

OTIMIZAÇÃO MULTI-OBJETIVO APLICADA À
ALOCÇÃO DINÂMICA DE ROTAS EM REDES
DE TELECOMUNICAÇÕES

FERNANDO AFONSO SANTOS
ORIENTADOR: GERALDO ROBSON MATEUS

OTIMIZAÇÃO MULTI-OBJETIVO APLICADA À
ALOCAÇÃO DINÂMICA DE ROTAS EM REDES
DE TELECOMUNICAÇÕES

Dissertação apresentada ao Programa de
Pós-Graduação em Ciência da Computação
da Universidade Federal de Minas Gerais
como requisito parcial para a obtenção do
grau de Mestre em Ciência da Computação.

Belo Horizonte
Março de 2009

© 2009, Fernando Afonso Santos.
Todos os direitos reservados.

Santos, Fernando Afonso
S237o Otimização multi-objetivo aplicada à alocação
dinâmica de rotas em redes de telecomunicações /
Fernando Afonso Santos. — Belo Horizonte, 2009
xviii, 64 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de
Minas Gerais
Orientador: Geraldo Robson Mateus

1. Otimização - pesquisa operacional.
2. Telecomunicações. 3. Roteamento. I. Título.

CDU 519.6*65



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Otimização Multi-Objetivo Aplicada à Alocação Dinâmica de Rotas em Redes
de Telecomunicações

FERNANDO AFONSO SANTOS

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. GERALDO ROBSON MATEUS - Orientador
Departamento de Ciência da Computação - UFMG

PROF. RICARDO HIROSHI CALDEIRA TAKAHASHI
Departamento de Matemática - UFMG

PROF. RODNEY REZENDE SALDANHA
Departamento de Engenharia Elétrica - UFMG

PROF. JOSÉ MARCOS SILVA NOGUEIRA
Departamento de Ciência da Computação - UFMG

Belo Horizonte, 27 de março de 2009.

*Aos meus pais Afonso e Rosângela, pelo
amor incondicional ao longo desta caminhada.*

Agradecimentos

Agradeço primeiramente a Deus por me iluminar e abençoar, mesmo eu estando distante em alguns momentos.

À minha família, meus amados pais Afonso e Rosângela e também meus irmãos Grazielle e Leonardo, que mesmo não estando fisicamente presentes não deixam de me apoiar e torcer por mim. Àqueles parentes que ajudaram em minha educação e com sentimentos muito verdadeiros me fazem sentir bem a cada vez que os vejo. Obrigado inclusive por me incluir em suas orações diárias, pedindo por saúde e proteção.

Agradeço também à Rita, que me ajudou a superar cada dificuldade encontrada ao longo desta caminhada com muito amor, carinho e dedicação. Em muitas oportunidades mesmo precisando de atenção e carinho, deixava de lado seus problemas para me confortar e aliviar o rotineiro estresse. Querida, parte desta conquista também é sua, muito obrigado!

Ao meu orientador Geraldo Robson Mateus, que sempre me atendeu prontamente e forneceu todo o suporte necessário para a realização deste trabalho. Isto sem contar as orientações que foram muito além de acadêmicas, me mostrando as implicações de minhas decisões profissionais e até mesmo ajudando a tomá-las. Obrigado por tudo, Robson.

Meus amigos também foram fundamentais para a conclusão desta etapa na minha vida e merecem aqui o meu agradecimento. Bobô obrigado pela ajuda desde minha chegada ao DCC, não apenas pela hospitalidade mas também pelos momentos diversão e companheirismo! Marcelo, agradeço pelo total apoio e suporte fornecido ainda na graduação, me incentivando a trilhar por este caminho. E a todos os outros que estiveram comigo ao longo deste mestrado, valeu pessoal!

Resumo

O desenvolvimento de novas tecnologias aplicadas às redes de computadores e a expansão da Internet trouxeram como consequência um grande aumento no tráfego de dados nas redes. As exigências de qualidade de serviço (QoS) por parte das aplicações aumentaram proporcionalmente e estão mais rigorosas. Tais exigências envolvem diferentes métricas de QoS, entre elas pode-se citar o atraso, o balanceamento de carga e o número de pacotes perdidos na comunicação.

Diferentes algoritmos de otimização são utilizados para alcançar os valores de QoS desejados pelas aplicações. Na otimização mono-objetivo considera-se uma métrica de QoS como função objetivo do problema e os valores para as demais métricas podem ser inseridos como restrições, enquanto na multi-objetivo é possível considerar cada métrica de QoS a ser otimizada como uma função objetivo do problema, otimizando-as simultaneamente. Por otimizar múltiplas funções objetivo simultaneamente, o resultado da otimização multi-objetivo consiste em um conjunto de soluções denominado conjunto Pareto-ótimo.

Aplicar a otimização multi-objetivo ao roteamento dinâmico das requisições é um grande desafio, pois além de calcular o resultado do roteamento é necessário um modelo de tomada de decisão para escolher entre as soluções disponíveis aquela que mais se adequa às exigências de QoS dinamicamente. Não existem na literatura trabalhos que contemplem esta estratégia, portanto, esta é a contribuição proposta neste trabalho.

Para verificar o comportamento desta estratégia em um contexto de redes de telecomunicações será utilizada a arquitetura MPLS (*Multi-Protocol Label Switching*). Esta arquitetura estende as funcionalidades do protocolo IP e apresenta como principal benefício a possibilidade de realizar o roteamento explícito das requisições na rede, definindo qual caminho deve ser seguido entre a origem e o destino de cada requisição.

Os resultados mostraram uma melhora no desempenho da rede ao avaliar diferentes métricas de QoS simultaneamente. Outra vantagem apresentada é a flexibilidade da otimização multi-objetivo proporcionando que soluções com diferentes características sejam escolhidas alterando-se apenas os parâmetros do procedimento de tomada de decisão.

Abstract

The new networking technologies and the Internet diffusion produced the data traffic increase in the networks. The quality of service (QoS) requirements for applications also increased and become stricter. Different QoS metrics can be evaluated by applications, for example the delay, the load balance and the packets lost in the communication.

Different optimization algorithms can be used to achieve the QoS metric values for applications. The single objective approach regards as objective function one QoS metric and the others are considered as problem constraints, while the multi-objective optimization considers each QoS metric as an objective function and optimize them together. Because different objective function are optimized at once, the multi-objective result consist in set of solutions, named Pareto-optimal set.

To apply multi-objective optimization for dynamic routing problems is a great challenge, because this strategy demands a decision support system to choose an appropriated solution from Pareto-optimal set dynamically, matching the QoS requirements. There is no works investigating this strategy in the literature, and this is the contribution proposed in this paper.

To verify the performance of this strategy in telecommunications networks context, the MPLS architecture (*Multi-Protocol Label Switching*) is used . This architecture extends the functionalities of the IP protocol and presents as main advantage the possibility to perform explicit routing of requests in the network, defining explicitly the path to data traffic flow for each communication request.

The results show improvements in overall performance of the network evaluating different QoS metrics simultaneously. Another advantage is the flexibility provide by the multi-objective optimization allowing to choose different solutions for the problem by changing the parameters in the decision support system.

Sumário

1	Introdução	1
1.1	Motivação	3
1.2	Definição do problema	4
1.3	Objetivos	5
1.4	Contribuições	5
1.5	Estrutura da dissertação	6
2	Conceitos básicos	7
2.1	Algoritmos de roteamento	7
2.2	A arquitetura MPLS	11
2.3	Trabalhos relacionados	16
2.3.1	Otimização mono-objetivo	16
2.3.2	Otimização multi-objetivo	17
3	Formulação matemática	19
3.1	Definições	19
3.2	Formulação matemática mono-objetivo	20
3.3	Formulação matemática multi-objetivo	22
3.3.1	Conceitos iniciais	22
3.3.2	Modelo matemático	27
3.3.3	Alternativas de solução	28
4	Heurística - Algoritmo Genético multi-objetivo	31
4.1	NSGA-II	32
4.2	Representação genética	34
4.3	População inicial	35
4.4	Cruzamento	36
4.5	Mutação	38
4.6	Complexidade assintótica	39
4.7	Qualidade das aproximações e eficiência	41

5	Resultados de simulações	47
5.1	Cenário de simulação - 1	49
5.2	Cenário de simulação - 2	51
5.3	Cenário de simulação - 3	52
5.4	Cenário de simulação - 4	54
5.5	Cenário de simulação - 5	55
5.6	Conclusão	57
6	Considerações finais	59
	Referências Bibliográficas	61

Lista de Figuras

1.1	Exemplo de domínio de uma rede MPLS	2
2.1	Estrutura da pilha de protocolos do modelo ISO/OSI	8
2.2	Etapas do algoritmo de roteamento inundação	9
2.3	Evolução das arquiteturas para provimento de QoS em redes TCP/IP	12
2.4	Manipulação do rótulo pelos roteadores da rede MPLS	14
3.1	Espaços de soluções para problemas de otimização mono-objetivo e multi-objetivo	23
3.2	Conjuntos de soluções no espaço dos objetivos	25
3.3	Exemplo de conflitos entre dois objetivos na compra de um notebook	25
3.4	Solução utópica de um problema de otimização multi-objetivo	26
4.1	Definição dos indivíduos na próxima geração pelo NSGA-II	33
4.2	Representação genética do algoritmo	35
4.3	Processo de cruzamento no algoritmo genético multi-objetivo	37
4.4	Procedimento alternativo para o cruzamento no algoritmo caso não sejam encontrados vértices em comum	37
4.5	Exemplo de mutação no algoritmo	38
4.6	Topologias de rede utilizadas nos experimentos	41
4.7	Influência do parâmetro número de gerações no algoritmo genético	44
4.8	Influência do parâmetro tamanho da população no algoritmo genético	45
5.1	Comparação da abordagem de otimização multi-objetivo com as estratégias de caminho mínimo e balanceamento de carga, considerando a topologia NSF	49
5.2	Diferentes abordagens de otimização no roteamento de requisições de broadcast, considerando a topologia Carrier	51
5.3	Flexibilidade da otimização multi-objetivo no problema de roteamento em redes MPLS	53
5.4	Variação dos valores de atraso e balanceamento de carga nas redes, ao se utilizar diferentes matrizes de preferência	54

