos nós-disjuntos (caminhos que não utilizam o mesmo nó nem o mesmo arco), um modelo em fluxo máximo pode ser obtido, pois caminhos nós-disjuntos são necessariamente arcos-disjuntos.

3.3.2 Heurística de Geração de Caminhos Aleatória

Como a obtenção de todos os caminhos se torna lenta com o aumento do grafo, propõe-se um algoritmo para obter um determinado número de caminhos. Para obter caminhos adicionais aos disjuntos, utilizou-se um algoritmo de busca em profundidade modificado para trabalhar com escolha aleatória dos nós que não foram visitados. Com o aumento do número de repetições do algoritmo, pode-se acrescentar cada vez mais caminhos até chegar próximo ao valor máximo de caminhos do grafo.

O Algoritmo 3 recebe como parâmetro o grafo, o nó s de origem e o nó t de destino. O estado de cada nó é marcado como **WHITE** (não visitado) e como **GRAY** (visitado). Depois de feita a inicialização dos nodos do grafo, o procedimento **CAVISIT** (Algoritmo 4) visita os nodos adjacentes ao anterior até atingir o de destino. A escolha do nodo a ser visitado é aleatória. Os adjacentes ao nodo inicial ou ao nodo escolhido são permutados e selecionados aleatoriamente se tornando o próximo nodo visitado e marcado como visitado. À medida que são visitados, os nodos são adicionados a uma lista que compõe o caminho. Ao final da execução, demonstra-se os caminhos entre s e t .

Algoritmo 3 Heurística aleatória para geração de caminhos

procedure *heuristicCA*()

- 1: **for all** nodos $u \in V[G]$ **do**
 - 2: **COLOR**[u] = **WHITE**
 - 3: **end for**
 - 4: **if** *CAVISIT*(G, s, t) = 0 **then**
 - 5: Return 0
 - 6: **end if**
 - 7: Return 1
-

3.3.3 Algoritmo de Geração de Todos os Caminhos

Um algoritmo com objetivo de encontrar todos os caminhos de um grafo [32] também foi utilizado para a realização dos experimentos. Entretanto, parece ser de maior interesse optar pela utilização de um número reduzido de caminhos para os grafos complexos, pois a utilização de todos os caminhos para resolver o problema de roteamento tornaria o tempo de solução elevado.

O algoritmo utilizado para obtenção de todos os caminhos do grafo foi obtido em [32]. Uma abordagem similar é apresentada em [25]. Devido ao esforço computacional e tempo necessário para computar todos os caminhos de um grafo relativamente grande, o cômputo de todos os caminhos somente foi utilizado em instâncias pequenas.

Algoritmo 4 Função de obtenção dos nodos de um caminho**procedure** *CAVISIT*(G, s, t)

```

1: COLOR[s] = GRAY
2: if  $s = t$  then
3:   adiciona nodo  $s$  ao caminho
4:   Return 1
5: else
6:   for all nodos adjacentes a  $s$  do
7:     if COLOR[v] = WHITE then
8:       adiciona a lista de nodos adjacentes
9:     end if
10:  end for
11: end if
12: if lista de nodos adjacentes está vazia then
13:   Return 0
14: end if
15: Escolhe aleatoriamente um nodo  $v$  da lista de adjacência
16: CAVISIT( $G, v, t$ )
17: adiciona nodo  $s$  ao caminho

```

O Algoritmo 5 apresenta o pseudocódigo para obtenção de todos os caminhos entre uma origem s e um destino t em um determinado grafo.

Algoritmo 5 Algoritmo para obtenção de todos os caminhos**procedure** *FINDALLPATHS*(G, s, t)

```

1: path = path + s;
2: if  $s = t$  then
3:   return path
4: end if
5: for all nodos  $v$  adjacentes a  $s$  do
6:   if node  $v \notin$  path then
7:     NEWPATHS = FINDALLPATHS( $G, v, t$ )
8:     for newpath  $\in$  NEWPATHS do
9:       adiciona newpath em paths
10:    end for
11:   end if
12: end for
13: return paths

```

Este algoritmo busca recursivamente os caminhos do grafo partindo do nodo de origem s e navegando pelos nodos adjacentes até atingir o nó de destino t .

A figura 3.1 apresenta o grafo utilizado para ilustrar o funcionamento do algoritmo.

Na tabela 3.1 são listados os nodos adjacentes, ou seja, uma matriz de adjacência.

As iterações para encontrar os caminhos entre o nó de origem A e o nó de destino D para o grafo da figura 3.1 podem ser visualizadas na figura 3.2. O nó inicial A tem como adjacentes os nós B e C . Na segunda iteração, é selecionado o nó B , que tem como adjacentes os C e D . Na terceira iteração,

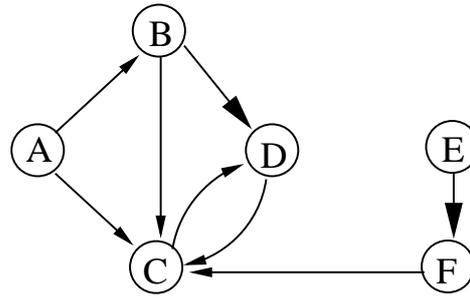


Figura 3.1: Grafo utilizado para ilustrar o funcionamento do algoritmo que gera todos os caminhos

Tabela 3.1: Nodo (chave) com arco ligado diretamente a outro nodo

Chave	ligada ao nodo	ligada ao nodo
A	B	C
B	C	D
C	D	–
D	C	–
E	F	–
F	C	–

o nó *C* é selecionado, que tem como adjacente o *D*. Na quarta iteração, o nó *D* é selecionado. Como na quarta iteração o destino é atingido, o caminho é adicionado à lista de caminhos e o algoritmo continua a quinta iteração a partir do outro nó adjacente ao *B*, o nó *D*. Como na quinta iteração o destino é atingido, o caminho é adicionado à lista e o algoritmo continua a sexta iteração a partir do outro nó adjacente ao *A*, o nó *C*. Na sétima iteração, o nó *D* é selecionado atingindo o nó destino. O caminho é adicionado à lista e o algoritmo termina a execução, pois não possui mais nós adjacentes.

3.4 Conclusões

O objetivo deste capítulo foi apresentar um novo modelo para roteamento com perfis variantes no tempo baseado em caminhos. Enquanto o modelo RM analisa todos os arcos do grafo para obter a solução ótima, o modelo CFC analisa caminhos. A abordagem baseada em caminhos permite controlar a qualidade da solução através do controle de caminhos utilizados no cômputo da melhor rota.

Três algoritmos para obtenção de caminhos foram apresentados. Os caminhos obtidos com estes algoritmos são utilizados nos experimentos do capítulo 4.

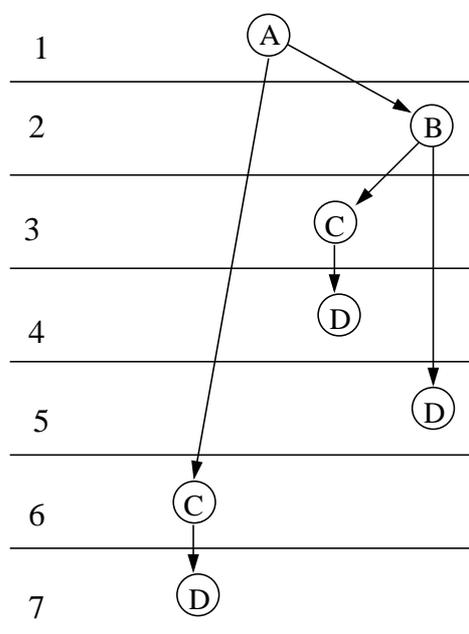


Figura 3.2: *Iterações na geração dos caminhos*

Capítulo 4

Estudo Comparativo Experimental

Os experimentos a serem expostos visam a analisar a viabilidade do modelo CFC proposto e as suas performances e compará-lo ao modelo RM quando possível. Este modelo permite controlar o tempo computacional influenciando a qualidade da solução através do número de caminhos que são pré-computados, além de obter valores similares aos obtidos com o modelo RM. A economia de recursos expressa em termos de banda e o número de demandas atendidas sem rejeição serão critérios para a análise comparativa dos modelos CFC e RM, tanto para o caso do roteamento fixo como o variável.

Primeiramente, demonstram-se as condições experimentais, seguidas de análise do parâmetro α . Após, faz-se uma comparação do roteamento fixo e variável e do impacto da granularidade do período de tempo na aceitação das demandas. Finalmente, são apresentados os resultados para o roteamento off-line e on-line.

Para poder realizar estas análises, os modelos PLI foram implementados na linguagem de modelagem de programação matemática AMPL [12] e solucionados com otimizador matemático CPLEX [15]. A máquina utilizada possui processador Intel Pentium 4 de 2,66 GHz e 1 GB de memória RAM com sistema operacional GNU/Linux Mandrake 9.2 e Kernel 2.4.22.

4.1 Condições Experimentais

Os experimentos foram realizados com as topologias apresentadas nas figuras 4.1 e 4.2. A topologia da figura 4.1 (10 nós, 18 enlaces) foi obtida a partir da remoção de alguns nós de uma das topologias utilizadas em [30] para que fosse possível solucionar o problema inteiro em menor

tempo. Os enlaces representados por uma linha fina têm capacidade denotada por C_{ij} , enquanto os representados por uma linha grossa possuem capacidade $3C_{ij}$. A topologia da figura 4.2 (30 nós, 61 enlaces) foi retirada de [30]; todos os enlaces dessa possuem a mesma capacidade C_{ij} .

A origem e o destino de cada demanda k foram gerados aleatoriamente com distribuição uniforme. Para cada demanda k em cada fatia de tempo τ , gerou-se o perfil aleatoriamente com distribuição uniforme, variando de 0 a 5 unidades. A capacidade C_{ij} é de 125 unidades em cada direção. Considerando que cada unidade corresponde a 20Mbps, tem-se que a capacidade C_{ij} é de 2.5 Gbps e que o máximo reservado para cada requisição é 100Mbps.

Para os experimentos off-line, a duração de cada requisição é persistente, ou seja, uma vez encaminhada, a requisição permanece ativa até o encerramento da simulação e o $\theta = 5$. No caso dos experimentos on-line, a duração de cada requisição é de 6 fatias de tempo τ com $\theta = 12$. Depois de certo tempo de alocação de requisições, ocorre a saturação dos enlaces, quando então é iniciada a rejeição de novas requisições. O motivo da escolha destas condições experimentais é a simplicidade e a não pretensão de representar uma distribuição real de tráfego em redes.

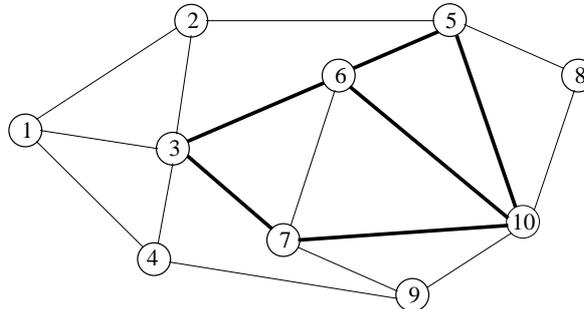


Figura 4.1: Rede de 10 nós utilizada para realização dos experimentos

Nos experimentos apresentados, a principal preocupação foi com a corretude dos algoritmos e não com o tempo computacional e eficiência destes.

4.2 Influência da Ponderação entre c_{max} e c_{mean}

Os modelos matemáticos dos problemas de encaminhamento de fluxo apresentados são do tipo multi-objetivo. A função objetivo que se deseja minimizar possui os critérios c_{mean} e c_{max} e um parâmetro α que serve para ponderar a importância destes dois objetivos. Os modelos buscam evitar que enlaces fiquem saturados (minimizando c_{max}) e que haja encaminhamentos através de caminhos muito longos (minimizando c_{mean}). Neste experimento, utilizou-se o modelo RM fixo (ver 2.3.2), sendo as instâncias PLI resolvidas utilizando o CPLEX [15].

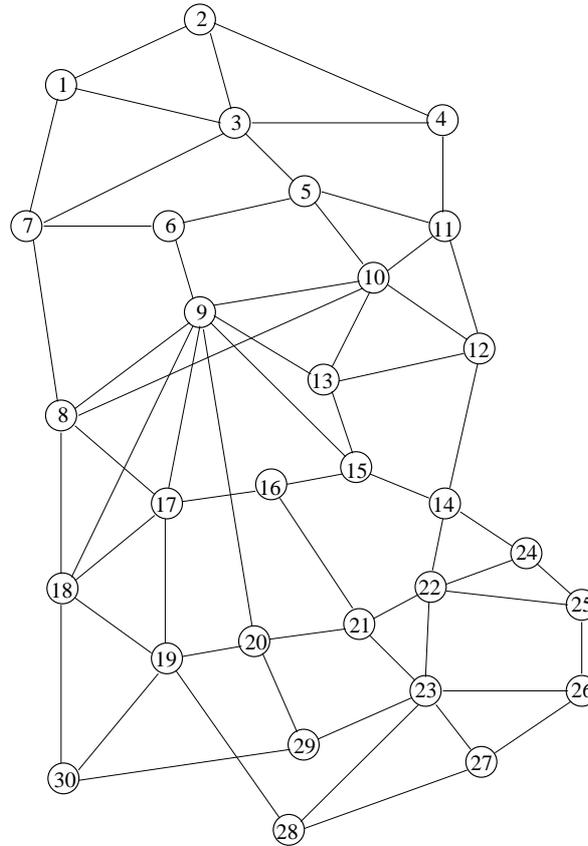


Figura 4.2: Rede de 30 nós utilizada para realização dos experimentos

Um grafo simplificado com cinco nós e cinco arcos, como ilustra a figura 4.3, foi utilizado para visualizar o comportamento dos critérios em relação à variação do α . A quantidade máxima de banda que pode ser encaminhada neste grafo entre o nó 1 e o 5 é de 200 unidades. Para avaliar o comportamento, 10 requisições entre o nó 1 (origem s) e 5 (destino t) foram encaminhadas, cada uma solicitando 10 unidades de banda totalizando 100 unidades de banda.