5.5	Comparação dos resultados obtidos pelas estratégias de roteamento estática e dinâmica	55
5.6	Comparação dos valores de rejeição para as diferentes classes de serviço . .	56
5.7	Avaliação da abordagem de diferenciação de classes, quanto ao parâmetro de QoS atraso das requisições	57

Lista de Tabelas

4.1	Características das topologias utilizadas nos experimentos	42
4.2	Solução utópica para problemas de roteamento multi-objetivo	43
4.3	Aproximação das soluções utópicas através do algoritmo genético	43
5.1	Indicadores de QoS nas redes utilizando diferentes estratégias de roteamento	50
5.2	Indicadores de QoS nas redes utilizando diferentes estratégias de roteamento para atender requisições de broadcast	52

Capítulo 1

Introdução

A ampla utilização da Internet e o desenvolvimento de tecnologias aplicadas às redes de computadores trouxeram como consequência o aumento no número de aplicações que operam através de redes. Em particular, aplicações de tempo real como a transmissão de som e vídeo, têm aumentado expressivamente o tráfego de dados nas redes.

Para a realização da comunicação de algumas destas aplicações, um importante conceito é o de SLA (*Service Level Agreement*) ou acordo de nível de serviço. O SLA trata de um acordo, explícito ou não, que as redes devem oferecer para que as aplicações sejam atendidas adequadamente. Como exemplos de métricas para definir SLA é possível citar: o atraso da comunicação, a perda de pacotes, a disponibilidade do serviço, entre outros.

A definição de SLAs, exige que a rede satisfaça requisitos mínimos para seu funcionamento. Neste contexto, surge o conceito de QoS (*Quality of Service*) ou (Qualidade de Serviço) que é amplamente utilizado na comunicação em redes. O QoS pode ser utilizado para medir a qualidade do serviço disponibilizado nas redes sob diferentes perspectivas. Em nível de aplicações a qualidade do serviço pode estar relacionada a alguns indicadores de QoS, enquanto ao considerar os aspectos físicos da rede os indicadores podem ser outros. Independente de quais indicadores de QoS sejam definidos, a qualidade da comunicação na rede está em função da sua satisfação.

No entanto, o roteamento no protocolo IP baseia-se no serviço de melhor-esforço, e o encaminhamento dos pacotes na rede é realizado objetivando a máxima eficiência no fluxo dos dados, inviabilizando a satisfação de alguns indicadores de QoS. Neste contexto, foi proposta a arquitetura MPLS (*MultiProtocol Label Switching*) Rosen et al. [2001].

A arquitetura MPLS introduz nas redes IP uma capacidade avançada de controle do roteamento, como a característica de roteamento explícito, e auxilia na elaboração de redes com suporte a QoS [Awduche, 1999]. São adicionados rótulos aos pacotes que

trafegam no domínio MPLS, permitindo o controle das rotas, que são denominadas LSP (*Label Switching Path*). A figura 1, retirada de Tanezine e dos Santos [2006], ilustra o domínio de uma rede MPLS.

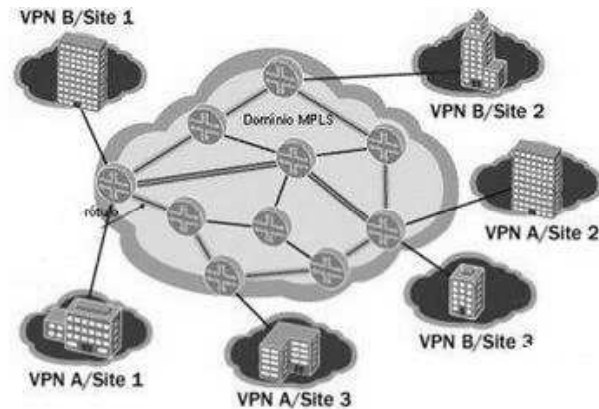


Figura 1.1. Exemplo de domínio de uma rede MPLS

Uma das vantagens proporcionadas pela definição de rotas LSP explícitas na arquitetura MPLS é a possibilidade de considerar vários indicadores de QoS no roteamento dos pacotes, o que não é feito por protocolos de roteamento comuns na Internet, como o OSPF (*Open Shortest Path First*). No entanto, a otimização das rotas pode ser uma tarefa complexa. O problema de se determinar rotas LSP sujeitas a mais de duas restrições de QoS é provado ser NP-Completo [Shao et al., 2006].

Técnicas de otimização têm sido propostas a fim de realizar o roteamento dos LSPs nas redes MPLS com a máxima eficiência, atendendo aos indicadores de QoS da rede. Devido à multiplicidade de indicadores de QoS a considerar na otimização, as abordagens de otimização se dividem em duas: mono-objetivo e multi-objetivo.

Na otimização mono-objetivo, a modelagem do problema é feita utilizando-se apenas uma função objetivo, as restrições do problema podem definir valores máximos e mínimos para os outros indicadores de QoS a serem otimizados. Já a otimização multi-objetivo tem a possibilidade de considerar cada indicador como uma função objetivo do problema e otimizá-las simultaneamente. Entretanto, a complexidade envolvida para solucionar os problemas de otimização multi-objetivo é maior. De acordo com Ulungu e Teghem [1994], diversos problemas de otimização combinatória solúveis em tempo polinomial, se tornam NP-Completo quando formulados como problemas de otimização multi-objetivo.

Uma outra característica encontrada na otimização multi-objetivo é que seu resultado pode ser caracterizado por um conjunto de soluções, denominado conjunto Pareto-ótimo, ou fronteira de Pareto. As soluções deste conjunto apresentam diferentes valores para as funções objetivo do problema e a escolha da solução mais adequada,

envolve técnicas de tomada de decisão, agregando complexidade à esta abordagem de otimização.

Ainda sobre a utilização de técnicas de otimização aplicadas às redes MPLS, estas podem considerar o problema de roteamento dos LSPs de maneira estática ou dinâmica. Na otimização estática, o estado da rede é conhecido bem como todas as requisições a serem alocadas na rede. O processo de otimização consiste em determinar rotas na rede que satisfaçam os indicadores de QoS propostos. Já na otimização dinâmica, não existe o conhecimento prévio de todas as requisições a serem alocadas. Estas surgem ao longo do tempo, e o processo de otimização deve alocá-las considerando o estado atual da rede, que muda conforme o atendimento das requisições. A dinamicidade desta abordagem faz com que a complexidade envolvida neste tipo de otimização seja maior que na otimização estática.

Além das diferentes estratégias de otimização possíveis de se aplicar ao problema, é possível também considerar diferentes indicadores de QoS a otimizar. Assim, fica evidente a importância das técnicas de otimização aplicadas ao roteamento de LSPs em redes MPLS, consistindo em uma importante área de pesquisa cujas contribuições podem ser relevantes para a realização da comunicação em redes de computadores.

1.1 Motivação

Por se tratar de um tema de grande relevância, diferentes propostas de solução para o problema de roteamento das rotas LSP em redes MPLS são encontradas. Entre elas, Oliveira [2005] em sua tese de doutorado utilizou um procedimento de otimização mono-objetivo considerando três indicadores de QoS: *(i)* o número de links utilizados pelas rotas LSP (relacionado ao seu atraso), *(ii)* a quantidade de rejeições das requisições e *(iii)* o balanceamento de carga da rede. A alocação dos LSPs é feita estaticamente.

O problema de roteamento em redes MPLS tem natureza multi-objetivo. Existem diferentes indicadores de QoS dos quais deseja-se obter valores ótimos, no entanto, existem conflitos ao otimizá-los simultaneamente. Como exemplo, suponha que os indicadores *(i)* e *(iii)* supracitados sejam duas funções objetivo distintas a serem minimizadas. Ao atender as requisições utilizando o menor número de links, é possível que alguns links fiquem sobrecarregados, piorando o balanceamento de carga da rede, ou então ao realizar o roteamento privilegiando o balanceamento de carga na rede é possível que se utilize uma quantidade de links maior, comprometendo o valor do objetivo *(i)* enquanto prioriza-se o objetivo *(iii)*.

Portanto, a investigação da abordagem de otimização multi-objetivo agrega complexidade à solução do problema, porém pode trazer uma importante contribuição ao problema de roteamento. Além do mais, não foram encontrados na literatura trabalhos que avaliem a otimização multi-objetivo das rotas LSP em um contexto dinâmico de otimização. Isto implica na necessidade de um procedimento dinâmico para tomada de decisão, em função do comportamento da rede e das requisições, o que aumentou a motivação para a realização deste trabalho.

1.2 Definição do problema

Este trabalho trata do problema da alocação dinâmica de rotas LSP em redes MPLS através de uma abordagem de otimização multi-objetivo. As requisições de comunicação na rede consistem de uma origem e um destino para o tráfego, além do seu valor de exigência de banda, necessário para o tráfego dos dados na rede. Cada requisição será alocada através de uma rota LSP na rede, que será responsável por definir um caminho entre a origem e o destino, havendo a reserva de banda nos links utilizados. A banda reservada para a alocação de uma rota LSP na rede não poderá ser utilizada por nenhuma outra, apenas quando a requisição for desalocada.