Além de mostrar o grafo utilizado, a figura 4.3 também apresenta para cada enlace (i, j) a máxima reserva de banda (v_{ij}) e a capacidade (C_{ij}) . Na figura 4.3(a), podemos visualizar a máxima reserva de banda v_{ij} para o caso de $\alpha = 0$ (minimizar c_{mean}) e na figura 4.3(b) para o caso $\alpha = 1$ (minimizar c_{max}). O valor utilizado para θ foi 1, como se fosse o perfil de um dia inteiro.

Para que uma requisição seja encaminhada do nó 1 para o 5, dois caminhos podem ser utilizados: caminho (1), formado apenas pelo arco $1 \rightarrow 5$; caminho (2), formado pelos arcos $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$.

Pode-se visualizar na figura 4.3(a) que a ocupação do caminho (1) é de 100 unidades e do (2) é de 0 unidades, sendo que 100 unidades correspondem ao valor que se desejava encaminhar do nó 1

ao 5. Na figura 4.3(b), pode-se visualizar que os dois caminhos estão igualmente ocupados.

Como a função que se deseja minimizar é multi-objetivo, tem-se o parâmetro α para ponderar os objetivos. Quando o valor de α é igual a 0, apenas c_{mean} é considerado na minimização, enquanto c_{max} é desconsiderado. Para α igual a 1, o contrário acontece: apenas c_{max} é considerado na minimização, enquanto c_{mean} é desconsiderado.

O motivo que levou o caminho (2) na figura 4.3(a) a ter ocupação 0 é que somente quando c_{mean} é minimizado, os arcos escolhidos para compor o caminho são selecionados de forma a obter o menor valor, pois $c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{C_{ij}} \cdot v_{ij}$. Tendo um caminho curto utilizado na sua capacidade máxima e um caminho longo inutilizado, obtém-se um valor médio menor, por isso o caminho mais curto é escolhido e a ocupação é máxima. No caso dos caminhos (1) e (2) na figura 4.3(b) terem uma ocupação de 50 unidades, é que para a minimização do c_{max} , os arcos escolhidos para compor um caminho são selecionados de forma a minimizar o valor máximo, pois $c_{max} \geq \frac{1}{C_{ij}} \cdot v_{ij}$. Para que isto ocorra é necessário distribuir o tráfego entre os caminhos para que o pico máximo v_{ij} de cada arco (i, j) dividido pela capacidade C_{ij} daquele arco tenha um valor mais baixo, havendo, desta forma, uma distribuição das demandas pelos dois caminhos.

A curva que relaciona as variáveis c_{mean} e c_{max} com o grafo da figura 4.3, quando variado o parâmetro α , é apresentada na figura 4.4. Esta curva ilustra como a ponderação entre os dois objetivos se comporta. Pode-se verificar que com a diminuição do valor de α , c_{max} aumenta (é máximo para $\alpha = 0$) e que c_{mean} diminui (é mínimo para $\alpha = 0$). O contrário acontece para o aumento do valor de α . Para o caso apresentado na figura 4.4, a partir de $\alpha = 0,4$, o valor de c_{mean} , c_{max} e do critério c permanece constantes.

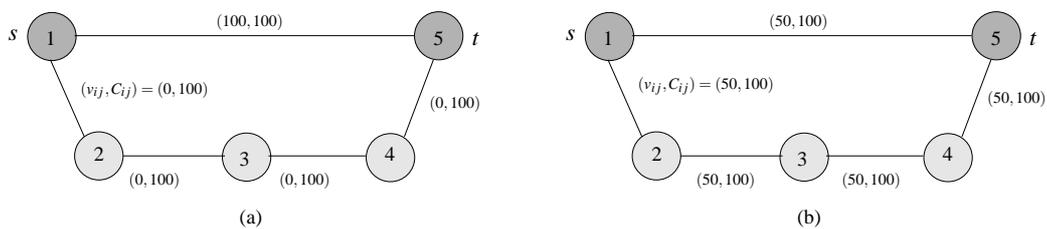


Figura 4.3: (a) Capacidade utilizada nos arcos para minimização de c_{mean} ($\alpha = 0$); (b) Capacidade utilizada nos arcos para minimização de c_{max} ($\alpha = 1$)

O mesmo experimento para avaliar a influência da ponderação de α entre c_{mean} e c_{max} realizado com o grafo da figura 4.3 foi realizado para o grafo da figura 4.1. Neste caso, as instâncias foram resolvidas pela heurística CFC off-line. Foram geradas 100 requisições entre os pares de nós origem-destino escolhidos aleatoriamente e uniformemente distribuídos conforme comentado na seção 4.1. As demandas de cada requisição foram escolhidas conforme explicado na seção 4.1.

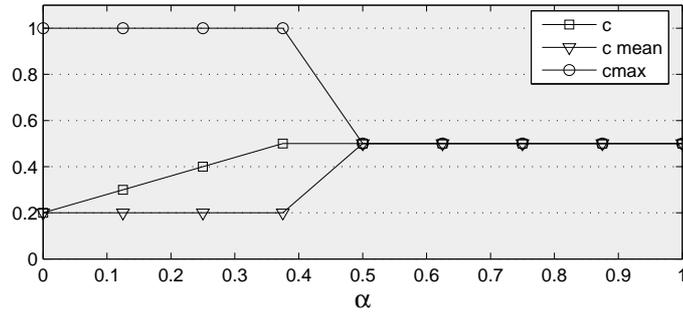


Figura 4.4: Relação entre c , c_{mean} e c_{max} com α variando

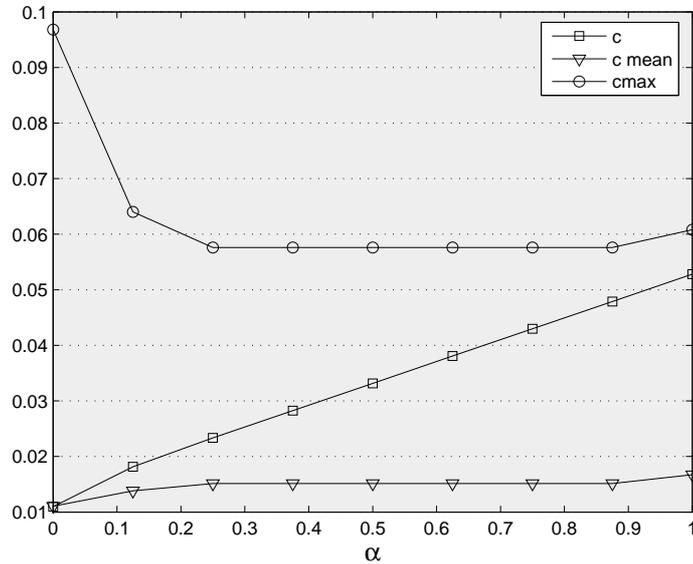


Figura 4.5: Relação entre c , c_{mean} e c_{max} com α variando - 10 requisições

Pode-se observar na figura 4.5 que a curva obtida apresenta comportamento muito similar à observada na figura 4.6. Este comportamento similar demonstra que a heurística apresenta resultados conforme os que se obtém através dos modelos matemáticos e que serve como forma de validar a implementação. Pôde-se observar que com o aumento do valor de α , há um aumento do critério c , de c_{mean} e uma diminuição de c_{max} .

4.3 Roteamento Fixo versus Variável

Ricciato e Monaco [30] realizaram uma comparação da capacidade de economia de banda entre o roteamento fixo e o variável. Os resultados encontrados em [30] mostram que o roteamento fixo tem pouca influência sobre a solução ótima. Resultados similares foram encontrados neste trabalho.

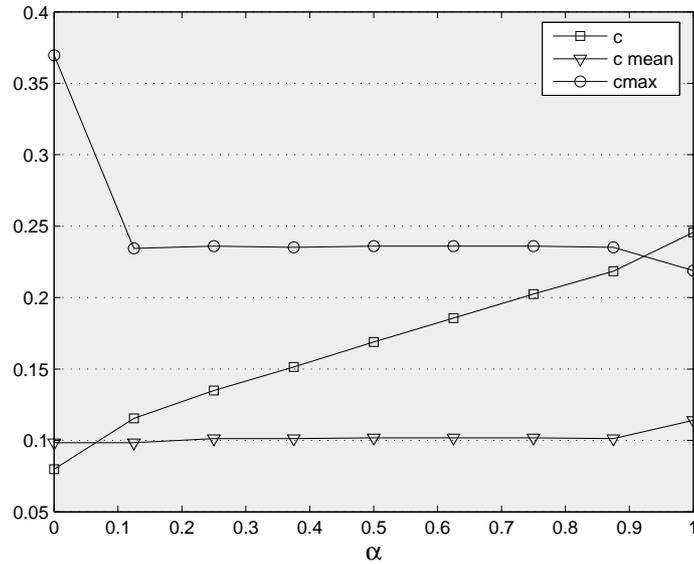


Figura 4.6: Relação entre c , c_{mean} e c_{max} com α variando - 100 requisições

Assim como em [30], as relaxações off-line do modelo CFC para roteamento fixo e variável foram solucionadas utilizando do CPLEX [15] para a topologia da figura 4.1. Também são apresentados os valores obtidos com a heurística (roteamento fixo).

Neste experimento, pretende-se realizar a mesma avaliação feita por Ricciato e Monaco [30], mas utilizando o modelo matemático baseado em caminhos proposto na seção 3.2. A comparação entre o roteamento fixo e o variável foi feita utilizando o critério c .

A tabela 4.1 apresenta os resultados obtidos para 20, 100, 200, 400, 600 e 800 requisições além da diferença percentual entre o roteamento fixo (obtido pela solução do modelo matemático) e o variável e entre a heurística e o roteamento variável. A diferença percentual entre os valores obtidos para os modelos é apresentada na tabela como GAP. A fórmula utilizada é $GAP = 100 \cdot \frac{(X-Y)}{Y}$, onde X é o valor obtido pelo roteamento fixo ou pela heurística e Y é o valor obtido pelo variável.

Pode-se visualizar no gráfico da figura 4.7 e na tabela 4.1 os valores médios obtidos com dez rodadas de experimentos com a relaxação para o roteamento fixo e variável e os valores médios obtidos com dez rodadas pela heurística descrita pelo Algoritmo 2 (seção 3.2.3).

Constatou-se que a heurística obteve bons resultados em relação ao valor obtido pela solução utilizando a relaxação (média menor que 10%) e que a sua utilização se apresenta como uma solução viável para o roteamento fixo. Resultados similares aos apresentados nesta seção foram obtidos por Ricciato e Monaco [30]. A diferença entre o roteamento fixo e o variável é tão pequena que pode ser utilizado o fixo a um custo mais baixo, pois a utilização do roteamento variável necessitaria da

Tabela 4.1: Comparativo: roteamento fixo e variável

Número de requisições	20	100	200	400	600	800
CFC Fixo (relaxação)	0,040314	0,135645	0,238862	0,414463	0,599995	0,790061
CFC Var. (relaxação)	0,039614	0,134468	0,237640	0,412932	0,598597	0,766059
CFC heurística (fixo)	0,048937	0,150052	0,257459	0,448367	0,657622	0,811630
GAP Fixo (%)	1,77	0,88	0,51	0,37	0,23	3,13
GAP heurística (%)	23,53	11,59	8,34	8,58	9,86	5,95

implementação de recursos adicionais nos protocolos de rede, o que tornaria o seu uso complexo.

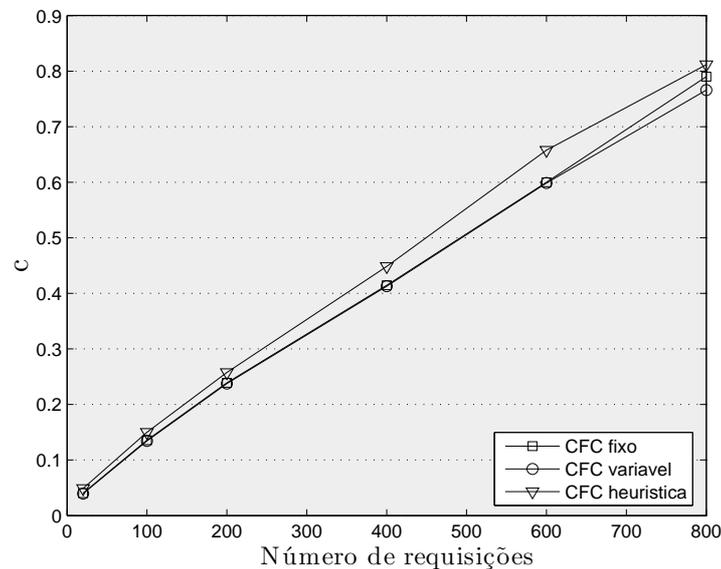


Figura 4.7: Comparativo: roteamento fixo e variável

4.4 Impacto da Granularidade do Período de Tempo na Aceitação de Demandas

Assim como em [30], foi feita uma avaliação simples em relação à granularidade do período de tempo para a rede da figura 4.1 utilizando todos os caminhos do grafo para cada requisição k . A tabela 4.2 apresenta o número médio de requisições alocadas antes da 1^a, 10^a e 100^a rejeição para as heurísticas dos modelos RM e CFC. Para cada θ , foram executadas dez rodadas a fim de obter a média. Para cada rodada, foram geradas requisições suficientes para congestionar o enlace. Contabilizou-se o número de requisições alocadas na ocorrência da 1^a, 10^a e 100^a rejeição. Para a execução da heurística do modelo CFC foram utilizados todos os caminhos do grafo. No entanto, observou-se

Tabela 4.2: Comparação do número de requisições alocadas

algoritmo	n -ésima rejeição	$\theta=5$	$\theta=10$	$\theta=15$	$\theta=20$
CFC	1	771	761	736	735
	10	780	770	745	744
	100	870	860	835	834
RM	1	812	813	773	770
	10	890	874	847	841
	100	1096	1070	1047	1035

que, com a diminuição do número de caminhos, os valores para rejeição permanecem os mesmos. Os resultados obtidos para o modelo CFC tiveram um comportamento similar ao observado para o modelo RM [30]: o aumento do número de períodos θ , por exemplo, em um dia, acarreta pequena diminuição no número de requisições alocadas pelos dois modelos fixos. O modelo RM apresenta uma maior alocação de requisições que o modelo CFC. O aumento da granularidade influencia muito pouco (inferior a 3 %) na diminuição de demandas alocadas para ambos os modelos.

4.5 Análise Experimental de Roteamento Off-line

Os experimentos off-line foram divididos em dois cenários. A seção 4.5.1 demonstra experimentos computacionais realizados com o grafo de 10 nós do cenário 1 e uma discussão dos resultados obtidos. A seção 4.5.2 apresenta experimentos computacionais realizados com o grafo de 30 nós do cenário 2 e uma discussão dos resultados obtidos. Nos experimentos, utilizou-se para o parâmetro α o valor $\frac{1}{2}$, no sentido de não dar preferência exclusiva a nenhum dos componentes do critério. Alguns experimentos apresentam resultados com a variação de α . O algoritmo CFC, descrito na seção 3.2.3, e o de Ricciato e Monaco, descrito na seção 2.3.3, foram implementados para a realização dos experimentos.

4.5.1 Cenário 1

Neste cenário adotou-se a rede da figura 4.1 com 10 nós e os perfis das demandas foram gerados conforme discutido na seção 4.1.

Os valores médios obtidos com dez execuções para os modelos em PLI para a solução inteira, relaxação e pela heurística para 15 requisições são apresentados na tabela 4.3. Diferentes sementes foram utilizadas para gerar as requisições. Para as tabelas referentes a 100, 400 e 600 requisições

somente se apresentam a relaxação e heurísticas, pois não houve possibilidade de obter os resultados para a solução inteira. Para o modelo CFC, baseado em caminhos, seis colunas são apresentadas para diferentes números de caminhos. A primeira coluna contém somente caminhos arcos-disjuntos. As cinco colunas seguintes contêm caminhos adicionais à coluna anterior até atingir o número máximo de caminhos do grafo (última coluna), para cada situação. As tabelas contêm os valores do critério c para todas essas situações. No caso do modelo RM, somente um valor do critério c é apresentado, pois o modelo não depende do número de caminhos.

A diferença percentual entre os valores obtidos para os modelos é apresentada na tabela como GAP. A comparação é feita entre inteiros, entre relaxações e entre heurísticas. A fórmula utilizada é $GAP = 100 \cdot \frac{(\overline{CFC} - \overline{RM})}{\overline{RM}}$. Nas comparações, \overline{CFC} indica o valor do critério c com o modelo CFC (relaxação ou heurística), enquanto \overline{RM} indica o valor do critério c obtido com o modelo RM (relaxação ou heurística).

Pode-se observar na tabela 4.3 que à medida que o número de caminhos aumenta (até atingir o máximo número de caminhos - penúltima coluna), a solução inteira e a relaxação se aproximam do valor das soluções obtidas usando o modelo de RM, obtendo o mesmo valor ótimo (solução inteira). A obtenção do mesmo valor ótimo valida o modelo CFC. Portanto, é possível afirmar que o modelo proposto baseado em caminhos consegue atingir a solução ótima quando alimentado com todos os caminhos. A solução obtida com a relaxação para o modelo CFC, aproxima-se dos valores obtidos com a relaxação do modelo RM apresentando uma diferença menor que 1%. Este comportamento pode ser observado mesmo com a variação do número de requisições, como pode ser visto nas tabelas 4.4, 4.5 e 4.6, correspondente a 100, 400 e 600 requisições respectivamente.

Os valores obtidos pela heurística do modelo CFC apresentam pouca variação com o aumento do número de caminhos. Apesar deste comportamento, o valor obtido é próximo do valor que se obtém através da relaxação, tanto do modelo RM, como do modelo CFC. Além disso, os valores obtidos com a heurística do modelo CFC para até 400 requisições são mais próximos do valor da relaxação do que os obtidos com a heurística de RM. A diferença é menor que 10%. Para mais de 400 requisições, CFC e RM ficam próximos um do outro com uma diferença menor que 8%.

É importante notar que, utilizando somente caminhos arcos-disjuntos (primeira coluna das tabelas), os valores obtidos pela heurística CFC apresentam bons resultados em relação aos obtidos com o modelo RM. Em conclusão para este cenário, o modelo CFC obteve bons resultados e pode substituir o modelo RM para realização de roteamento com perfis variantes no tempo.

Tabela 4.3: Valor do critério *c* - 15 requisições - cenário 1 off-line

		Número de Caminhos						Média
		46	253	432	660	878	1135	567
CFC	Inteiro	0,0403	0,0400	0,0400	0,0399	0,0399	0,0399	0,0400
	Relaxação	0,0347	0,0343	0,0343	0,0342	0,0342	0,0342	0,0343
	Heurística	0,0471	0,0427	0,0426	0,0427	0,0427	0,0427	0,0434
RM	Inteiro	0,0399						0,0399
	Relaxação	0,0341						0,0341
	Heurística	0,0482						0,0482
GAP (%)	Inteiro	1,083	0,318	0,294	0,166	0,223	0,0	0,347
	Relaxação	2,055	0,796	0,701	0,499	0,468	0,467	0,831
	Heurística	-0,89	-10,80	-11,03	-10,78	-10,82	-10,81	-9,19

Tabela 4.4: Valor do critério *c* - 100 requisições - cenário 1 off-line

		Número de Caminhos						Média
		310	1660	2824	4318	5731	7391	3706
CFC	Relaxação	0,1367	0,1367	0,1358	0,1357	0,1357	0,1356	0,1360
	Heurística	0,1494	0,1502	0,1488	0,1505	0,1500	0,1501	0,1498
RM	Relaxação	0,1345						0,1345
	Heurística	0,1588						0,1588
GAP (%)	Relaxação	1,874	1,874	1,211	1,111	1,078	1,073	1,089
	Heurística	-5,85	-5,49	-6,33	-5,25	-5,66	-5,62	-5,70

Tabela 4.5: Valor do critério *c* - 400 requisições - cenário 1 off-line

		Número de Caminhos						Média
		1877	6669	11301	17232	22859	29834	14962
CFC	Relaxação	0,4156	0,4156	0,4146	0,4145	0,4145	0,4145	0,4151
	Heurística	0,4511	0,4501	0,4493	0,4494	0,4476	0,4484	0,4493
RM	Relaxação	0,4129						
	Heurística	0,4862						
GAP (%)	Relaxação	0,633	0,633	0,404	0,385	0,371	0,370	0,466
	Heurística	-7,18	-7,39	-7,57	-7,53	-7,91	-7,76	-7,56

Tabela 4.6: Valor do critério c - 600 requisições - cenário 1 off-line

		Número de Caminhos						Média
		1876	9981	16990	25740	34467	43982	
CFC	Relaxação	0,6010	0,6010	0,6001	0,6000	0,6000	0,6000	0,6004
	Heurística	0,6500	0,6627	0,6558	0,6611	0,6604	0,6576	0,6575
RM	Relaxação	0,5975						0,5975
	Heurística	0,6322						0,6322
GAP (%)	Relaxação	0,598	0,598	0,447	0,426	0,423	0,423	0,486
	Heurística	2,81	4,84	3,75	4,59	4,49	4,04	4,09

4.5.2 Cenário 2

Neste cenário, foi adotada a rede da figura 4.2 com 30 nós e os perfis das demandas foram gerados conforme discutido na seção 4.1. As tabelas 4.7, 4.8, 4.9 e 4.10 apresentam alguns dos resultados obtidos para os experimentos realizados utilizando o grafo de 30 nós, para 15, 100, 400 e 600 requisições respectivamente. Acima de 800 requisições, algumas instâncias apresentaram-se infactíveis, tanto para o modelo CFC como para o RM.

Foram calculados valores médios do critério c correspondentes à execução dos experimentos para a relaxação dos modelos e pelas heurísticas de RM e CFC. Assim como na seção 4.5.1, a primeira coluna da tabela apresenta os valores obtidos utilizando caminhos arcos-disjuntos. As quatro colunas seguintes contêm caminhos adicionais à coluna anterior, mas neste cenário não se apresenta o número máximo de caminhos do grafo, devido ao tempo computacional necessário para computá-los. Além disso, demonstra-se o valor máximo e mínimo para a heurística CFC.

Assim como no cenário 1, o valor obtido com a relaxação do modelo CFC diminuiu com o aumento do número de caminhos. Apesar de pequena, esta diminuição mostra que o número de caminhos influencia na solução. Diferentemente dos valores obtidos para o cenário 1, neste cenário houve uma diferença significativa entre os resultados obtidos com as relaxações do modelo RM e CFC, como pode ser visto na linha que apresenta o GAP, sendo que o modelo RM é melhor que o CFC para relaxação.

Os valores obtidos com a relaxação do modelo CFC são aproximadamente 50% superiores aos obtidos com o modelo RM. Apesar da diferença para as relaxações e do valor superior obtido pelo modelo CFC, a heurística apresentou um valor inferior ao obtido com a heurística RM. A heurística CFC obteve resultados melhores que de RM, mesmo utilizando somente caminhos arcos-disjuntos, a diferença entre elas variando de 4% a 25%. Constatou-se que a alternativa do aumento do número de

Tabela 4.7: Valor do critério c - 15 requisições - cenário 2 off-line

		Número de Caminhos					Média
		52	643	1371	2757	5187	2002
CFC	Relaxação	0,2537	0,0237	0,0234	0,0232	0,0231	0,0694
	Heurística	0,0411	0,0358	0,0350	0,0321	0,0315	0,0351
	Heurística - Máximo	0,0493	0,0448	0,0443	0,0427	0,0419	0,0446
	Heurística - Mínimo	0,0282	0,0268	0,0270	0,0266	0,0265	0,0270
RM	Relaxação	0,0190					0,0190
	Heurística	0,0475					0,0475
	Heurística - Máximo	0,0706					0,0706
	Heurística - Mínimo	0,0382					0,0382
GAP (%)	Relaxação	34,40	25,41	23,78	22,64	22,04	25,65
	Heurística	-11,24	-24,17	-24,82	-31,35	-32,70	-24,86

Tabela 4.8: Valor do critério c - 100 requisições - cenário 2 off-line

		Número de Caminhos					Média
		345	4357	9200	18312	34254	13294
CFC	Relaxação	0,0937	0,0917	0,0912	0,0908	0,0906	,0916
	Heurística	0,1247	0,1230	0,1195	0,1173	0,1196	0,1208
	Heurística - Máximo	0,1450	0,1379	0,1362	0,1392	0,1354	0,1387
	Heurística - Mínimo	0,1107	0,1089	0,1061	0,1050	0,1038	0,1069
RM	Relaxação	0,0647					0,0647
	Heurística	0,1457					0,1457
	Heurística - Máximo	0,1720					0,1720
	Heurística - Mínimo	0,1187					0,1187
GAP (%)	Relaxação	44,58	41,48	40,81	40,20	39,86	41,39
	Heurística	-13,42	-14,63	-17,10	-18,55	-16,70	-16,08

caminhos não apresentou uma melhoria substancial para todos os casos (relaxação e heurística).

Os valores máximo e mínimo apresentados nas tabelas permitem verificar que há grande variação no valor da heurística CFC, o que pode justificar a não diminuição do valor com relação ao aumento do número de caminhos. A utilização de caminhos arcos-disjuntos com a heurística CFC, tanto para o grafo de 10 nós, como para de 30, apresenta-se como uma solução viável para economia de recursos, e o controle sobre o número de caminhos permite obter resultados próximos do ótimo com baixo tempo de processamento. A seguir, estudou-se a influência do coeficiente α nos resultados.

Tabela 4.9: Valor do critério c - 400 requisições - cenário 2 off-line

		Número de Caminhos					Média
		1364	7030	17480	27172	36622	17934
CFC	Relaxação	0,3127	0,3106	0,3091	0,3087	0,3082	0,3099
	Heurística	0,3695	0,3745	0,3714	0,3701	0,3706	0,3712
	Heurística - Máximo	0,3950	0,3992	0,3894	0,3981	0,3830	0,3929
	Heurística - Mínimo	0,3453	0,3485	0,3434	0,3484	0,3463	0,3464
RM	Relaxação	0,2086					0,2086
	Heurística	0,4210					0,4210
	Heurística - Máximo	0,4721					0,4721
	Heurística - Mínimo	0,3888					0,3888
GAP (%)	Relaxação	49,88	48,84	48,14	47,93	47,71	48,50
	Heurística	-12,10	-10,89	-11,66	-11,93	-11,79	-11,67

Tabela 4.10: Valor do critério c - 600 requisições - cenário 2 off-line

		Número de Caminhos					Média
		2045	10612	18491	26140	40878	19633
CFC	Relaxação	0,4520	0,4495	0,4483	0,4476	0,4469	0,4489
	Heurística	0,5197	0,5320	0,5319	0,5281	0,5280	0,5279
	Heurística - Máximo	0,5624	0,5608	0,5612	0,5589	0,5482	0,5583
	Heurística - Mínimo	0,4876	0,5098	0,5061	0,5003	0,4986	0,5000
RM	Relaxação	0,3076					0,3076
	Heurística	0,5505					0,5505
	Heurística - Máximo	0,5783					0,5783
	Heurística - Mínimo	0,5334					0,5334
GAP (%)	Relaxação	46,95	47,09	45,72	45,50	45,27	46,11
	Heurística	-5,61	-3,36	-3,37	-4,04	-4,06	-4,09

4.5.3 Influência de α no cenário 2

As tabelas 4.11, 4.12, 4.13 e 4.14 mostram os valores obtidos para os modelos CFC e RM utilizando os valores de α iguais a 0,0, 0,25, 0,75 e 1,0 respectivamente. O valor de α foi variado somente para o experimento com 100 requisições. Com a variação do valor de α , o comportamento permanece o mesmo. Há uma diminuição do valor da relaxação CFC com o aumento do número de caminhos e a heurística CFC continua mantendo um valor melhor ao obtido utilizando a heurística RM, com a média entre os modelos variando de 15% a 60%.