Por se tratar de um contexto dinâmico de otimização, a quantidade de requisições a serem alocadas na rede não é previamente conhecida, bem como os valores de origem, destino e exigência de banda de cada uma. As requisições surgem aleatoriamente agrupadas ou não, e assim que alocadas na rede elas utilizam os seus recursos enquanto estiverem ativas.

Para solucionar o problema de roteamento nas redes MPLS, é proposta uma formulação matemática multi-objetivo que fornece soluções do conjunto Pareto-ótimo para o problema, no entanto, demanda um elevado tempo computacional e só pode ser aplicada para solucionar o problema de roteamento estático. Para solucionar o problema de roteamento dinâmico é apresentada uma heurística para aproximar os resultados do conjunto Pareto-ótimo eficientemente. Esta heurística é um algoritmo genético multi-objetivo e poderá ser aplicado ao problema dinâmico, devido à eficiência de suas respostas.

Este problema multi-objetivo será modelado considerando três funções objetivo, que são:

- a quantidade de links utilizados para atender cada uma das requisições;
- a quantidade de rejeição das requisições na rede;
- o balanceamento de carga na rede;

Este problema será modelado e solucionado matematicamente através de programação linear inteira mista (PLIM). Mas devido à elevada complexidade computacional para solucioná-lo, é proposta uma heurística para obter aproximações do resultado ótimo eficientemente. As soluções obtidas serão avaliadas em um ambiente de simulação, onde são experimentados diferentes contextos de comunicação para verificar o seu comportamento.

1.3 Objetivos

Com a realização deste trabalho espera-se que as soluções obtidas para o problema de roteamento em redes MPLS estejam mais próximas das situações reais de roteamento encontradas nas redes. Isto porque, ao considerar o problema em sua versão multi-objetivo e através de uma abordagem dinâmica de otimização, sua representação se aproxima da realidade e, portanto, pode contribuir de maneira mais efetiva para o roteamento dos LSPs na rede.

A utilização de um procedimento de otimização multi-objetivo pode oferecer também mais flexibilidade na realização do roteamento. Como são avaliados diferentes objetivos simultaneamente, é possível privilegiar um deles em função do comportamento da rede em um determinado momento, ou seja, a otimização pode se adequar melhor ao estado atual da rede, fornecendo soluções que melhor satisfaçam os requisitos de QoS da rede em um dado instante.

É comum encontrar aplicações nas redes com necessidades diferenciadas umas das outras, por exemplo, existem aquelas aplicações que devem ser atendidas imediatamente após surgirem ou outras que necessitam de menores valores de atraso na comunicação para que possam funcionar corretamente. Diante deste cenário, este trabalho propõe também um estudo que classifica as requisições em diferentes classes de serviço. Assim, espera-se que o procedimento de otimização das rotas possa solucionar de maneira mais efetiva o problema de roteamento nas redes MPLS, aproximando-se ainda mais dos contextos reais de comunicação.

1.4 Contribuições

As contribuições obtidas com a realização deste trabalho estão relacionadas ao roteamento de requisições em redes MPLS, onde procura-se uma abordagem inovadora que consiga agregar qualidade e flexibilidade na definição das rotas LSP na rede à medida em que estas requisições surgem. Mais especificamente, as contribuições são:

1. Modelagem matemática do problema de roteamento através de uma abordagem multi-objetivo de otimização, utilizando programação linear inteira mista (PLIM)
2. Codificação e solução da modelagem PLIM apresentada utilizando o solver CPLEX [Ilog Incorporation, 2006].
3. Implementação de uma solução heurística para obter aproximações dos resultados eficientemente, de modo que possam ser aplicados em um contexto dinâmico de otimização
4. Avaliação da proposta através de simulações que abrangem diferentes contextos de comunicação nas redes, inclusive agrupando as requisições em classes para melhor aproximar as simulações dos cenários reais de roteamento nas redes MPLS

1.5 Estrutura da dissertação

O restante deste documento apresenta a seguinte estrutura: O Capítulo 2 servirá como referência aos conceitos básicos envolvidos nesta dissertação, onde serão apresentados os principais conceitos sobre roteamento e redes MPLS. No Capítulo 3 é apresentada a formulação matemática do problema. Alguns aspectos relacionados à modelagem mono-objetivo são apresentados bem como conceitos fundamentais da otimização multi-objetivo, que servirão de base para a formulação proposta. Como este trabalho estuda o roteamento em um contexto dinâmico, foi implementada uma solução heurística para realizar aproximações dos resultados eficientemente. Esta heurística se trata de um algoritmo genético multi-objetivo e é apresentada no Capítulo 4. Para verificar o desempenho da abordagem proposta, no Capítulo 5 são apresentados os resultados de simulações utilizando diferentes cenários de rede. Finalmente, o Capítulo 6 apresenta as considerações finais deste trabalho, incluindo as conclusões e as perspectivas de trabalho futuro.

Capítulo 2

Conceitos básicos

Este capítulo se destina a introduzir conceitos elementares utilizados nesta dissertação. Como este trabalho apresenta uma nova estratégia de roteamento em redes, os principais conceitos relacionados aos algoritmos de roteamento serão abordados na seção 2.1. Por apresentar conceitos básicos acerca dos algoritmos de roteamento, esta seção é fortemente embasada nos conceitos apresentados por Tanenbaum [2002] que é considerado uma das melhores referências sobre o assunto.

A arquitetura MPLS será utilizada neste trabalho para validar o algoritmo de roteamento proposto, assim a seção 2.2 se destina a descrever esta arquitetura, introduzindo-a no contexto das estratégias para provimento de QoS e destacando os principais conceitos a ela relacionados. Serão abordados também os principais trabalhos relacionados a esta arquitetura na seção 2.3, dando ênfase àqueles que tratam da alocação de rotas LSP explícitas na rede.

2.1 Algoritmos de roteamento

Uma pilha de protocolos é constituída de diversos softwares integrados que permitem a comunicação nas redes de telecomunicações. A cada protocolo é atribuída uma tarefa específica, e a integração entre os protocolos assegura a transmissão dos dados pelas redes. O modelo ISO/OSI foi proposto com o intuito de padronizar o desenvolvimento de protocolos de comunicação, estabelecendo camadas com aplicações bem definidas. A figura 2.1 ilustra este modelo de comunicação.

O foco desta seção se concentra na camada 3 deste modelo. A camada de Rede é responsável por definir como os dados trafegarão de sua origem até o destino em um procedimento denominado roteamento. Existem propriedades desejáveis que os algoritmos de roteamento devem tentar satisfazer ao definir rotas na rede, são elas: corretude, simplicidade, robustez, estabilidade, clareza e otimalidade. A satisfação de



Figura 2.1. Estrutura da pilha de protocolos do modelo ISO/OSI

todas estas propriedades pode não ser possível, devido à complexidade envolvida neste problema, no entanto, a qualidade das rotas está em função de sua satisfação.

Os algoritmos de roteamento podem ser agrupados em duas classes: adaptativos e não-adaptativos. Os algoritmos não-adaptativos não utilizam informações sobre o funcionamento da rede para realizar o roteamento, as rotas entre origem e destino são definidas baseadas em concepções pré-definidas. Por outro lado, os algoritmos adaptativos fazem uso das informações disponíveis sobre o tráfego e a topologia da rede para direcionar suas decisões no roteamento. Como consequência, a complexidade envolvida em algoritmos de roteamento adaptativos é maior do que aquela dos algoritmos não-adaptativos.

Uma outra classificação considerada para os algoritmos de roteamento é aquela que os agrupa entre algoritmos estáticos e dinâmicos. Os algoritmos estáticos são aqueles que consideram a rede antes de sua operação para definir o roteamento. Assume-se que todas as requisições a serem alocadas na rede são conhecidas e que não haverá alterações na topologia. O roteamento dinâmico é realizado pelos algoritmos que não têm um conhecimento prévio das requisições, estas podem surgir durante a operação da rede e alterações na topologia são permitidas.

Considerando as estratégias utilizadas para definir o roteamento, uma das mais conhecidas é a de caminho mínimo. Esta estratégia se destaca por sua simplicidade e pelos bons resultados obtidos. Para utilizá-la a rede deve ser modelada como um grafo cujos vértices são os roteadores e as arestas seus links. Para calcular os caminhos mínimos na rede, os algoritmos mais utilizados são os de Dijkstra [1959] que calcula o caminho mínimo entre um par de vértices e Floyd [1962] que se baseia em programação dinâmica para calcular caminhos mínimos para todos os pares de vértices.

Embora seja uma estratégia simples e eficiente, a definição dos custos envolvidos para que o caminho mínimo seja obtido pode ser uma tarefa complexa. Ao atribuir custos unitários às arestas, os algoritmos de caminho mínimo obtêm rotas com o menor número de saltos para atender às requisições. Mas diversas outras abordagens de atribuição de custos podem ser encontradas na literatura, como a distância euclidiana entre os roteadores, o atraso médio em cada link, a banda disponível, entre outros. Sendo possível também combinar diferentes métricas para na definição destes custos.

Já o algoritmo de inundação realiza o roteamento na rede utilizando uma característica simples para o encaminhamento dos pacotes. Dado um roteador X, o algoritmo de inundação consiste em encaminhar os pacotes que chegam em X para todos os roteadores adjacentes, exceto aquele que lhe enviou o pacote. Ao difundir o pacote a todos os roteadores da rede garante-se que o destino será alcançado. A figura 2.2 ilustra um exemplo de funcionamento do algoritmo de roteamento inundação. Os passos necessários para que um pacote saia de sua origem 1 e alcance o seu destino 8 são descritos por 2.2(a), 2.2(b) e 2.2(c).