A figura 4.8 apresenta as curvas da diferença percentual (GAP) entre a heurística CFC e a RM e entre a relaxação CFC e a RM.

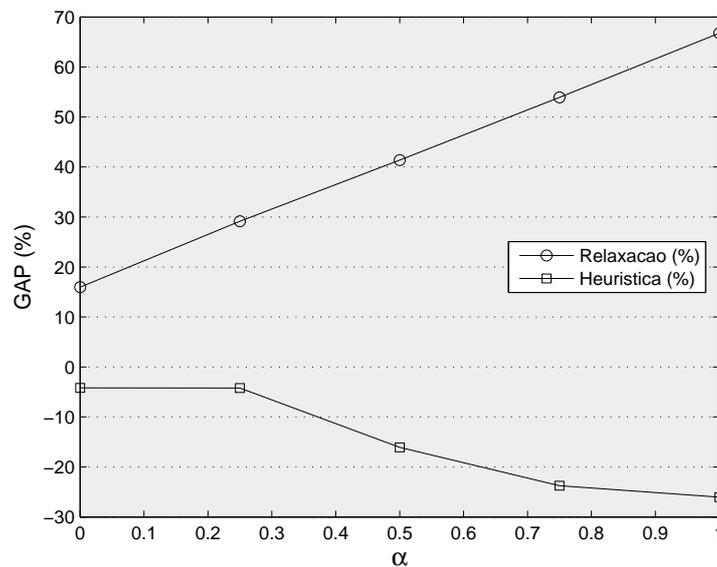


Figura 4.8: Diferença percentual entre os modelos CFC e RM com variação do α

Com o aumento de α , a diferença no valor obtido para as relaxações aumenta, indicando que a solução da relaxação CFC é pior que a de RM. No caso da heurística, ocorre o contrário: os valores obtidos pela heurística melhoram em relação à de RM (valor negativo de % GAP). Os resultados obtidos com a heurística se aproximam do valor obtido pela relaxação RM.

Intuitivamente, o encaminhamento de fluxos tende a utilizar caminhos mais longos quando $\alpha = 1$, pois busca-se minimizar o pico máximo em detrimento dos valores médios. Uma vez que o problema de caminhos mais longos é NP-Difícil, tanto a solução ótima (via programação linear) quanto a aproximativa (via solução heurística) do modelo RM implicitamente resolvem problema de natureza difícil. Por outro lado, a solução exata e heurística do modelo CFC não são dessa dificuldade tendo em vista o pré-cômputo de caminhos. Essa intuição, leva a especular que a qualidade da solução

Tabela 4.11: Valor do critério c - 100 requisições - $\alpha = 0,0$

		Número de Caminhos					Média	
		α	345	4348	9213	18328	34354	39105
CFC	0,0	Relaxação	0,0656	0,0640	0,0634	0,0630	0,0627	0,0637
		Heurística	0,0691	0,0685	0,0685	0,0681	0,0679	0,0684
RM	0,0	Relaxação	0,0549					0,0549
		Heurística	0,0714					0,0714
GAP (%)	0,0	Relaxação	19,35	16,40	15,39	14,54	14,11	15,96
		Heurística	-3,19	-4,03	-4,12	-4,65	-4,88	-4,17

Tabela 4.12: Valor do critério c - 100 requisições - $\alpha = 0,25$

		Número de Caminhos					Média	
		α	345	4343	9183	18339	34322	13306
CFC	0,25	Relaxação	0,0819	0,0794	0,0786	0,0780	0,0778	0,0791
		Heurística	0,1024	0,1040	0,1031	0,1038	0,1037	0,1034
RM	0,25	Relaxação	0,0612					0,0612
		Heurística	0,1085					0,1085
GAP (%)	0,25	Relaxação	33,81	29,63	27,75	27,43	27,06	29,14
		Heurística	-5,20	-3,46	-4,54	-3,95	-3,84	-4,20

RM tende a aumentar com o aumento de α , como indica o gráfico da figura 4.8. Observa-se ainda o distanciamento entre a relaxação e o valor da heurística com o aumento de α , o que é comum em problemas de otimização difícil.

4.6 Análise Experimental de Roteamento On-line

Os experimentos on-line foram divididos em dois cenários. A seção 4.6.1 expõe experimentos computacionais realizados com o grafo de 10 nós do cenário 1 e uma discussão dos resultados obtidos. A seção 4.6.2 apresenta experimentos computacionais realizados com o grafo de 30 nós do cenário 2 e uma discussão dos resultados obtidos. Nos experimentos utilizou-se para o parâmetro α o valor $\frac{1}{2}$ no sentido de não dar preferência exclusiva a nenhum dos componentes do critério. As requisições são geradas previamente contendo o perfil de tráfego para cada fatia de tempo τ , sendo que somente admitidas pela heurística on-line na fatia de tempo correspondente ao início da requisição.

Tabela 4.13: Valor do critério c - 100 requisições - $\alpha = 0,75$

	α		Número de Caminhos					Média
			345	4359	9218	18330	34354	13321
CFC	0,75	Relaxação	0,1050	0,1040	0,1038	0,1036	0,1034	0,1040
		Heurística	0,1468	0,1366	0,1351	0,1319	0,1335	0,1368
RM	0,75	Relaxação	0,0676					0,0676
		Heurística	0,1828					0,1828
GAP (%)	0,75	Relaxação	55,52	54,02	53,61	53,30	53,13	53,92
		Heurística	-18,42	-23,85	-24,64	-26,23	-25,66	-23,76

Tabela 4.14: Valor do critério c - 100 requisições - $\alpha = 1,0$

	α		Número de Caminhos					Média
			345	4368	9202	18325	34265	13301
CFC	1,0	Relaxação	0,1164	0,1163	0,1163	0,1163	0,1163	0,1163
		Heurística	0,1736	0,1576	0,1544	0,1560	0,1528	0,1589
RM	1,0	Relaxação	0,0701					0,0701
		Heurística	0,2200					0,2200
GAP (%)	1,0	Relaxação	66,85	66,73	66,73	66,73	66,73	66,75
		Heurística	-19,14	-26,63	-28,01	-27,51	-28,82	-26,02

4.6.1 Cenário 1

Esta análise experimental visa a demonstrar a viabilidade do modelo baseado em caminhos proposto e compará-lo ao modelo baseado em fluxos [30]. A comparação toma ainda como base limites inferiores computados através da relaxação contínua dos modelos em programação matemática.

Para o cenário 1 foi utilizada a topologia apresentada na figura 4.1. A duração de cada requisição é de 6 fatias de tempo τ com $\theta = 12$ conforme relatado na seção 4.1.

Nos experimentos realizados, a principal preocupação foi a corretude dos algoritmos, existindo, portanto, a oportunidade de melhorar a implementação e eficiência destes. De qualquer forma, as soluções foram obtidas em um tempo de execução relativamente baixo (menos de 1 segundo). A obtenção dos resultados utiliza-se do valor médio de dez execuções do modelo PLI e da heurística com diferentes sementes para o gerador de números aleatórios.

A seguir, serão apresentadas quatro tabelas com os resultados obtidos na experimentação numérica para 20 requisições (tabela 4.15), 100 requisições (tabela 4.16), 400 requisições (tabela 4.17) e 600 requisições (tabela 4.18). As tabelas contêm os valores do critério c para todas as situações. Para

o modelo CFC, apresentaram-se diferentes alternativas de número de caminhos. A primeira coluna contém somente caminhos arcos-disjuntos. As cinco colunas seguintes contêm caminhos adicionais à coluna anterior até atingir o número máximo de caminhos do grafo (última coluna) para cada situação. No caso do modelo RM, somente um valor é apresentado para o critério c , pois o modelo não depende do número de caminhos.

A expressão $GAP = 100 \cdot \frac{(\overline{CFC} - \overline{RM})}{\overline{RM}}$ foi utilizada para obter a diferença percentual entre os valores obtidos para os modelos e é apresentada na tabela como GAP. A comparação é feita entre relaxações e entre heurísticas. Nas comparações, \overline{CFC} indica o valor do critério c com o modelo CFC (relaxação ou heurística), enquanto \overline{RM} indica o valor do critério c obtido com o modelo RM (relaxação ou heurística).

Pode-se observar nas tabelas 4.15, 4.16, 4.17 e 4.18 que, à medida que o número de caminhos aumenta (até atingir o máximo número de caminhos - última coluna), as relaxações CFC se aproximam do valor da relaxação obtida usando o modelo de RM, atingindo uma diferença percentual que varia de aproximadamente 0,5% até aproximadamente 2,1%. Estes resultados indicam que os resultados obtidos com o modelo CFC (PLI) se tornam equivalentes aos do modelo RM (PLI) à medida que caminhos são adicionados.

A heurística do modelo CFC apresenta pouca variação com o aumento do número de caminhos (diferença de aproximadamente 3%). Observa-se que a heurística CFC tem um desempenho superior à heurística RM (valores negativos de GAP), atingindo ganhos percentuais de até 25% para os cenários com poucas requisições. À medida que o número de requisições aumenta, os resultados de ambas as heurísticas se aproximam, com a heurística CFC tendo ainda ganhos relativos da ordem de 1% em relação à heurística RM. Isto nos permite concluir que, com menos requisições, a heurística CFC obtém resultados melhores que a heurística RM e, com o aumento do número de requisições, ambas as heurísticas permitem obter resultados similares.

É importante notar que, utilizando somente caminhos arcos-disjuntos (primeira coluna das tabelas), os valores obtidos pela heurística CFC apresentam bons resultados em relação aos resultados obtidos pela do modelo RM (23% de melhoria para 20 requisições, 1,4% de melhoria para 600 requisições), além de ganho no tempo de processamento utilizando caminhos arcos-disjuntos. Deste modo, o modelo CFC poderia substituir com melhoria o modelo RM para realização do encaminhamento de requisições com perfis variantes no tempo.

Tabela 4.15: Valor do critério c - 20 requisições - cenário 1 on-line

		Número de Caminhos						Média
		61	332	575	884	1166	1512	441
CFC	Relaxação	0,0350	0,0345	0,0343	0,0343	0,0342	0,0342	0,0344
	Heurística	0,0443	0,0421	0,0424	0,0422	0,0422	0,0430	
RM	Relaxação	0,0335						0,0335
	Heurística	0,0589						0,0589
GAP (%)	Relaxação	5,878	4,479	3,886	0,630	0,545	0,537	2,659
	Heurística	-23,765	-27,234	-26,724	-26,952	-27,051	-25,843	-26,224

Tabela 4.16: Valor do critério c - 100 requisições - cenário 1 on-line

		Número de Caminhos						Média
		315	1664	2870	4317	5745	7361	3712
CFC	Relaxação	0,1033	0,1028	0,1026	0,1026	0,1026	0,1026	0,1028
	Heurística	0,1217	0,1222	0,1181	0,1194	0,1205	0,1200	0,1203
RM	Relaxação	0,0992						0,0992
	Heurística	0,1414						0,1414
GAP (%)	Relaxação	4,280	3,783	3,629	3,569	3,565	3,564	3,138
	Heurística	-13,992	-13,557	-16,468	-15,474	-14,735	-15,083	14,885

Tabela 4.17: Valor do critério c - 400 requisições - cenário 1 on-line

		Número de Caminhos						Média
		1255	6708	11359	17208	22862	29345	14790
CFC	Relaxação	0,3468	0,3462	0,3457	0,3457	0,3457	0,3457	0,3460
	Heurística	0,3847	0,3935	0,3946	0,3906	0,3933	0,3905	0,3912
RM	Relaxação	0,3386						0,3386
	Heurística	0,4206						0,4206
GAP (%)	Relaxação	2,536	2,368	2,215	2,196	2,187	2,187	2,282
	Heurística	-8,146	-6,031	-5,787	-6,707	-6,081	-6,733	6,581

Tabela 4.18: Valor do critério c - 600 requisições - cenário 1 on-line

		Número de Caminhos						Média
		1883	9872	17025	25842	34375	43901	
CFC	Relaxação	0,5019	0,5014	0,5013	0,5012	0,5011	0,5011	0,5013
	Heurística	0,5522	0,5550	0,5599	0,5620	0,5569	0,5619	0,5580
RM	Relaxação	0,4962						0,4962
	Heurística	0,5605						0,5605
GAP (%)	Relaxação	1,195	1,095	1,074	1,056	1,047	1,047	0,910
	Heurística	-1,416	-0,908	-0,035	0,337	-0,584	0,306	-0,598

4.6.2 Cenário 2

Neste cenário, foi adotada a rede da figura 4.2 com 30 nós e os perfis das demandas foram gerados conforme discutido na seção 4.1. As tabelas 4.19, 4.20, 4.21 e 4.22 apresentam alguns dos resultados obtidos para os experimentos realizados utilizando o grafo de 30 nós, para 20, 100, 400 e 600 requisições respectivamente. Acima de 800 requisições, algumas instâncias apresentaram infactíveis, tanto para o modelo CFC como para o RM.

Foram calculados valores médios do critério c correspondentes à execução dos experimentos para a relaxação dos modelos e pelas heurísticas de RM e CFC. Assim como na seção 4.6.1, a primeira coluna da tabela apresenta os valores obtidos utilizando caminhos arcos-disjuntos. As 5 colunas seguintes contêm caminhos adicionais à coluna anterior, mas, neste cenário, não é apresentado o número máximo de caminhos do grafo, devido ao tempo computacional necessário para computá-los.

Os valores obtidos com a relaxação do modelo CFC diminuem com o aumento do número de caminhos. Apesar de pequena, esta diminuição mostra que o número de caminhos influencia na solução. Este cenário apresentou uma diferença mais significativa que o caso do cenário 1 entre os resultados obtidos com as relaxações do modelo RM e CFC, como pode ser visto na linha que apresenta o GAP, sendo que o modelo RM é melhor que o CFC para relaxação (22,22% no melhor caso).

Os valores obtidos com a relaxação do modelo CFC são em 20% a 50% superiores aos obtidos com o RM. Mesmo havendo esta diferença para as relaxações, a heurística CFC apresentou um valor inferior ao obtido com a heurística RM. A primeira obteve resultados melhores que a segunda até quando utilizados somente caminhos arcos-disjuntos, com uma diferença variando entre 7% a 30%. Do mesmo modo que para o roteamento off-line, constatou-se que a alternativa do aumento do número de caminhos não apresenta uma melhoria substancial para todos os casos (relaxação e heurística).

Tabela 4.19: Valor do critério c - 20 requisições - cenário 2 on-line

		Número de Caminhos						Média
		67	894	1907	2880	3787	7113	2775
CFC	Relaxação	0,0282	0,0263	0,0260	0,0259	0,259	0,0258	0,0264
	Heurística	0,0442	0,0400	0,0379	0,0379	0,0375	0,0366	0,0390
RM	Relaxação	0,0209						0,0209
	Heurística	0,0502						0,0502
GAP (%)	Relaxação	34,52	24,70	23,47	22,96	22,71	22,22	25,10
	Heurística	-11,63	-20,15	-24,45	-24,56	-25,37	-27,39	-22,26

Tabela 4.20: Valor do critério c - 100 requisições - cenário 2 on-line

		Número de Caminhos						Média
		339	4354	9241	13874	18289	34253	13392
CFC	Relaxação	0,0777	0,0755	0,0751	0,0749	0,0747	0,0746	0,0754
	Heurística	0,1054	0,1003	0,1009	0,0984	0,0994	0,0970	0,1002
RM	Relaxação	0,0545						0,0545
	Heurística	0,1242						0,1242
GAP (%)	Relaxação	42,49	38,505	37,71	37,38	37,06	36,78	38,32
	Heurística	-13,97	-18,28	-17,59	-19,65	-19,06	-20,92	-18,25

A utilização de caminhos arcos-disjuntos com a heurística CFC, tanto para o grafo de 10 nós, como para o grafo de 30 nós, apresenta-se como uma solução de qualidade para economia de recursos e o controle sobre o número de caminhos permite obter resultados próximos do ótimo com baixo tempo de processamento.

4.7 Conclusões

O objetivo deste capítulo foi apresentar os experimentos realizados utilizando o modelo baseado em caminhos para roteamento com perfis variantes no tempo proposto no capítulo 3 e compará-lo ao modelo proposto por Ricciato e Monaco [30]. Apresentaram-se as condições experimentais, um estudo sobre a influência do parâmetro α nos critérios c_{max} e c_{mean} e roteamento fixo *versus* variável. Além disso, apresentaram-se o impacto da granularidade do período de tempo na aceitação de demandas e uma análise experimental para o roteamento off-line e online utilizando dois cenários.

A heurística RM apresenta maior aceitação de demandas que a CFC e ambas apresentam redução de aproximadamente 3% na aceitação quando aumenta o número de períodos de tempo. O

Tabela 4.21: Valor do critério c - 400 requisições - cenário 2 on-line

		Número de Caminhos						Média
		1362	17413	36815	55316	73254	136448	53435
CFC	Relaxação	0,2651	0,2627	0,2621	0,2619	0,2617	0,2616	0,2625
	Heurística	0,3085	0,3144	0,3141	0,3112	0,3101	0,3098	0,3114
RM	Relaxação	0,1784						0,1784
	Heurística	0,3642						0,3642
GAP (%)	Relaxação	48,62	47,26	46,93	46,80	46,71	35,15	45,25
	Heurística	-14,89	-13,25	-13,36	-14,09	-14,52	-22,26	15,40

Tabela 4.22: Valor do critério c - 600 requisições - cenário 2 on-line

		Número de Caminhos						Média
		2045	26174	40967	55281	82993	109647	52851
CFC	Relaxação	0,3849	0,3823	0,3819	0,3817	0,3815	0,3813	0,3823
	Heurística	0,4457	0,4467	0,4461	0,4432	0,4406	0,4406	0,4438
RM	Relaxação	0,2573						0,2573
	Heurística	0,4812						0,4812
GAP (%)	Relaxação	49,63	48,62	48,47	48,39	48,30	48,23	48,61
	Heurística	-7,32	-7,10	-7,25	-7,85	-8,37	-8,42	-7,72

roteamento fixo pode ser utilizado a um custo menor que o variável obtendo soluções de mesma qualidade. Os experimentos demonstraram que o modelo CFC pode obter soluções de qualidade similar as obtidas pelo modelo RM, podendo substituí-lo com controle sobre a qualidade da solução.

No capítulo 5, realizou-se uma introdução sobre o tratamento de restrições no roteamento, uma nova formulação com suporte a múltiplas restrições e um exemplo ilustrativo que utiliza uma restrição adicional.

Capítulo 5

Tratamento de Restrições no Problema de Roteamento

Nesta seção, são apresentados conceitos sobre restrições e seus tratamentos para o problema de roteamento. Anteriormente, considerou-se no processo de otimização, somente a largura de banda. Entretanto, é interessante considerar-se outras restrições associadas às de QoS, tais como atraso, variação do atraso, perda de pacotes e número de saltos, para poder garantir a qualidade de serviço. Em um problema de otimização, as restrições podem ser classificadas em não aditivas e aditivas.

Restrições não aditivas são definidas por estarem limitadas pelo menor valor (ou maior) de uma determinada medida encontrada em um arco pertencente a um trajeto [33]. Um exemplo de restrição não aditiva em redes é a largura de banda de uma conexão, pois a restrição é limitada pelo arco com menor capacidade. Por outro lado, restrições aditivas em redes são definidas pela soma dos valores de cada arco no trajeto [4, 24, 33, 36].

Além das restrições aditivas e não aditivas, há os objetivos aditivos e não aditivos. Objetivo não aditivo busca minimizar um custo dado pelo arco de maior valor ou maximizar um benefício dado pelo arco de menor valor [33]. Um exemplo de objetivo não aditivo é quando se deseja maximizar a largura de banda a ser utilizada para uma conexão. Assim, a largura de banda é determinada pelo arco com menor disponibilidade de largura de banda ao longo do trajeto desta conexão. O objetivo aditivo é aquele que busca minimizar ou maximizar um valor qualquer relacionado a todos os arcos de um caminho em um grafo. Como exemplo de objetivo aditivo, pode-se mencionar a minimização do número de saltos ou do atraso.

As restrições podem ainda ser classificadas diferentemente, baseadas em caminho ou em arco.

Uma restrição aditiva é dita de caminho se considerar a soma dos valores dos arcos que compõem o caminho, e é dita de arco se considera a soma dos recursos em cada arco. Uma restrição não aditiva é dita de caminho quando considera para um dado recurso o mínimo dos valores de todos os arcos do caminho, e é dita de arco quando considera o máximo ou mínimo de recursos de todos os valores do arco.

A seguir apresentam-se alguns exemplos:

- Restrição baseada em caminho aditiva: o tempo de transmissão de uma requisição é definido pela soma dos tempos ao longo dos arcos pertencentes ao caminho;
- Restrição baseada em arco aditiva: o consumo de banda de um arco é dado pela soma das reservas de todas as requisições que atravessam o arco;
- Restrição baseada em caminho não aditiva: a probabilidade máxima de perda de pacote de uma conexão é dada pelo arco com maior probabilidade de perda; e
- Restrição baseada em arco não aditiva: a prioridade máxima das requisições encaminhadas através de um arco é dada pela requisição de maior importância, ou seja, uma restrição para restringir a prioridade das requisições encaminhadas através do arco pode ser empregada em arcos de baixa confiabilidade, onde não se deseja encaminhar requisições de alta prioridade. Desta forma, somente requisições de baixa prioridade são encaminhadas no arco.

A formulação proposta por Ricciato e Monaco [30], apresentada na seção 2.3.1, não considera a utilização de restrições adicionais. Para adicionar tais restrições, o modelo baseado em fluxos se tornaria muito mais complexo. No novo modelo apresentado neste trabalho, é possível tratar as restrições aditivas baseadas em caminhos de forma implícita, como mencionado anteriormente e como se mostrará a seguir. Este tratamento é realizado no momento em que os caminhos são computados. Do mesmo modo que caminhos que não possuem largura de banda disponível suficiente para atender a requisição são eliminados pela heurística, podem ser eliminados por não atenderem aos requisitos exigidos pelas restrições adicionais de cada serviço. Por exemplo, para uma requisição que necessita de atraso mínimo de $10ms$, caminhos que não possuem atraso menor ou igual a esse tempo são eliminados da lista de caminhos disponíveis para encaminhar a requisição. Os que satisfazem as restrições baseadas em caminhos (aditivas e não-aditivas) podem ser pré-computados através de algoritmos de programação dinâmica ou programação inteira [33].

Pode ser ainda que seja necessário considerar restrições cujos custos são variáveis aleatórias. Um exemplo é o atraso de comunicação entre dois nodos de uma rede que pode variar conforme a

utilização do enlace ou a qualidade do sinal em uma rede de comunicação sem fio (*wireless*) quando utilizada por um dispositivo móvel. Essas restrições podem ser tratadas explicitamente pelo algoritmo que computa os caminhos do grafo. Shima [33] apresenta a aplicação de restrições probabilísticas em caminhos mínimos.

5.1 Revisão da Literatura

O uso de restrições aditivas é comum em problemas de roteamento com restrições. A seção 2.1 apresentou as principais métricas utilizadas no roteamento com QoS: atraso, variação do atraso, largura de banda, perda de pacotes e número de saltos. Vários trabalhos utilizam no máximo duas restrições em seus algoritmos. Em geral buscam minimizar a largura de banda ou o custo e utilizam as métricas mencionadas anteriormente como restrições [3, 11, 17, 34].

Para realizar o roteamento com QoS, necessita-se selecionar caminhos que respeitem a restrições impostas pelas métricas de QoS. Computar caminhos que atendam às métricas de QoS requeridas por uma aplicação se enquadra como um problema de caminhos mínimos sob restrições [33].

Kuipers *et al.* [18] consideraram algoritmos para obtenção de caminhos mínimos sob múltiplas restrições e, de forma geral, as restrições aditivas como, por exemplo, o atraso. Para o problema de caminhos sob múltiplas restrições, se para todas as restrições o somatório dos pesos dos arcos pertencentes ao caminho da origem s até o destino d for menor que o valor da restrição, este caminho é considerado factível. Este problema pode ser especializado para obter o menor caminho entre os considerados factíveis. Apresentam alguns algoritmos aproximados para obtenção da solução.

Korkmaz *et al.* [17] apresentam uma formulação similar a de Kuipers *et al.* [18], mas especializam para duas restrições aditivas não negativas quaisquer. O algoritmo tem como objetivo buscar um caminho factível (utilizando *Dijkstra*) através da minimização de uma função custo.

Fortz e Thorup [11] otimizam a definição de pesos dos enlaces com objetivo de minimizar o congestionamento. Minimizam a máxima utilização dos enlaces e consideram uma função custo que penaliza enlaces congestionados.

Banerjee e Sidhu [3] consideraram o cômputo de caminhos para requisições tendo como requisitos largura de banda e atraso, com o objetivo de aumentar o rendimento da rede limitando custo e distribuindo a carga da rede.

Wang e Crowcroft [36] utilizam a largura de banda e o atraso como métricas de QoS. Selecionam o caminho eliminando enlaces que não possuem largura de banda residual suficiente para

atender a demanda, de forma que qualquer caminho no grafo resultante que satisfaça a requisição. Em seguida, buscam um caminho com atraso mínimo utilizando o algoritmo de *Dijkstra* de forma a respeitar a restrição.

Lin e Shroff [20] apresentam um modelo que visa a maximizar o rendimento, que pode ser monetário ou a preferência por uma classe de serviço. Consideram a capacidade do enlace como métrica de QoS e formulam a restrição baseado nela. No caso da utilização de múltiplas métricas de QoS, consideram que cada nó da rede pode fornecer garantia de atraso ou probabilidade de perdas para todos os fluxos pertencentes a uma mesma classe e não para cada fluxo separadamente. Desta forma, assumem que cada enlace fornecerá garantia de QoS para a métrica largura de banda e atraso e/ou probabilidade de perdas.