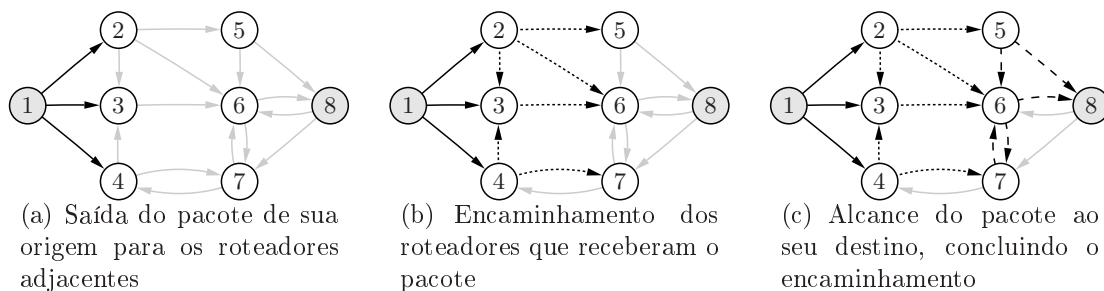


Figura 2.2. Etapas do algoritmo de roteamento inundação

Este algoritmo privilegia a propriedade de robustez no roteamento. A entrega dos pacotes ao destino pode ser realizada por diferentes roteadores, assegurando o sucesso da comunicação. No entanto, o encaminhamento dos pacotes a todos os roteadores adjacentes aumenta consideravelmente o tráfego nas redes, tornando este algoritmo inviável quando existem restrições de banda. Um outro inconveniente apresentado por este algoritmo é que deve existir um controle para que um pacote que seja recebido por um roteador por mais de uma vez não seja re-encaminhado, aumentando sua complexidade. No exemplo da figura 2.2, o roteador 3 recebe o pacote no primeiro passo do algoritmo vindo de 1, e novamente o recebe no segundo passo de 2 e de 4.

O roteamento através do algoritmo de inundação obtém rotas que seguirão o menor número de links na rede, embora diversos outros roteadores possam continuar recebendo o pacote, mesmo após este ter sido processado pelo destino.

Entre os algoritmos utilizados para realizar o roteamento dos pacotes na Internet está o de Vetor de Distâncias. Este é um algoritmo de roteamento dinâmico, que depende das informações a respeito do estado da rede para definir as rotas. A cada intervalo de tempo os roteadores enviam a seus adjacentes uma tabela (ou vetor) contendo informações locais sobre o roteamento. Cada roteador informa qual o menor custo para se alcançar qualquer outro roteador da rede, e por onde os pacotes devem ser encaminhados. A troca destes Vetores de Distância permite o mapeamento dos menores valores para cada roteador.

Diferentes métricas podem ser utilizadas para o cálculo do Vetor de Distâncias pelos roteadores. A mais comum delas é o número de saltos, mas também outras podem ser convenientemente adotadas como o atraso, o tamanho da fila dos roteadores, a distância física entre eles, entre outros. O intervalo de tempo em que os Vetores de Distância são atualizados também é uma questão de implementação a ser discutida. A atualização em um curto intervalo de tempo pode acarretar em *overhead* na rede devido ao número de mensagens trocadas pelos roteadores para atualizar os Vetores de Distância. Entretanto, caso o intervalo de atualização seja elevado, alterações na rede não serão percebidas e pacotes poderão ser perdidos.

O algoritmo de Vetor de Distâncias tem uma grande representação histórica no contexto de roteamento em redes. Ele foi o algoritmo de roteamento adotado pela ARPANET, rede precursora da Internet. Atualmente é utilizado pelo protocolo de roteamento RIP (*Routing Information Protocol*) [Hedrick, 1988] no roteamento dos pacotes na Internet.

Embora o algoritmo de Vetor de Distâncias funcione corretamente, ele apresenta algumas dificuldades quanto à sua convergência. O Vetor de Distâncias de cada roteador é enviado a seus adjacentes a cada intervalo de tempo, as informações são atualizadas e novamente repassadas aos adjacentes até que todos os roteadores da rede tenham informações atualizadas sobre o estado da rede. A necessidade de sucessivas trocas de mensagens entre os roteadores para atualizar os valores do Vetor de Distâncias pode levar o algoritmo a uma lenta convergência e, conseqüentemente, a rede pode ser incorretamente representada por um intervalo de tempo, prejudicando o roteamento.

Com o objetivo de corrigir este problema, foi proposto em 1979 um sucessor para o algoritmo de Vetor de Distâncias no roteamento da ARPANET. Este algoritmo é denominado de Estado do Link e soluciona o problema da lenta convergência. O algoritmo de Estado do Link pode ser dividido em 5 partes:

1. **Descobrir seus vizinhos e aprender sobre seus endereçamentos:** Ao ficar ativo na rede, o roteador precisa notificar sua atividade e saber a quais vizinhos está conectado e qual o endereço de acesso de cada um;

2. **Medir o custo do link entre os vizinhos:** Suponha que a métrica para o roteamento seja o atraso. Assim, o roteador deve estimar o tempo de atraso entre ele e seus adjacentes para que esta informação seja considerada no roteamento dos pacotes. Devido à variação do tráfego, este valor pode se alterar durante a operação da rede;
3. **Construir um pacote com as informações obtidas:** As informações armazenadas por cada roteador nos passos 1 e 2 devem ser compiladas em um pacote informativo, que possa servir para outros roteadores atualizarem suas informações sobre a rede;
4. **Enviar este pacote a todos os outros roteadores:** Para que não exista o problema da lenta convergência, os pacotes contendo informações sobre cada roteador são enviados periodicamente a todos os demais roteadores. O algoritmo de inundação é utilizado para esta finalidade;
5. **Calcular o menor caminho a todos os outros roteadores:** Uma vez que as informações de todos os roteadores estão disponíveis e atualizadas, o algoritmo calcula o menor caminho para se alcançar todos os demais roteadores. O algoritmo de Dijkstra é geralmente utilizado nesta etapa;

O algoritmo de roteamento do Estado do Link é atualmente utilizado em um grande número de redes. O protocolo de roteamento OSPF (*Open Shortest Path First*) [Moy, 1991] é um dos que utilizam este algoritmo, sendo o mais comum na Internet. Outro protocolo baseado no Estado do Link é o ISIS (*Intermediate System - Intermediate System*) [Oran, 1990] que realiza o roteamento em *backbones* de Internet.

2.2 A arquitetura MPLS

Os mecanismos de roteamento adotados pelo protocolo IP para o encaminhamento dos pacotes na rede se baseiam no serviço de melhor-esforço. Este serviço consiste em prover a comunicação com a máxima eficiência dos recursos da rede. O encaminhamento dos pacotes pelo menor caminho entre origem e destino é uma das políticas adotadas. Entretanto, ao utilizar o serviço de melhor-esforço não há garantias sobre a qualidade da comunicação com relação a algumas métricas na rede, como o atraso da comunicação e a perda de pacotes, o que pode dificultar o funcionamento de algumas aplicações.

Mas com o avanço das tecnologias computacionais e a disseminação das aplicações em rede, surge a necessidade de novos mecanismos de roteamento que consigam assegurar às aplicações que suas exigências com relação a algumas métricas da rede serão

atendidas. Neste sentido são propostas as arquiteturas para provimento de QoS nas redes TCP/IP. Entenda como arquitetura neste contexto um conjunto de mecanismos e padronizações com o intuito de fornecer serviços fim-a-fim oferecendo garantias de QoS ou assegurando a satisfação das métricas de SLA.

O motivo principal para propor melhorias no roteamento do protocolo IP é a sua popularidade. Este protocolo é o mais utilizado para comunicação em redes, devido à Internet, e ao agregar novas arquiteturas para prover QoS às aplicações será possível aproveitar toda a sua estrutura e escalabilidade. A figura 2.3 ilustra os principais avanços quanto ao provimento de QoS na arquitetura TCP/IP. A sigla RFC (*Request For Comment*) seguida de um número denota o documento onde os conceitos e padrões envolvidos na tecnologia são definidos.

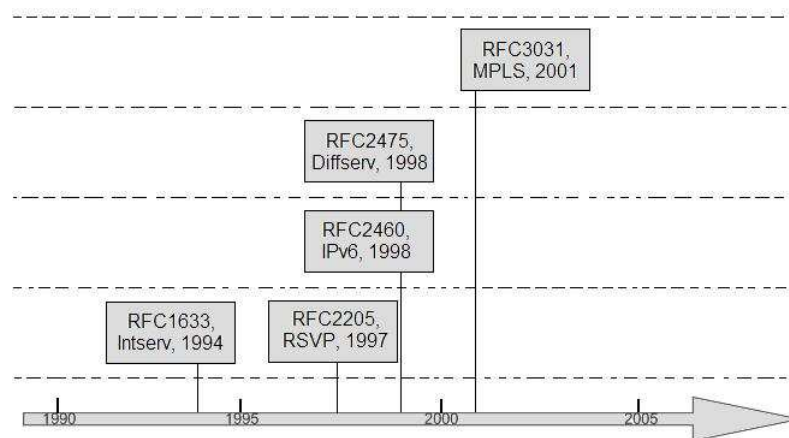


Figura 2.3. Evolução das arquiteturas para provimento de QoS em redes TCP/IP

Um dos primeiros avanços quanto ao provimento de QoS nas redes TCP/IP, foi a arquitetura IntServ (*Integrated Services*). Esta arquitetura oferece a característica de controle de admissão e reservas de recursos para os tráfegos nas redes TCP/IP [Evans e Filsfils, 2007]. Assim, é possível garantir banda suficiente para os tráfegos com esta exigência, como VoIP e video. Para a reserva dos recursos da rede, a arquitetura IntServ utiliza o protocolo RSVP (*ReSerVation Protocol*), que permite a reserva de banda para os tráfegos na rede. Por este motivo, IntServ e RSVP são muitas vezes considerados sinônimos.