Orda e Sprintson [27] consideram as métricas como sendo o peso dos arcos de um grafo, mas são definidas de forma diferenciada para cada tipo de restrição. Por exemplo, é definida de uma forma para restrições aditivas e de outra para largura de banda. As métricas utilizadas são: número de saltos e custo dos enlaces. Consideram que um caminho restringe a requisição quando o seu peso é menor que o solicitado. Com base nas métricas, obtêm o caminho de menor custo que satisfaça os requisitos de QoS. Apresentam um esquema de pré-cômputo de caminhos para uma posterior minimização do número de saltos tendo a largura de banda como restrição. Também aplicaram este esquema de pré-cômputo com o uso de restrições aditivas. Demonstraram que o pré-cômputo oferece uma melhora em relação a outras abordagens padrões.

Todas as métricas de QoS utilizadas nos trabalhos citados nos parágrafos anteriores podem ser solucionadas de forma explícita ou implícita pela heurística CFC proposta. Estas métricas são modeladas como restrições através de inequações que limitam o valor máximo ou mínimo da métrica para um caminho ou enlace.

5.2 Modelos Matemáticos com Múltiplas Restrições

Nesta seção expõem-se as extensões do modelo CFC fixo (equações (3.1a) – (3.1h)) e do RM fixo (equações (2.2a) – (2.2j)) para tratamento de múltiplas restrições aditivas e não aditivas. Note que a função objetivo continua a mesma, levando em conta a largura de banda, mas os modelos estendidos passam a considerar restrições que não podiam ser tratadas pelos modelos apresentados anteriormente.

5.2.1 Modelo CFC Estendido

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (5.1a)$$

Sujeito a :

$$\sum_{l \in \mathcal{L}(k)} r_l^k = 1, \quad k \in \mathcal{K} \quad (5.1b)$$

$$u_{ij}^h(\tau) = \sum_{k \in \mathcal{K}} \sum_{l \in \mathcal{L}(k)} \{f_{ij}^{k,h}(\tau) \cdot r_l^k \text{ se } (i,j) \in P_l^k\}, \quad (i,j) \in E, \tau \in \Theta, h \in \mathcal{H}_{ad}^{arc} \quad (5.1c)$$

$$u_k^h(\tau) = \sum_{l \in \mathcal{L}(k)} \sum_{(i,j) \in P_l^k} f_{ij}^h(\tau) \cdot r_l^k, \quad \tau \in \Theta, k \in \mathcal{K}, h \in \mathcal{H}_{ad}^{path} \quad (5.1d)$$

$$u_k^h(\tau) = \sum_{l \in \mathcal{L}(k)} \max\{f_{ij}^h(\tau) : (i,j) \in P_l^k\} \cdot r_l^k, \quad \tau \in \Theta, k \in \mathcal{K}, h \in \mathcal{H}_{nad}^{path} \quad (5.1e)$$

$$v_{ij}^h \geq u_{ij}^h(\tau), \quad (i,j) \in E, \tau \in \Theta, h \in \mathcal{H}_{ad}^{arc} \quad (5.1f)$$

$$v_{ij}^h \leq \lambda_{ij}^h, \quad (i,j) \in E, h \in \mathcal{H}_{ad}^{arc} \quad (5.1g)$$

$$v_k^h \geq u_k^h(\tau), \quad \tau \in \Theta, k \in \mathcal{K}, h \in \mathcal{H}_{ad}^{path} \cup \mathcal{H}_{nad}^{path} \quad (5.1h)$$

$$v_k^h \leq \lambda_k^h, \quad k \in \mathcal{K}, h \in \mathcal{H}_{ad}^{path} \cup \mathcal{H}_{nad}^{path} \quad (5.1i)$$

$$c_{max} \geq \frac{1}{\lambda_{ij}^1} \cdot v_{ij}^1, \quad (i,j) \in E \quad (5.1j)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i,j) \in E} \frac{1}{\lambda_{ij}^1} \cdot v_{ij}^1, \quad (5.1k)$$

$$r_l^k \in \{0, 1\}, \quad k \in \mathcal{K}, l \in \mathcal{L}(k) \quad (5.1l)$$

onde:

- \mathcal{H}_{ad}^{arc} é o conjunto de índices das restrições aditivas sobre arcos;
- \mathcal{H}_{ad}^{path} é o conjunto de índices das restrições aditivas sobre caminhos;
- \mathcal{H}_{nad}^{path} é o conjunto de índices das restrições não aditivas sobre caminhos;
- r_l^k assume o valor 1 se a requisição k é encaminhada pelo caminho l e 0 caso contrário;
- $u_{ij}^h(\tau)$ é a reserva do recurso h no enlace (i, j) para o período τ ;
- $f_{ij}^{k,h}(\tau)$ é o perfil de consumo do recurso h correspondente à requisição k durante o período τ ;
- $f_{ij}^h(\tau)$ é o perfil de consumo do recurso h no arco (i, j) durante o período τ ;

- $u_k^h(\tau)$ é a reserva do recurso h pela requisição k para o período τ ;
- v_{ij}^h é a máxima reserva do recurso aditivo h para o enlace (i, j) considerando todos os períodos de tempo;
- λ_{ij}^h é a quantidade de recurso aditivo h disponível no arco (i, j) ;
- v_k^h é a máxima reserva do recurso h para a requisição k considerando todos os períodos de tempo;
- λ_k^h é a quantidade de recurso h disponível para a requisição k ;
- c_{max} é a máxima reserva percentual de todos os picos de banda ($h = 1$) considerando todos os enlaces e períodos de tempo;
- c_{mean} é a média percentual dos picos de banda para todos os períodos e enlaces;
- $|E|$ é a cardinalidade de E , ou seja, o número de enlaces;
- P_l^k é o l -ésimo caminho do conjunto de caminhos disponíveis para encaminhamento da k -ésima requisição.
- $\mathcal{L}(k) = \{1, 2, \dots, l(k)\}$ corresponde ao conjunto de índices dos caminhos tendo origem s_k e destino d_k e $l(k)$ é o número de caminhos disponíveis para a requisição k

Para um caso particular, assume-se que $h = 1$ corresponde à disponibilidade de banda dos enlaces, ou seja, $1 \in \mathcal{H}_{ad}^{arc}$. Como exemplo, $\mathcal{H}_{ad}^{arc} = \{1, \dots, p_1\}$, $\mathcal{H}_{ad}^{path} = \{p_1 + 1, \dots, p_2\}$, $\mathcal{H}_{nad}^{path} = \{p_2 + 1, \dots, p_3\}$ com p_1, p_2 e $p_3 \in \mathbb{Z}$. Define-se que $h = 1$ corresponde à disponibilidade de banda nos enlaces, $h = 2$ corresponde ao atraso no caminho P_l^k ($\mathcal{H}_{ad}^{path} = \{2\}$) e $h = 3$ corresponde à confiabilidade do caminho P_l^k ($\mathcal{H}_{nad}^{path} = \{3\}$);

A função objetivo (5.1a) é a mesma apresentada na seção 2.3.2. A equação (5.1b) assegura que cada requisição k será encaminhada por um único caminho l . A equação (5.1c) estabelece que $u_{ij}^h(\tau)$ é a quantidade de recurso $h \in \mathcal{H}_{ad}^{arc}$ consumido em cada enlace (i, j) na fatia de tempo τ , sendo esta dada pela soma das demandas $f_{ij}^k(\tau)$ de cada requisição k se o enlace pertencer ao caminho P_l^k , e se este caminho for selecionado para o encaminhamento. A equação (5.1d) estabelece que $u_k^h(\tau)$ é a quantidade de recurso $h \in \mathcal{H}_{ad}^{path}$ consumido pela requisição k durante a fatia de tempo τ , sendo esta dada pela soma das demandas $f_{ij}^k(\tau)$ para os arcos (i, j) do caminho P_l^k selecionado para encaminhamento da requisição k . A equação (5.1e) estabelece que $u_k^h(\tau)$ é a quantidade de recurso $h \in \mathcal{H}_{nad}^{path}$ consumido pela requisição k durante a fatia de tempo τ , sendo esta dada pela soma das

demandas $f_{ij}^h(\tau)$ para os arcos (i, j) do caminho P_l^k selecionado para encaminhamento da requisição k . A desigualdade (5.1f) define o pico do recurso h para cada enlace (i, j) e a desigualdade (5.1g) garante que o pico não ultrapassa a quantidade de recurso h disponível para o enlace (i, j) . A desigualdade (5.1h) define o pico do recurso h para cada requisição k e a desigualdade (5.1i) garante que o pico não ultrapassa a quantidade de recurso h disponível para a requisição k . As desigualdades (5.1j) e (5.1k) definem respectivamente o valor máximo e a média de uso de recurso $h = 1$ em cada enlace (i, j) .

5.2.2 Modelo RM Estendido

$$\text{Minimizar } c = \alpha \cdot c_{max} + (1 - \alpha) \cdot c_{mean} \quad (5.2a)$$

Sujeito a :

$$\sum_{j:(s_k, j) \in E} r_{s_k j}^k = 1, \quad k \in \mathcal{X} \quad (5.2b)$$

$$\sum_{i:(i, d_k) \in E} r_{i d_k}^k = 1, \quad k \in \mathcal{X} \quad (5.2c)$$

$$\sum_{j:(i, j) \in E} r_{ij}^k - \sum_{j:(j, i) \in E} r_{ji}^k = 0, \quad k \in \mathcal{X}, i \in V \setminus \{s_k, d_k\} \quad (5.2d)$$

$$u_{ij}^h(\tau) = \sum_{k \in \mathcal{X}} f_{ij}^{k,h}(\tau) \cdot r_{ij}^k, \quad (i, j) \in E, \tau \in \Theta, h \in \mathcal{H}_{ad}^{arc} \quad (5.2e)$$

$$u_k^h(\tau) = \sum_{(i, j) \in E} f_{ij}^h(\tau) \cdot r_{ij}^k, \quad \tau \in \Theta, k \in \mathcal{X}, h \in \mathcal{H}_{ad}^{path} \quad (5.2f)$$

$$u_k^h(\tau) = \max\{f_{ij}^h(\tau) \cdot r_{ij}^k : (i, j) \in E\}, \quad \tau \in \Theta, k \in \mathcal{X}, h \in \mathcal{H}_{nad}^{path} \quad (5.2g)$$

$$v_{ij}^h \geq u_{ij}^h(\tau), \quad (i, j) \in E, \tau \in \Theta, h \in \mathcal{H}_{ad}^{arc} \quad (5.2h)$$

$$v_{ij}^h \leq \lambda_{ij}^h, \quad (i, j) \in E, h \in \mathcal{H}_{ad}^{arc} \quad (5.2i)$$

$$v_k^h \geq u_k^h(\tau), \quad k \in \mathcal{X}, \tau \in \Theta, h \in \mathcal{H}_{ad}^{path} \cup \mathcal{H}_{nad}^{path} \quad (5.2j)$$

$$v_k^h \leq \lambda_k^h, \quad k \in \mathcal{X}, h \in \mathcal{H}_{ad}^{path} \cup \mathcal{H}_{nad}^{path} \quad (5.2k)$$

$$c_{max} \geq \frac{1}{\lambda_{ij}^1} \cdot v_{ij}^1, \quad (i, j) \in E \quad (5.2l)$$

$$c_{mean} = \frac{1}{|E|} \cdot \sum_{(i, j) \in E} \frac{1}{\lambda_{ij}^1} \cdot v_{ij}^1, \quad (5.2m)$$

$$r_{ij}^k \in \{0, 1\}, \quad k \in \mathcal{X}, (i, j) \in E \quad (5.2n)$$

onde:

- \mathcal{H}_{ad}^{arc} é o conjunto de índices das restrições aditivas sobre arcos;
- \mathcal{H}_{ad}^{path} é o conjunto de índices das restrições aditivas sobre caminhos;
- \mathcal{H}_{nad}^{path} é o conjunto de índices das restrições não aditivas sobre caminhos;
- r_{ij}^k assume o valor 1 se a requisição k for encaminhada ao longo de um caminho que atravessa o arco (i, j) e caso contrário, assume valor 0;
- $u_{ij}^h(\tau)$ é a reserva do recurso h no enlace (i, j) para o período τ ;
- $f^{k,h}(\tau)$ é o perfil do recurso h correspondente à requisição k durante o período τ ;
- $f_{ij}^h(\tau)$ é o perfil do recurso h no arco (i, j) durante o período τ ;
- $u_k^h(\tau)$ é a reserva do recurso h pela requisição k para o período τ ;
- v_{ij}^h é a máxima reserva do recurso h para o enlace (i, j) considerando todos os períodos de tempo;
- λ_{ij}^h é o quantidade de recurso h disponível no arco (i, j) ;
- v_k^h é a máxima reserva do recurso h para a requisição k considerando todos os períodos de tempo;
- λ_k^h é a quantidade de recurso h disponível para a requisição k ;
- c_{max} é a máxima reserva percentual de todos os picos do recurso $h = 1$ (banda) considerando todos os enlaces e períodos de tempo;
- c_{mean} é a média percentual dos picos do recurso $h = 1$ (banda) para todos os períodos e enlaces;
e
- $|E|$ é a cardinalidade de E , ou seja, o número de enlaces.

Assim como para o modelo CFC estendido, para um caso particular assume-se que $h = 1$ corresponde à disponibilidade de banda dos enlaces, ou seja, $1 \in \mathcal{H}_{ad}^{arc}$. Como exemplo, $\mathcal{H}_{ad}^{arc} = \{1, \dots, p_1\}$, $\mathcal{H}_{ad}^{path} = \{p_1 + 1, \dots, p_2\}$, $\mathcal{H}_{nad}^{path} = \{p_2 + 1, \dots, p_3\}$ com p_1, p_2 e $p_3 \in \mathcal{Z}$. Define-se que $h = 1$ corresponde à disponibilidade de banda nos enlaces, $h = 2$ corresponde ao atraso no caminho P_1^k ($\mathcal{H}_{ad}^{path} = \{2\}$) e $h = 3$ corresponde à confiabilidade do caminho P_1^k ($\mathcal{H}_{nad}^{path} = \{3\}$);

A função objetivo (5.2a) é a mesma apresentada na seção 2.3.2. As restrições (5.2b) e (5.2c) asseguram que exatamente um caminho será estabelecido entre o nó ingresso s_k e egresso d_k para cada

requisição k , enquanto a restrição (5.2d) se refere aos nós intermediários. A equação (5.2e) estabelece que $u_{ij}^h(\tau)$ é a quantidade máxima de recurso $h \in \mathcal{H}_{ad}^{arc}$ reservado para o enlace (i, j) durante a fatia de tempo τ para as requisições k que fazem uso do enlace (i, j) . A equação (5.2f) estabelece que $u_k^h(\tau)$ é a reserva do recurso $h \in \mathcal{H}_{ad}^{path}$ para cada requisição k definida pela soma $f_{ij}^h(\tau)$ dos enlaces (i, j) pelos quais atravessa a requisição k . A equação (5.2g) estabelece que $u_k^h(\tau)$ é a quantidade máxima de recurso $h \in \mathcal{H}_{nad}^{path}$ consumido pela requisição k , dado pelo arco (i, j) com maior consumo pelo qual atravessa a requisição k . A desigualdade (5.2h) define o pico do recurso h para cada enlace (i, j) durante todos os períodos τ . A desigualdade (5.2i) garante que o pico do recurso h em cada enlace (i, j) não ultrapassa a quantidade de recurso disponível para o enlace (i, j) . A desigualdade (5.2j) define o pico do recurso h para cada requisição k para todos os períodos τ . A desigualdade (5.2k) garante que o pico do recurso h para cada requisição k não ultrapassa a quantidade de recurso disponível para a requisição k . As desigualdades (5.2l) e (5.2m) definem respectivamente o valor máximo e a média de uso de recurso h em cada enlace.

5.3 Exemplo ilustrativo

Esta seção apresenta um exemplo ilustrativo de utilização de restrições adicionais. Para isso realizou-se um ensaio tendo como restrição adicional o número de saltos pelos quais os pacotes podem transitar.

5.3.1 Cenário

O ensaio foi realizado utilizando o grafo da figura 5.1, que apresenta junto aos arcos suas capacidades.

Cinco requisições entre o nó origem 2 e o destino 6 foram geradas demandando 10 unidades de banda cada. Em seguida, outras cinco requisições foram geradas entre o nó origem 5 e destino 1 demandando 10 unidades de banda cada. O modelo PLI CFC (equações (3.1a) – (3.1h)) foi solucionado utilizando o CPLEX [15], com $\alpha = \frac{1}{2}$, $\Theta = 1$ e tendo como restrição 4 saltos¹. Para o par de nós 2 e 6, existem 5 caminhos nós-disjuntos possíveis e para o par de nós 5 e 1 existem 6 caminhos nós-disjuntos possíveis, ambos com o número de saltos variando de 3 a 6.

Neste exemplo, a restrição de número de saltos é aditiva. No contexto do modelo matemático apresentado na seção 5.2 esta restrição é representada pelo conjunto \mathcal{H}_{ad}^{path} .

¹Note que a situação do modelo CFC estendido (5.1a) – (5.1l) pode ser feita com o modelo CFC (5.1a) – (5.1h), bastando utilizar apenas caminhos que satisfazem a restrição de número de saltos.

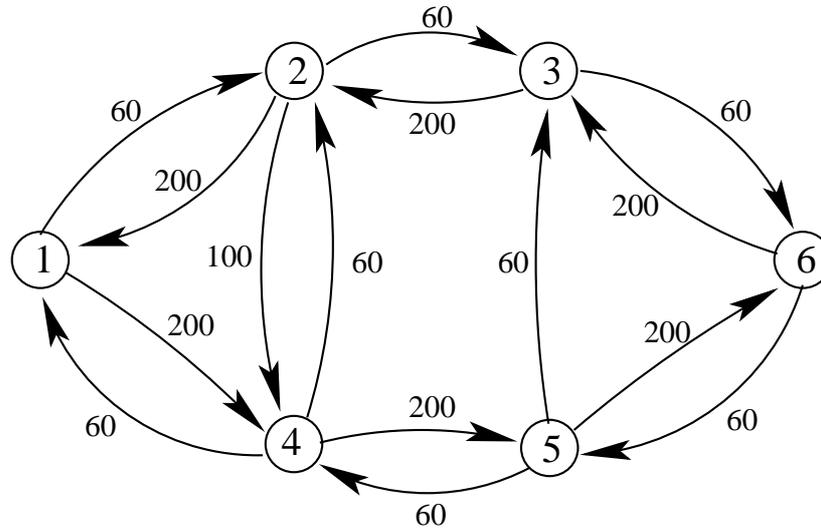


Figura 5.1: Grafo de 6 nós para ensaio com restrição de saltos

5.3.2 Resultados Numéricos

A tabela 5.1 mostra a solução para as 10 requisições utilizando o modelo CFC com todos os caminhos sem restrição, com a restrição de número de saltos e para o modelo RM sem restrição.

Pode-se observar que os modelos RM e CFC sem restrições não garantem o número de saltos desejado (4), como era esperado, enquanto o modelo CFC com restrições atende os requisitos de número máximo de saltos utilizando a mesma heurística. No caso do modelo RM seria necessário a implementação de uma nova heurística.

A figura 5.2 apresenta para cada enlace (i, j) a máxima reserva de banda (v_{ij}^1) e a capacidade (λ_{ij}^1) para o modelo CFC com restrição do número de saltos. Para o caso anterior, 5 requisições entre os nós 2 e 6, de 10 unidades cada são encaminhadas ao longo do caminho representado pelos arcos $2 \rightarrow 4 \rightarrow 5 \rightarrow 6$ (arcos tracejados), totalizando 50 unidades. Três requisições entre os nós 5 e 1, de 10 unidades cada são encaminhadas ao longo do caminho representado pelos arcos $5 \rightarrow 3 \rightarrow 2 \rightarrow 1$ (arcos traços-pontilhados), totalizando 30 unidades. Duas requisições são encaminhadas do nó 5 para o nó 1 ao longo do caminho representado pelos arcos $5 \rightarrow 4 \rightarrow 1$ (arcos pontilhados), totalizando 20 unidades. A soma das unidades encaminhadas corresponde às 100 unidades que se desejava encaminhar.

A figura 5.3 apresenta para cada enlace (i, j) a máxima reserva de banda (v_{ij}^1) e a capacidade (λ_{ij}^1) para o modelo RM (sem restrição no número de saltos), pois não é possível tratar esta restrição implicitamente pela heurística. Um fluxo de 20 unidades é encaminhado do nó 2 para o nó 6 ao longo do caminho $2 \rightarrow 1 \rightarrow 4 \rightarrow 5 \rightarrow 6$. Um fluxo de 20 unidades é encaminhado do nó 2 para o nó 6 ao

Tabela 5.1: Caminhos utilizados para solução dos modelos

	Sem restrição		Com restrição (4 saltos)
2 → 6	CFC	RM	CFC
req. 1	2 → 4 → 5 → 6	2 → 1 → 4 → 5 → 6	2 → 4 → 5 → 6
req. 2	2 → 1 → 4 → 5 → 6	2 → 4 → 5 → 6	2 → 4 → 5 → 6
req. 3	2 → 3 → 6	2 → 1 → 4 → 5 → 6	2 → 4 → 5 → 6
req. 4	2 → 4 → 5 → 6	2 → 3 → 6	2 → 4 → 5 → 6
req. 5	2 → 4 → 5 → 6	2 → 4 → 5 → 6	2 → 4 → 5 → 6
5 → 1			
req. 6	5 → 6 → 3 → 2 → 1	5 → 4 → 1	5 → 3 → 2 → 1
req. 7	5 → 3 → 2 → 1	5 → 3 → 2 → 1	5 → 3 → 2 → 1
req. 8	5 → 6 → 3 → 2 → 1	5 → 6 → 3 → 2 → 1	5 → 4 → 1
req. 9	5 → 4 → 1	5 → 6 → 3 → 2 → 1	5 → 4 → 1
req. 10	5 → 3 → 2 → 1	5 → 3 → 2 → 1	5 → 3 → 2 → 1

longo do caminho $2 \rightarrow 4 \rightarrow 5 \rightarrow 6$. Um fluxo de 10 unidades é encaminhado do nó 2 para o nó 6 ao longo do caminho $2 \rightarrow 3 \rightarrow 6$. Um fluxo de 20 unidades é encaminhado do nó 5 para o nó 1 ao longo do caminho $5 \rightarrow 6 \rightarrow 3 \rightarrow 2 \rightarrow 1$. Um fluxo de 20 unidades é encaminhado do nó 5 para o nó 1 ao longo do caminho $5 \rightarrow 3 \rightarrow 2 \rightarrow 1$. Um fluxo de 10 unidades é encaminhado do nó 5 para o nó 1 ao longo do caminho $5 \rightarrow 4 \rightarrow 1$.

5.3.3 Discussão

Para obter a solução de encaminhamento com múltiplas restrições no caso do modelo CFC, necessita-se apenas que a heurística remova os caminhos que não possuem recursos necessários para encaminhar as requisições sem violar as restrições. O modelo CFC permite tratar de forma implícita todas as restrições baseadas em caminhos, tanto aditivas como não aditivas. Deste modo, tendo apenas uma restrição aditiva de arco, como largura de banda, por exemplo, pode-se resolver o modelo CFC estendido (equações (5.1a) – (5.11)) utilizando o modelo básico (equações (3.1a) – (3.1h)), bastando para isso remover os caminhos que não satisfazem a restrição.

Diferentemente do modelo CFC, o modelo RM foi formulado como um modelo baseado em fluxo em redes tornando complexo o tratamento de restrições adicionais. A heurística RM apresentada na seção 2.3.3 necessitaria resolver o problema de caminhos mínimos sob restrições ao invés de resolver um caminho mínimo sem restrições (utilizando o algoritmo de Dijkstra) a cada encaminhamento.

Segundo [33], caminhos mínimos sob uma única restrição são conhecidos por serem classifica-

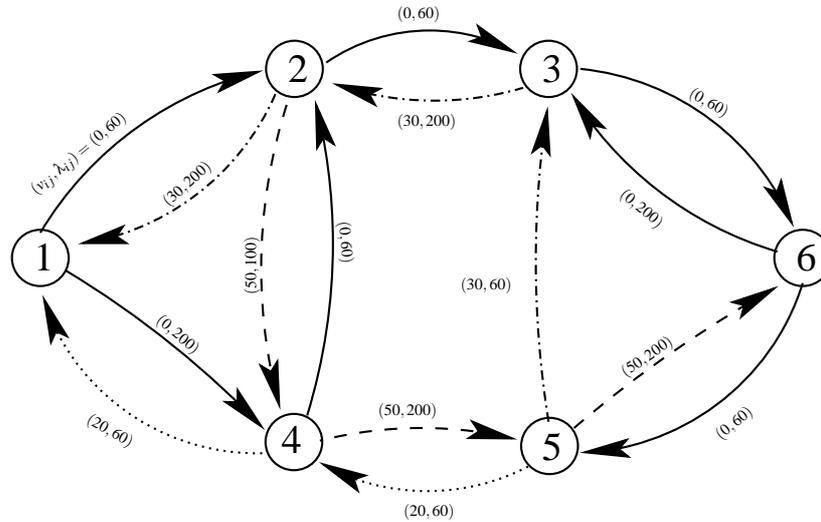


Figura 5.2: Ocupação dos arcos com restrição do número de saltos usando o modelo CFC.

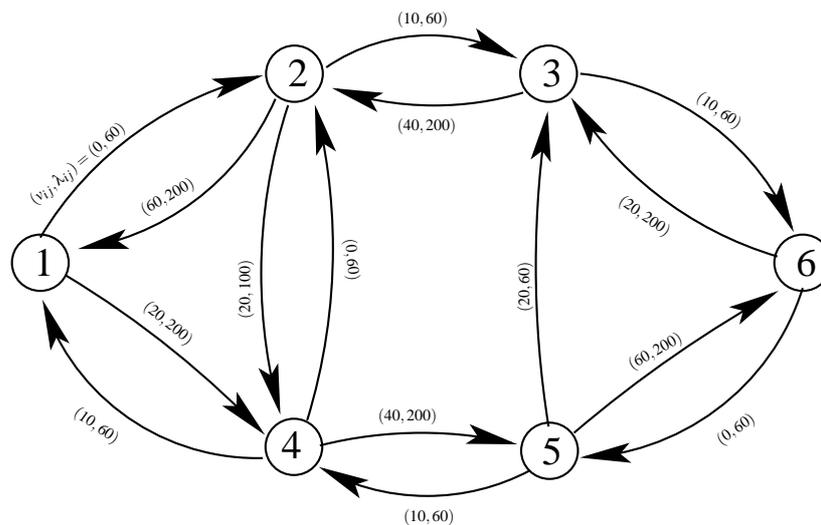


Figura 5.3: Ocupação dos arcos usando o modelo RM

dos como NP-Difíceis. No caso do roteamento com QoS, precisa-se considerar múltiplas restrições. O problema de roteamento com QoS que considera múltiplas restrições é classificado como NP-Difícil, não sendo prático resolver tais problemas de forma exata [24]. Logo, a heurística CFC se apresenta como uma alternativa para a solução do problema de roteamento com QoS sob múltiplas restrições.

5.4 Conclusões

O objetivo deste capítulo foi apresentar conceitos sobre restrições em redes e uma revisão de alguns trabalhos relacionados. Conceituaram-se as restrições aditiva e não aditivas, baseadas em arcos

e em caminhos.