O protocolo de roteamento padrão da Internet é o IP, mais especificamente IPv4 se referindo à sua 4ª versão. Devido à expansão da Internet, problemas de endereçamento surgiram no IPv4 e como alternativa de solução foi proposto em 1998 o IPv6, que elimina os problemas de endereçamento do seu antecessor. Entretanto, a idéia que o

IPv6 oferece melhor suporte de QoS às aplicações não é correta. Em sua especificação ele suporta todos os avanços disponíveis ao IPv4, mas não oferece novas funcionalidades para o provimento de QoS.

Já a arquitetura DiffServ (*Differentiated Services*) tem como principal objetivo o fornecimento de serviços diferenciados com relação às métricas de QoS impostas na rede. Desde seu surgimento é uma arquitetura amplamente utilizada para prover QoS em redes TCP/IP, sendo adotada tanto em redes empresariais quanto em provedores de serviços de internet (ISP). A arquitetura DiffServ acrescenta funcionalidades à arquitetura IntServ, pois além de permitir o controle de admissão e a reserva de recursos para os tráfegos na rede, possibilita também que os tráfegos tenham tratamento diferenciados uns dos outros. Esta diferenciação de serviços é feita ao definir Classes de Serviço (ou *Class of Service – CoS*) e pode incluir diferentes métricas de QoS na rede, como o atraso, a perda de pacotes e a disponibilidade do serviço.

A arquitetura MPLS (*MultiProtocol Label Switching*) é a mais recente tecnologia para a obtenção de QoS nas redes TCP/IP. Ela busca agregar todos os avanços anteriores à sua especificação aliados a novos conceitos de encaminhamento de pacotes nas redes IP, o que lhe garante maior eficiência no roteamento além de oferecer funcionalidades importantes para satisfação dos indicadores de QoS na rede.

O termo “MultiProtocol” da sigla MPLS é justificado pela possibilidade de integração desta arquitetura com diferentes protocolos de rede, porém sua integração com o protocolo IP se destaca, devido à sua abrangência na comunicação em redes. Todos os pacotes que entram em um domínio MPLS recebem um rótulo (*label*). As diretrizes de encaminhamento dos pacotes na rede se baseiam nestes rótulos, tornando desnecessária a verificação de outras informações existentes no cabeçalho dos pacotes. Ao sair do domínio MPLS, o rótulo é retirado do pacote que continuará a ser roteado pelos mecanismos disponíveis na rede. A figura 2.4 mostra a manipulação do rótulo nos pacotes ao entrar em um domínio MPLS.

Em uma rede não-comutada, a comunicação entre a origem e o destino do tráfego não é realizada por um caminho pré-estabelecido. Os pacotes podem seguir diferentes caminhos em função das condições da rede. O encaminhamento é definido com base em tabelas armazenadas pelos roteadores que definem o próximo salto de cada pacote ao ser processado. As tabelas de roteamento do protocolo IP são construídas utilizando a estratégia de melhor-esforço, minimizando o número de saltos da origem ao destino. Como exemplo, é possível citar os protocolos de roteamento OSPF e RIP, que se fundamenta em informações provenientes no estado do link e vetor de distâncias, respectivamente, para a construção das tabelas.

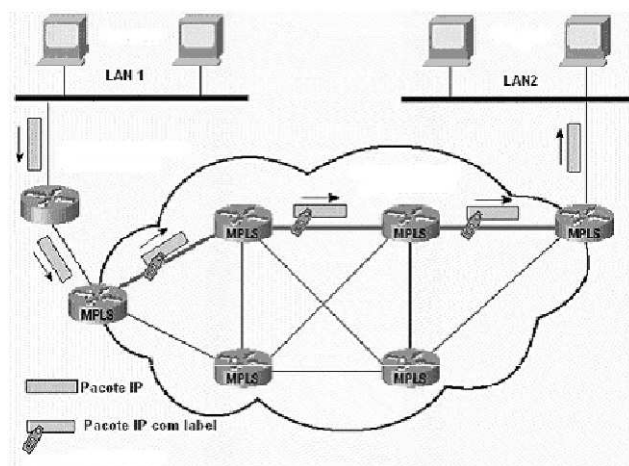


Figura 2.4. Manipulação do rótulo pelos roteadores da rede MPLS

Uma característica proporcionada pela arquitetura MPLS de grande relevância para a manutenção dos indicadores de QoS na rede é a possibilidade de se realizar o roteamento explícito dos pacotes. Neste caso, é possível definir explicitamente o caminho que um determinado tráfego deverá seguir da sua origem ao destino. Assim é possível assumir o roteamento dos pacotes na rede, tornando-o adequado às necessidades das aplicações envolvidas na comunicação.

A arquitetura MPLS é composta por software e hardware específicos capazes de suportar as funcionalidades por ela oferecidas. Entre os softwares se destacam os protocolos de comunicação e quanto ao hardware, são necessários roteadores que consigam tratar os pacotes rotulados do domínio MPLS de maneira adequada. A seguir, serão abordados os principais conceitos envolvidos no entendimento desta arquitetura, seguidos de uma breve explicação e sua contribuição no domínio MPLS:

- LSR (*Label Switching Router*); um LSR é um roteador que suporta os serviços da arquitetura MPLS. É responsável por gerenciar os *labels* utilizados para o roteamento dos pacotes no domínio MPLS. Quando um LSR insere um *label* em um pacote que entra no domínio MPLS, ele é denominado *Ingress LSR*. Já quando o *label* é retirado de um pacote que sai de um domínio MPLS, o LSR responsável é denominado *Egress LSR*.
- FEC (*Forwarding Equivalence Class*); Uma classe de pacotes IP cujo roteamento deve ser realizado da mesma maneira é denominada FEC. Nos esquemas de roteamento convencional, as FECs são definidas em cada roteador para realizar o encaminhamento dos pacotes. Já na arquitetura MPLS, as FECs são definidas ou verificadas apenas quando o pacote entra no domínio MPLS, a partir deste momento cada FEC é associada a um *label* que determina o seu roteamento ao

longo do domínio MPLS. Quando o pacote é encaminhado ao próximo LSR o seu label contém as informações necessárias para o roteamento.

- LSP (*Label Switched Path*); o caminho que um pacote deve seguir ao entrar em um domínio MPLS até deixá-lo será definido pelo *label* que lhe for atribuído. Nas redes MPLS este caminho é denominado LSP. A definição dos LSPs nas redes MPLS pode ser feita de maneira implícita, quando realizado pelos protocolos de roteamento desta arquitetura, ou explícita, quando o caminho é definido utilizando estratégias particulares de roteamento. A definição explícita de LSPs desperta grande interesse de empresas e pesquisadores, pois é possível agregar ao roteamento características particulares de cada domínio e satisfazer os indicadores de QoS esperados na rede.
- LDP (*Label Distribution Protocol*); ao definir uma FEC na rede e associá-la a um *label*, é necessário informar a todos os LSR do domínio MPLS qual o encaminhamento deve ser realizado aos pacotes com este *label*. A estratégia mais utilizada para este finalidade é o *broadcast*, onde todos os LSR trocam entre si as informações contidas em sua tabela de roteamento para a atualização. A distribuição das informações sobre o roteamento na rede é de responsabilidade do protocolo LDP.
- TTL (*Time-To-Live*); por um motivo qualquer, um pacote pode não ser corretamente encaminhado em um domínio MPLS. Como consequência este pacote poderia ser repassado por um tempo indeterminado entre os roteadores do domínio MPLS sem alcançar seu destino. Isto acarretaria em um processamento desnecessário dos roteadores com o pacote perdido. Para evitar este inconveniente, ao entrar em um domínio MPLS o pacote recebe um valor que determina o número máximo de saltos possíveis para este pacote no domínio MPLS. Este valor é o TTL. Sempre que atinge um LSR, o contador de saltos é incrementado e caso ultrapasse o TTL, o pacote é descartado.

A arquitetura MPLS suporta as funcionalidades necessárias para o provimento de QoS em redes TCP/IP, e por isto tem se destacado. A grande aceitação por parte de empresas e provedores de serviço pode ser justificada por esta arquitetura permitir a utilização de serviços como IntServ e DiffServ, aliados à funcionalidades inovadoras para as redes TCP/IP, como a definição explícita de rotas em seu domínio.

2.3 Trabalhos relacionados

Entre os trabalhos relacionados é possível encontrar contribuições em diferentes componentes da arquitetura MPLS, como melhorias em protocolos de controle, como o LDP, ou mesmo sugestões de alterações no cabeçalho dos pacotes que trafegam em seu domínio, com o intuito de melhorar a eficiência da comunicação.

A definição de rotas explícitas para os LSPs no domínio MPLS é também um tema que tem sido amplamente estudado em pesquisas. Esta funcionalidade permite explorar particularidades provenientes das aplicações, bem como características da rede para o roteamento dos pacotes, sendo um benefício particular da arquitetura MPLS. Neste contexto, a utilização de técnicas de otimização através de programação matemática tem se destacado, pois permite a elaboração de rotas ótimas na rede seguindo critérios estabelecidos pelos indicadores de QoS que se deseja atender.

Os trabalhos relacionados à esta área de pesquisa procuram dar um tratamento matemático ao problema de roteamento nas redes MPLS e então solucioná-lo. Diferentes técnicas de otimização podem ser aplicadas. Entretanto, a complexidade envolvida na solução destes problemas é elevada. No trabalho de Shao et al. [2006] é provado que o problema de se determinar as rotas LSP sujeito a mais de duas restrições de QoS é NP-Completo. Assim, as soluções propostas por estes trabalhos consistem em heurísticas para a alocação dos LSPs na rede que se aproximem da solução ótima.