Apresentaram-se extensões para os modelos RM e CFC de forma que aceitem restrições aditivas e não aditivas de caminhos e arcos. Em seguida, ilustrou-se o uso de uma restrição aditiva que demonstrou a capacidade da heurística do modelo CFC de tratar restrições adicionais sem a necessidade de implementar uma nova heurística.

Capítulo 6

Conclusão

Este trabalho propôs uma abordagem baseada em caminhos para otimização do uso global de recursos numa rede MPLS intra-domínio, considerando situações de demandas de fluxos com perfis de banda predefinidos variantes no tempo. Mostra-se ainda que esta abordagem pode ser facilmente estendida para tratar múltiplas restrições.

Um modelo baseado em caminhos formulado em PLI foi apresentado com objetivo de obter soluções de mesma qualidade que [30], além de permitir a adição de restrições adicionais. Essa formulação em PLI foi utilizada para definir limites inferiores a serem empregados na avaliação da qualidade dos resultados. Uma heurística para roteamento off-line e on-line baseado no modelo PLI para roteamento fixo com perfis de banda variantes no tempo foi apresentada para alocação de LSPs em uma rede MPLS.

Inicialmente, os experimentos numéricos consideraram apenas a alocação de banda. Posteriormente apresentaram-se outras métricas que caracterizam a qualidade de serviço dos fluxos na formulação, utilizando o mesmo critério como objetivo.

Constatou-se que, a solução obtida para o roteamento fixo é muito próxima à obtida com roteamento variável e similar ao apresentado por Ricciato e Monaco [30]. Devido a necessidade de recursos adicionais para implantação de roteamento variável, o resultado obtido permite afirmar que a solução para roteamento fixo é satisfatória e atende de forma muito similar as necessidades de roteamento com baixo custo e economia de recursos sem necessidade de modificar a estrutura da rede. Comportamento similar ao obtido pela proposta RM [30] quanto à granularidade do período de tempo foi alcançado utilizando o modelo CFC.

Uma comparação do modelo proposto baseado em caminhos em relação ao modelo existente

baseado em fluxo em redes foi apresentado e discutido. Os resultados obtidos para o modelo baseado em caminhos ficam muito próximos dos valores obtidos pelo modelo de fluxo em redes, mesmo utilizando poucos caminhos (disjuntos, por exemplo) tanto para o roteamento on-line como off-line, mas a um custo bem menor. Pode-se afirmar que é possível obter resultados de qualidade similar em ambos os modelos. A possibilidade de controlar o número de caminhos permite atingir um resultado viável em um tempo computacional baixo.

O solução inteira do modelo CFC utilizando todos os caminhos do grafo apresenta o mesmo resultado que a obtido pelo modelo RM, permitindo concluir que o modelo está modelado corretamente. A relaxação do modelo CFC comparado com o do modelo RM no caso de roteamento obteve resultados próximos, com uma diferença menor que 1% para o cenário 1 e menor que 50% para o cenário 2, o que demonstra que o modelo CFC pode substituir plenamente o modelo RM para o encaminhamento de fluxos. A heurística do modelo CFC para o roteamento off-line e on-line apresenta melhores resultados em relação a heurística do modelo RM em particular com poucas requisições. Com o aumento do número de requisições os resultados passam a ficar muito próximos. Deste modo, é possível concluir que as heurísticas (off-line e on-line) propostas possibilitam um encaminhamento de fluxos de mesma qualidade ou melhor em certas situações que a heurística do modelo RM.

Finalmente, com a heurística CFC é possível tratar explicitamente restrições adicionais sem alterá-la, bastando para isso remover os caminhos que não atendem aos requisitos exigidos pelas requisições. No caso dos modelos baseados em fluxo (como RM), seria necessário propor uma nova heurística.

Para tratar do caso de múltiplas restrições – banda e métricas de QoS – foi proposta uma formulação estendida do modelo CFC, a qual possibilita a utilização de restrições adicionais as utilizadas no modelo anterior e do modelo em fluxo em redes. A heurística tem a propriedade de tratar múltiplas restrições aditivas, ao contrário daquela proposta por [30], demonstrando também bons resultados computacionais frente aos limites inferiores obtidos com relaxação linear contínua.

6.1 Contribuições

Em geral, os trabalhos encontrados na literatura apresentados anteriormente não tratam múltiplas restrições de QoS. O trabalho de Ricciato e Monaco [30], estudo base utilizado para a realização deste trabalho não considera o uso de restrições adicionais.

Este trabalho apresentou como contribuição uma proposta para resolver o problema de roteamento em redes baseada em caminhos modelada em programação linear inteira e uma heurística.

Finalmente, são fornecidas uma formulação estendida para os modelos CFC e RM, que permite o tratamento de restrições adicionais. Para este caso, verificou-se que a heurística CFC permite tratar de forma implícita as restrições adicionais de forma a respeitar o modelo CFC estendido, o que não é possível com a heurística RM. Este trabalho também gerou uma publicação [7].

6.2 Trabalhos Futuros

Com o intuito de aperfeiçoar este trabalho, sugerem-se como trabalhos futuros:

- Aperfeiçoar a implementação dos algoritmos: como os algoritmos foram implementados visando sua corretude, não se preocupou com a eficiência dos mesmos. Para tanto, é necessário reimplementá-los.
- Extensão dos experimentos: Apresentou-se somente um exemplo ilustrativo do uso de uma restrições adicional, apesar da possibilidade de utilizar múltiplas restrições. Experimentos considerando múltiplas restrições podem ser realizados.
- Aplicação em situações reais: Pode-se aplicar esta abordagem para situações reais através da construção e implementação de uma ferramenta de gerência de redes para apoio de decisão.
- Possibilidade do critério c levar em conta métricas adicionais à largura de banda (por exemplo, tempo).

Nomenclatura

BGP	Border Gateway Protocol
DSCP	Differentiated Service Code Point
ET	Engenharia de Tráfego
FEC	Forwarding Equivalence Class
GB	Gigabyte
GHz	Gigahertz
GNU	GNU's Not Unix
LSP	Label Switched Path
LIB	Label Information Base
MPLS	Multiprotocol Label Switching
OSI	Open System Interconnection
OSPF	Open Shortest Path First
PLI	Programação Linear Inteira
QoS	Qualidade de Serviço
RAM	Random Access Memory
RSVP	Resource ReSerVation Protocol
SLA	Service Level Agreement
ToS	Type of Service

Referências Bibliográficas

- [1] Ahuja, R. K., Magnanti, T. L., e Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall.
- [2] Awduche, D. O. (1999). MPLS and traffic engineering in IP networks. *IEEE Communications Magazine*, Vol. 37, No. 12, pp. 42–47.
- [3] Banerjee, G. e Sidhu, D. (2002). Comparative analysis of path computation techniques for MPLS traffic engineering. *Computer Networks: The Int. Journal of Computer and Telecommunications Networking*, Vol. 40, No. 17, pp. 149–165.
- [4] Bauer, D., Daigle, J. N., Iliadis, I., e Scotton, P. (2006). Topology aggregation for combined additive and restrictive metrics. *Computer Networks*, Vol. 50pp. 3284–3299.
- [5] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., e Weiss, W. (1998). An architecture for differentiated services. rfc 2475.
- [6] Braden, R., Clark, D., e Shenker, S. (1994). Integrated services in the internet architecture: an overview. rfc 1633.
- [7] Carlson, A. C., Camponogara, E., e Farines, J.-M. (2007). Roteamento com perfis variantes no tempo: Uma abordagem baseada em caminhos. Proc. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos 2007*, Belém.
- [8] Cormen, T. H., Leiserson, C. E., Rivest, R. I., e Stein, C. (2001). *Introduction to Algorithms*. MIT Press, 2 edição.
- [9] Dias, R. A. (2004). *Engenharia de Tráfego em Redes IP Sobre Tecnologia MPLS: Otimização Baseada em Heurísticas*. Tese (Doutorado em Engenharia Elétrica), Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis.

- [10] Dias, R. A., Farines, J.-M., Camponogara, E., Willrich, R., e Campestrini, A. O. (2004). Engenharia de tráfego dinâmica em redes IP sobre tecnologia MPLS: Otimização baseada em heurísticas. *Proc. Simpósio Brasileiro de Redes de Computadores 2004*, Gramado.
- [11] Fortz, B. e Thorup, M. (2002). Optimizing OSPF/IS-IS weights in a changing world. *IEEE Journal on Selected Areas in Communications*, Vol. 20, No. 4, pp. 756–767.
- [12] Fourer, R., Gay, D. M., e Kernighan, B. W. (2002). *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, 2 edição.
- [13] Guerin, R., Williams, D., e Orda, A. (1997). QoS routing mechanisms and OSPF extensions. *Proc. Globecom'97*, pp. 1903–1908, Arizona, EUA.
- [14] Halabi, S. (2000). *Internet Routing Architectures*. Cisco Press, 2 edição.
- [15] ILOG (2006). CPLEX. <http://www.ilog.com>.
- [16] Kar, K., Kodialam, M., e Lakshman, T. (2000). Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications. *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 12, pp. 2566–2579.
- [17] Korkmaz, T., Krunz, M., e Tragoudas, S. (2002). An efficient algorithm for finding a path subject to two additive constraints. *Computer Communications*, Vol. 25, No. 3, pp. 225–238.
- [18] Kuipers, F., Mieghem, P. V., Korkmaz, T., e Krunz, M. (2002). An overview of constraint-based path selection algorithms for QoS routing. *IEEE Computer Magazine*, Vol. 40, No. 12, pp. 50–55.
- [19] Kurose, J. F. e Ross, K. W. (2001). *Computer networking : a top-down approach featuring the internet*. Addison-Wesley.
- [20] Lin, X. e Shroff, N. B. (2006). An optimization-base approach for QoS routing in high-bandwidth networks. *IEEE/ACM Transactions on Networking*, Vol. 14, No. 6, .
- [21] Ma, Q. e Steenkiste, P. (1997). On path selection for traffic with bandwidth guarantees. *Proc. Fifth International Conference on Network Protocols (ICNP'97)*, Atlanta, EUA.
- [22] Magalhães, M. F. e Cardozo, E. (2001). *Livro Texto dos Minicursos - SBRC 2001*, chapter 3, pp. 89–136. SBC e LARC, Florianópolis.
- [23] Masip-Bruin, X., Yannuzzi, M., Domingo-Pascual, J., Fonte, A., Curado, M., Monteiro, E., Kuipers, F., Mieghem, P. V., Avallone, S., Ventre, G., Aranda-Gutiérrez, P., Hollick, M., Steinmetz, R., Iannone, L., e Salamatian, K. (2006). Research challenges in QoS routing. *Computer Communications*, Vol. 29, No. 5, pp. 563–581.

- [24] Mieghem, P. V. e Kuipers, F. A. (2004). Concepts of exact QoS routing algorithms. *IEEE/ACM Transactions on Networking*, Vol. 12, No. 5, pp. 851–864.
- [25] Migliore, M., Martorana, V., e Sciortino, F. (1990). An algorithm to find all paths between two nodes in a graph. *Journal of Computational Physics*, Vol. 87, No. 1, pp. 231–236.
- [26] Neve, H. D. e Mieghem, P. V. (2000). TAMCRA: a tunable accuracy multiple constraints routing algorithm. *Computer Communications*, Vol. 23, No. 7, pp. 667–679.
- [27] Orda, A. e Sprintson, A. (2003). Precomputation schemes for QoS routing. *IEEE/ACM Transactions on Networking*, Vol. 11, No. 4, pp. 578–591.
- [28] Paul, P. e Raghavan, S. V. (2002). Survey of QoS routing. *Proc. Proceedings of the 15th International Conference on Computer Communications*, Vol. 1, pp. 50–75. International Council for Computer Communication.
- [29] Plotkin, S. A. (1995). Competitive routing of virtual circuits in ATM networks. *IEEE Journal of Selected Areas in Communications*, Vol. 13, No. 6, pp. 1128–1136.
- [30] Ricciato, F. e Monaco, U. (2005). Routing demands with time-varying bandwidth profiles on a MPLS network. *Computer Networks: The Int. Journal of Computer and Telecommunications Networking*, Vol. 47, No. 1, pp. 47–61.
- [31] Ricciato, F., Salsano, S., Belmonte, A., e Listanti, M. (2002). Off-line configuration of a MPLS over WDM network under time-varying offered traffic. *Proc. INFOCOM 2002*, pp. 57–65.
- [32] Rossum, G. V. (2006). Python patterns – implementing graphs. <http://www.python.org/>.
- [33] Shima, R. B. (2006). Caminhos Mínimos sob Restrições: Uma Revisão e Aplicações. Dissertação (Mestrado em Engenharia Elétrica), Centro Tecnológico, Universidade Federal de Santa Catarina, Florianópolis.
- [34] Suri, S., Waldvogel, M., Bauer, D., e Warkhede, P. R. (2003). Profile-based routing and traffic engineering. *Computer Communications*, Vol. 26, No. 4, pp. 351–365.
- [35] Swallow, G. (1999). MPLS advantages for traffic engineering. *IEEE Communications Magazine*, Vol. 37, No. 12, pp. 54–57.
- [36] Wang, Z. e Crowcroft, J. (1996). Quality-of-service routing for supporting multimedia applications. *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 7, pp. 1228–1234.

- [37] Wang, Z. e Crowcroft, J. (1999). Bandwidth-delay based routing algorithms. Proc. *Proceedings of IEEE GLOBECOM*, Vol. 3, pp. 2129–2133, Rio de Janeiro, Brasil.
- [38] Yang, S.-J. e Chou, H.-C. (2003). Adaptative QoS parameters approach to modeling Internet performance. *International Journal of Network Management*, Vol. 13, No. 1, pp. 69–82.