A definição das rotas explícitas pode ser realizada ao considerar o problema em sua versão estática ou dinâmica. Na estática, os tráfegos a serem alocados são previamente conhecidos bem como os recursos disponíveis da rede, e as rotas LSP são determinadas considerando que não haverá alterações na rede ao longo do tempo. Na versão dinâmica as requisições surgem em função do tempo, sendo também possível tratar variações no comportamento da rede, como falhas de seus componentes ou interferência, por exemplo.

O provimento de QoS na rede geralmente baseia-se no atendimento de diferentes métricas da rede, como o atraso, a perda de pacotes ou a disponibilidade do serviço. Devido à multiplicidade de indicadores de QoS a considerar, é possível utilizar diferentes estratégias de otimização para a modelagem e solução do problema: a mono-objetivo e a multi-objetivo.

2.3.1 Otimização mono-objetivo

Dentre as estratégias mono-objetivo, um algoritmo clássico para este problema é o MIRA (*Minimum Interference Routing Algorithm*) [Kar et al., 2000], que realiza o roteamento dinâmico das rotas LSP buscando minimizar uma possível interferência nos

demais caminhos da rede. Utiliza a estratégia de fluxo máximo para solucionar o problema com um tempo computacional satisfatório, embora não seja na otimalidade. Um inconveniente apresentado por esta abordagem é a suposição que em apenas alguns LSR do domínio MPLS haverá surgimento de novas requisições de LSP, o que limita sua aplicação em diversos cenários.

No trabalho desenvolvido por Dias et al. [2003] foi tratado o problema de se determinar o menor caminho que minimize o atraso de transmissão dos dados, sujeito à restrição de largura de banda dos enlaces. Este problema também é NP-Completo, pois pode ser polinomialmente reduzido ao problema RSP (*Restricted Shortest-path Problem*) [Papadimitriou, 1997]. Para solucionar o problema foi proposta uma abordagem utilizando relaxação lagrangeana com heurísticas, denominada RLH. O tempo de execução do procedimento foi considerado razoável, entretanto, inviável para uma aplicação dinâmica. Como uma evolução deste trabalho, Dias et al. [2005] apresentaram uma heurística eficiente que permitiu sua avaliação em um contexto dinâmico.

Também com o objetivo de obter soluções eficientes, Oliveira e Mateus [2006] modelaram o problema de alocação estática das rotas através de programação linear inteira mista utilizando dois passos. No primeiro, é realizada a otimização baseada no balanceamento de carga da rede. A solução é utilizada como parâmetro de entrada no segundo passo, que objetiva agora minimizar o número de links utilizados restrito ao valor de balanceamento de carga obtido no primeiro passo. Devido ao grande tempo computacional necessário para se obter a solução ótima do problema, um algoritmo genético mono-objetivo foi proposto. Este trabalho também é apresentado na tese de Oliveira [2005].

2.3.2 Otimização multi-objetivo

Devido à multiplicidade de indicadores de QoS a serem considerados no roteamento, as técnicas de otimização multi-objetivo têm se apresentado como alternativas mais viáveis para a solução deste problema. Ao considerar os indicadores de QoS da rede como funções objetivo da otimização, é possível procurar por soluções que forneçam valores ótimos para estes indicadores otimizando-os de maneira independente. Existe um aumento da complexidade, pois o resultado da otimização fornece um conjunto de soluções para o problema, denominado conjunto Pareto-ótimo, diferente da otimização mono-objetivo, onde apenas uma solução é obtida como resultado.

Entre os primeiros a utilizar técnicas de otimização multi-objetivo neste contexto, embora não tenha sido no domínio MPLS, Knowles et al. [2000] utilizaram os algoritmos PAES (*Pareto Archived Evolution Strategy*) [Knowles e Corne, 2000b] e M-PAES (*Memetic Pareto Archived Evolution Strategy*) [Knowles e Corne, 2000a] para solucio-

nar o problema de roteamento estático. Os resultados apresentaram boas aproximações do conjunto Pareto-ótimo, em particular a busca local incluída no algoritmo memético trouxe melhorias significativas às soluções, mas a eficiência dos algoritmos não foi discutida.

O trabalho de Koyama et al. [2004] abordou o problema de roteamento dinâmico na rede, mas novamente a arquitetura MPLS não foi considerada. Foram consideradas como funções objetivo: (i) o custo e (ii) o atraso da comunicação, que propunham ser minimizadas. Um importante parâmetro que não foi levado em consideração neste trabalho é o balanceamento de carga da rede.

Aplicando a técnica de otimização multi-objetivo para o roteamento na arquitetura MPLS, Erbas e Erbas [2003] apresentaram uma solução para o problema estático. São consideradas três funções objetivo: (i) o custo da comunicação, (ii) o balanceamento de carga na rede e (iii) o número de LSPs necessários para atender a todas as requisições. Para solucionar o problema foi utilizado o método de pesos lexicográficos de Chebyshev [Steuer, 1985], que por se tratar de um método exato demanda um elevado tempo computacional e impede que esta estratégia seja adaptada para o contexto dinâmico de otimização.

Em Shao et al. [2006] a alocação de rotas LSP é considerada em um ambiente com diferenciação de serviços (*DiffServ*), também através de uma abordagem multi-objetivo. Os objetivos considerados também são três: (i) o balanceamento de carga na rede, (ii) a taxa de rejeição dos pacotes e (iii) o atraso na comunicação. As simulações demonstraram um bom desempenho do procedimento proposto, porém, não foram apresentados resultados que permitissem verificar o seu comportamento em um ambiente dinâmico, e além do mais o processo de tomada de decisão não foi explicitado.

É possível observar que as técnicas de otimização são amplamente aplicadas ao problema de roteamento, incluindo a arquitetura MPLS. Em particular, as técnicas de otimização multi-objetivo têm permitido grandes avanços nesta área, pois sua utilização permite considerar múltiplos indicadores de QoS simultaneamente no processo de otimização, tornando-a mais adequada ao problema.

Não foram encontrados na literatura trabalhos que tratem da otimização multi-objetivo em um contexto dinâmico de otimização, onde deve existir um processo de tomada de decisão eficiente para selecionar dinamicamente as melhores soluções para o problema. Este trabalho procura contribuir neste contexto e estuda o roteamento dinâmico em redes MPLS utilizando a otimização multi-objetivo.

Capítulo 3

Formulação matemática

Conforme descrito na seção 2.3, o problema de definir rotas explícitas em redes MPLS é normalmente abordado matematicamente, devido à complexidade envolvida em sua solução. Assim, são aplicadas técnicas de teoria dos grafos e programação matemática que permitem modelar e solucionar o problema.

Este capítulo mostrará a formulação matemática proposta para o problema de definir rotas explícitas em redes MPLS. Inicialmente, na seção 3.1 serão descritas algumas definições utilizadas para a representação do problema, então a seção 3.2 exibirá a formulação matemática mono-objetivo proposta na tese de Oliveira [2005], que foi uma das motivações para a realização deste trabalho. A formulação multi-objetivo do problema será abordada na seção 3.3, onde serão introduzidos os conceitos básicos desta técnica de otimização e exibido o modelo matemático proposto neste trabalho.

3.1 Definições

A rede a ser otimizada será representada por um grafo $G = (V, A)$, onde $V = \{1, 2, \dots, \|V\|\}$ são os roteadores do domínio MPLS, ou LSR, e $A = \{(i, j)\}$ são os links existentes entre os LSR. A capacidade de banda de cada link (i, j) será representada no modelo como B_{ij} e os links são unidirecionais.

As requisições de comunicação são representadas por (o^k, d^k, b^k) , sendo que o^k e d^k representam respectivamente os roteadores LSR de origem e de destino do tráfego para a k -ésima requisição. Enquanto b^k indica a quantidade de banda que deve ser reservada para esta requisição. O conjunto de todas as requisições a serem otimizadas é denotado por R .

As formulações tratadas neste trabalho permitem que requisições sejam rejeitadas, o que pode acontecer por restrições de recursos da rede, ou mesmo para privilegiar um determinado indicador de QoS na rede. No entanto, é possível definir um número

mínimo de requisições que devem ser atendidas no modelo, que será representado por C , e deve ser fornecido como parâmetro da otimização.

3.2 Formulação matemática mono-objetivo

Na tese de Oliveira [2005] foi proposto um modelo matemático mono-objetivo para solucionar o problema de alocação de rotas LSP nas redes MPLS. Este modelo, denominado *two-step model* (modelo de dois passos), propõe a solução de dois problemas mono-objetivo para a alocação das rotas LSP na rede.

O primeiro passo procura por valores ótimos para o balanceamento de carga na rede. Para isto, é formulado um problema de programação linear inteira misto do tipo *Min – Max*, que obtém o valor mínimo para o link com máxima utilização na rede. Este modelo é denominado *P1* e pode ser descrito como:

$$P1: \text{Min } \alpha$$

sujeito a:

$$\sum_{k \in R} b^k x_{ij}^k \leq \alpha B_{ij} \quad \forall (i, j) \in A \quad (3.1)$$

$$\sum_{(i,j) \in \Gamma_i^+} x_{ij}^k - \sum_{(l,i) \in \Gamma_i^-} x_{li}^k = \begin{cases} 1 & (o_k) \\ -1 & (d_k) \\ 0 & \end{cases} \quad \forall i \in V, k \in R \quad (3.2)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j) \in A, \forall k \in R \quad (3.3)$$

Nesta modelagem a variável α representa o percentual de utilização do link mais congestionado da rede. Portanto, a sua minimização implica que o balanceamento de carga da rede é obtido, ou seja, a carga está distribuída de maneira uniforme entre os links da rede. A variável de decisão x_{ij}^k assumirá o valor 1 caso o link (i, j) seja utilizado para atender a requisição de índice k , caso contrário seu valor será 0.

As restrições 3.1 asseguram que a variável de decisão α represente a maior carga entre todos os links (i, j) da rede. Caso a solução para o problema apresente $\alpha \geq 1$, a solução do problema não será viável, pois a exigência dos tráfegos é maior que a capacidade do link. Em 3.2 são exibidas as restrições de conservação de fluxo, que garantem que todo o tráfego que entra em um vértice deverá ser o mesmo a sair, exceto na origem e destino da requisição. As restrições 3.3 definem a integralidade das variáveis de decisão x_{ij}^k definindo seus possíveis valores em 0 ou 1.

O segundo passo da otimização tem como parâmetro de entrada o resultado obtido no primeiro passo. O valor de α obtido em $P1$ será utilizado em $P2$ para assegurar que a solução obtida pelo segundo passo terá o balanceamento de carga ótimo na rede, enquanto procura-se minimizar outra função objetivo. O modelo $P2$ é a seguir exibido:

$$P2 : \sum_{(i,j) \in A} \sum_{k \in R} x_{ij}^k + \sum_{k \in R} M(1 - a^k)$$

sujeito a:

$$\sum_{k \in R} b^k x_{ij}^k \leq \alpha^* B_{ij} \quad \forall (i, j) \in A \quad (3.4)$$

$$\sum_{(i,j) \in \Gamma_i^+} x_{ij}^k - \sum_{(l,i) \in \Gamma_i^-} x_{li}^k = \begin{cases} 1 & (o_k) \\ -1 & (d_k) \\ 0 & \end{cases} \quad \forall i \in V, k \in R \quad (3.5)$$

$$\sum_{k \in R} a^k \geq C \quad (3.6)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j) \in A, \forall k \in R \quad (3.7)$$

$$a^k \in \{0, 1\} \quad \forall k \in R \quad (3.8)$$

O modelo $P2$ tem características muito semelhantes à $P1$, pois tratam do mesmo contexto de otimização. No entanto, neste segundo passo da otimização, α^* não se trata de uma variável de decisão, mas sim de um parâmetro que é calculado ao se otimizar $P1$. Sendo assim, $P2$ consiste em procurar por soluções com balanceamento de carga ótimo na rede.

Como função objetivo, procura-se otimizar diferentes entidades responsáveis pelo provimento de QoS na rede: o número de links utilizados na comunicação e também o máximo atendimento das requisições. Isto é feito ao somar os valores obtidos com cada uma das funções.

Em $P2$ também está presente a restrição 3.6 com a finalidade de assegurar que a solução do problema atenderá um número mínimo de requisições, denotado por C . A variável de decisão a^k indica se a k -ésima requisição foi atendida, assumindo valor 1, ou 0 caso contrário.

Observe que este modelo procura por valores ótimos para diferentes entidades para provimento de QoS. Porém, isto é feito através de uma modelagem do problema em dois passos de otimização, além de combinar os valores de diferentes entidades em uma mesma função objetivo em $P2$. A utilização da otimização multi-objetivo pode auxiliar na alocação dos LSPs na rede neste contexto, pois permitirá que diferentes funções objetivo sejam otimizadas independentemente em apenas um modelo.

3.3 Formulação matemática multi-objetivo

Esta seção tratará da formulação matemática do problema de alocação dos LSPs na rede MPLS, através da abordagem de otimização multi-objetivo. Inicialmente, serão descritos os principais conceitos relacionados a esta técnica de otimização e então será mostrado o modelo matemático proposto.

3.3.1 Conceitos iniciais

A otimização multi-objetivo ou multi-critério, é caracterizada pela presença de duas ou mais funções objetivo para serem maximizadas ou minimizadas, sujeito às restrições do problema. Matematicamente, o problema de otimização multi-objetivo pode ser enunciado como [Deb, 2001]:

$$\text{Min } f_m(x)$$

sujeito a:

$$g_j(x) \geq 0, \quad j = \{1, \dots, J\} \quad (3.9)$$

$$h_k(x) = 0, \quad k = \{1, \dots, K\} \quad (3.10)$$

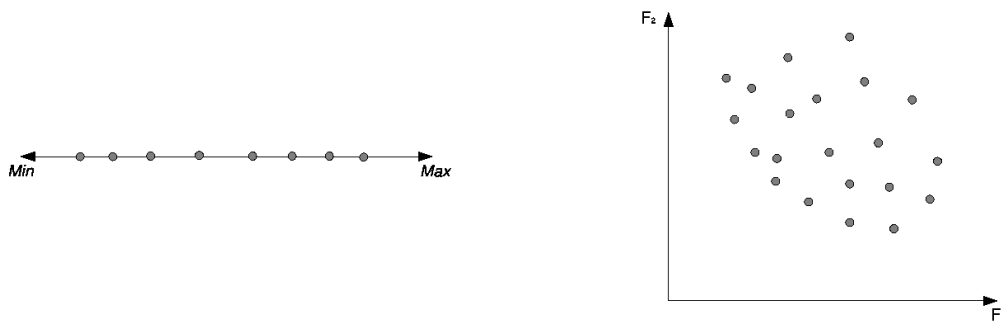
$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = \{1, \dots, n\} \quad (3.11)$$

O vetor n -dimensional x representa as variáveis de decisão do problema, sendo composto pelas variáveis $x_i, i = \{1, \dots, n\}$. Este vetor é também conhecido como solução do problema. Já $f_m, m = \{1, \dots, M\}$ denota um vetor com M funções objetivo que deverão ser minimizadas ou maximizadas. Embora algumas funções objetivo possam ser maximizadas enquanto outras minimizadas, para realizar a otimização é necessário que todas as funções tenham apenas o propósito de maximização ou minimização, o que pode ser obtido através de operações algébricas elementares.

Para que o problema seja considerado factível, é necessário que ele satisfaça a todas as J restrições de desigualdade e as K restrições de igualdade do problema, embora seja possível que um dos conjuntos de restrições 3.9 ou 3.10 seja vazio. Finalmente, as restrições 3.11 indicam a possibilidade de se definir limites inferiores e superiores para os valores das variáveis de decisão x_i . Caso a solução apresente valores que extrapolem os limites, o problema é considerado inviável. O conjunto de todos os vetores x factíveis para o problema é dado por X .

3.3.1.1 O espaço de soluções

Uma diferença fundamental existente entre a otimização mono-objetivo e a multi-objetivo é o espaço de soluções de cada uma. Na otimização mono-objetivo, cada vetor $x \in X$ que satisfaz as restrições do problema leva a uma solução em um espaço unidimensional. Já na otimização multi-objetivo, cada vetor solução $x \in X$ leva a soluções em um espaço multi-dimensional, denominado *espaço de objetivos* (Z). A figura 3.1 ilustra a diferença entre os espaços de solução da otimização mono-objetivo e multi-objetivo.



(a) Espaço de soluções uni-dimensional da otimização mono-objetivo

(b) Espaço de soluções bi-dimensional de um problema multi-objetivo

Figura 3.1. Espaços de soluções para problemas de otimização mono-objetivo e multi-objetivo

Na otimização mono-objetivo, devido ao espaço de soluções ser unidimensional, procura-se por um vetor x cujos valores de coordenada levem a uma solução com valor máximo ou mínimo no espaço de soluções. Já a otimização multi-objetivo lida com o espaço multi-dimensional das variáveis de decisão e também com o espaço de objetivos multi-dimensional. É possível que exista um conjunto de vetores x no espaço das variáveis que levem a soluções ótimas no espaço dos objetivos, o que justifica a maior complexidade desta abordagem de otimização.

3.3.1.2 O conjunto Pareto-ótimo

O método de otimização multi-objetivo tem como princípio a investigação de vetores $x \in X$ que levem às melhores soluções no espaço de objetivos. Assim, $f_m(x)$, $m = \{1, \dots, M\}$ tem a finalidade de mapear os valores do vetor n -dimensional x para o espaço de objetivos Z . Suponha que o vetor x está inserido nos números reais, então o mapeamento seria dado como $f_m(x) : \mathfrak{R}^n \rightarrow \mathfrak{R}^M$.

A procura por vetores $x \in X$ que solucionem o problema de otimização multi-objetivo é realizada ao avaliar as soluções utilizando a definição de dominância. Para definir dominância é necessário formalizar a notação dos símbolos \leq e \neq para os conjuntos da otimização. Seja x e y dois vetores n -dimensionais de um conjunto C , tem-se que

$$x \leq y \Rightarrow \{x_i \leq y_i, i = 1, \dots, n\} \text{ e}$$

$$x \neq y \Rightarrow \{\exists i \mid x_i \neq y_i\}.$$

Assim a definição de dominância segundo Takahashi [2004] pode ser enunciada como:

Definição 1 (Dominância): *Diz-se que o ponto $x_1 \in X$ domina o ponto $x_2 \in X$ se $f_m(x_1) \leq f_m(x_2)$ e $f_m(x_1) \neq f_m(x_2)$. Equivalentemente, diz-se que $f_m(x_1) \in Z$ domina $f_m(x_2) \in Z$ nestas mesmas condições.*

A otimização das soluções em um problema multi-objetivo está diretamente relacionada ao conceito de dominância. No processo de otimização procura-se por soluções não-dominadas até que todas as soluções viáveis $x \in X$ sejam investigadas. Neste sentido, a próxima definição de Takahashi [2004] permite compreender o conjunto de soluções ótimas da otimização multi-objetivo:

Definição 2 (Solução Pareto-ótima): *Diz-se que o ponto $x^* \in X$ é uma solução Pareto-ótima do problema de otimização multi-objetivo se não existe qualquer outra solução $x \in X$ tal que $f_m(x) \leq f_m(x^*)$ e $f_m(x) \neq f_m(x^*)$, ou seja, se x^* não é dominado por nenhum outro ponto factível.*

Esta definição atesta que, considerando o conjunto de soluções factíveis para o problema multi-objetivo, todas aquelas não-dominadas compõe o conjunto Pareto-ótimo da otimização multi-objetivo. Mas isto não limita o tamanho deste conjunto, que para alguns problemas pode ser infinito. A próxima ilustração, vista na figura 3.2, mostra dois conjuntos de soluções para um problema multi-objetivo, um composto por soluções dominadas e não-dominadas enquanto o outro se trata de um conjunto Pareto-ótimo, apenas com soluções não-dominadas.

Diversos problemas reais onde deseja-se maximizar ou minimizar recursos levam a problemas de otimização multi-objetivo. Isto devido à presença de diferentes critérios que se busca otimizar simultaneamente. Um exemplo intuitivo que ilustra esta afirmação pode ser encontrado na compra de um notebook. Neste caso, dois objetivos considerados frequentemente são o preço e o desempenho do notebook. A figura 3.3 apresenta um possível conjunto de soluções para este problema considerando um espaço dos objetivos bidimensional.

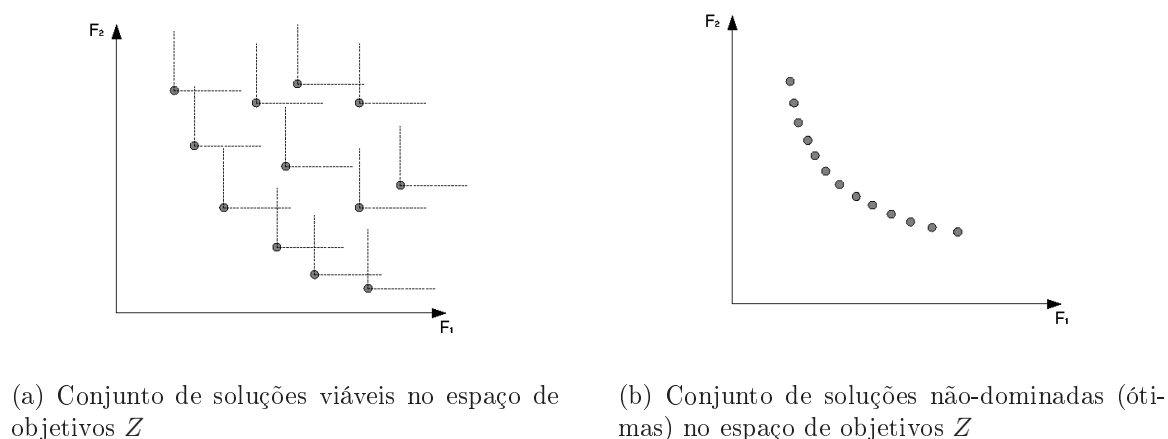


Figura 3.2. Conjuntos de soluções no espaço dos objetivos

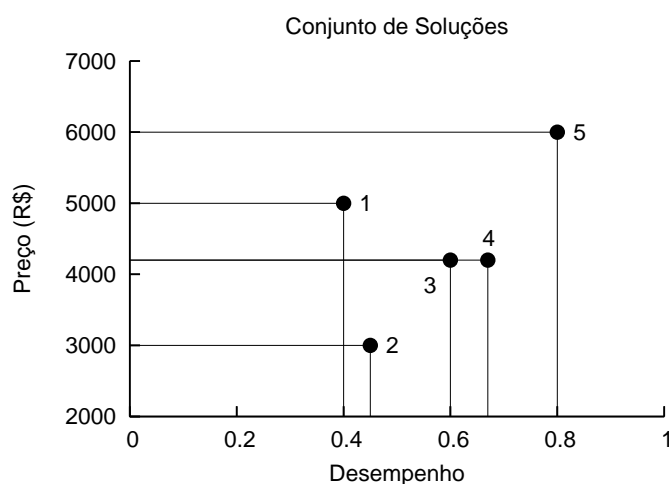


Figura 3.3. Exemplo de conflitos entre dois objetivos na compra de um notebook

Deseja-se maximizar o desempenho e simultaneamente procura-se por soluções que minimizem o preço do produto. Após uma pesquisa, diferentes opções de solução são encontradas. A solução 5 será mais conveniente àqueles que têm poucas restrições financeiras ou que realmente necessitam de um grande poder de processamento, enquanto a solução 2 privilegia aqueles que têm menor poder de compra e dificuldades de orçamento.

Ainda sobre a figura 3.3 é possível observar que não é conveniente escolher as soluções 1 e 3 pois existem outras soluções que oferecem pelo mesmo preço ou menos, um melhor desempenho. As soluções 1 e 3 são, portanto, soluções dominadas e não pertencem ao conjunto de soluções ótimas, ou conjunto Pareto-ótimo.

Observa-se que os possíveis conflitos existentes ao se otimizar simultaneamente as funções objetivo do problema levam à diversidade das soluções no conjunto

Pareto-ótimo. A solução obtida caso não existissem conflitos ao otimizar as funções objetivo é denominada utópica e é descrita através da próxima definição de Takahashi [2004]:

Definição 3 (Solução utópica): A solução utópica y^* do problema de otimização multi-objetivo é definida como:

$$y^{*i} = f_i(x^i), \quad i = \{1, \dots, M\}$$

onde:

$$x^i = \underset{x \in X}{\operatorname{arg\,min}} f_i(x)$$

A solução utópica é aquela que apresenta valores ótimos para cada uma das funções objetivo do problema, e recebe este nome devido à impossibilidade de ser obtida quando existem conflitos ao otimizar todas as funções objetivo simultaneamente.

Esta solução pode ter diferentes finalidades no contexto de otimização multi-objetivo, como permitir fazer estimativas da distância entre as soluções no espaço dos objetivos ou mesmo ser utilizada como métrica para a implementação de tomada de decisão no conjunto Pareto-ótimo. Um exemplo de solução utópica é exibido na figura 3.4, considerando um espaço de objetivos bidimensional.

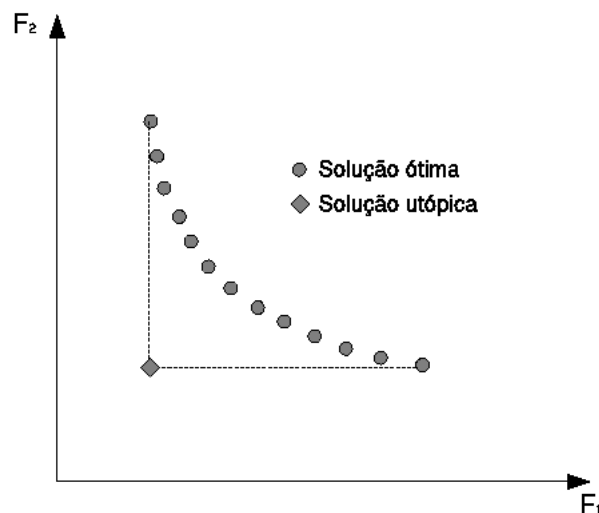


Figura 3.4. Solução utópica de um problema de otimização multi-objetivo

3.3.2 Modelo matemático

O modelo proposto neste trabalho trata do mesmo problema apresentado em Oliveira [2005], porém através da abordagem de otimização multi-objetivo. Ao invés de tratar o problema de otimização em um modelo de dois passos inter-dependentes, neste trabalho é realizada uma formulação matemática multi-objetivo que considera três funções objetivo.

Todas as definições expostas na seção 3.1 se aplicam tanto ao problema mono-objetivo quanto ao multi-objetivo. O grande diferencial apresentado pela otimização multi-objetivo é a possibilidade de se considerar simultaneamente todas as funções objetivo do problema, a saber:

- a quantidade de links utilizados para atender cada uma das requisições;
- a quantidade de rejeição das requisições na rede;
- o balanceamento de carga na rede;

Quanto à modelagem, as variáveis de decisão x_{ij}^k assumem valor 1 caso a k -ésima requisição passe pelo link (i, j) ou 0 caso contrário, a^k que indica o atendimento da k -ésima requisição, assumindo valor 1 em caso positivo, senão 0. Além de α que assumirá valores contínuos no intervalo entre $[0, 1]$ e representa o percentual de utilização do link mais utilizado da rede. Há também o parâmetro C que indica a quantidade mínima de requisições que devem ser atendidas. O modelo multi-objetivo é expresso como:

$$Min \begin{cases} F_1 = \sum_{k \in R} \sum_{(i,j) \in A} x_{ij}^k \\ F_2 = \sum_{k \in R} (1 - a^k) \\ F_3 = \alpha \end{cases}$$

sujeito a:

$$\sum_{k \in R} b^k x_{ij}^k \leq \alpha B_{ij} \quad \forall (i, j) \in A \quad (3.12)$$

$$\sum_{(i,j) \in \Gamma_i^+} x_{ij}^k - \sum_{(l,i) \in \Gamma_i^-} x_{li}^k = \begin{cases} a^k & (o_k) \\ -a^k & (d_k) \\ 0 & \end{cases} \quad \forall i \in V, k \in R \quad (3.13)$$

$$\sum_{k \in R} a^k \geq C \quad (3.14)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j) \in A, \forall k \in R \quad (3.15)$$

$$a^k \in \{0, 1\} \quad \forall k \in R \quad (3.16)$$

$$\alpha \in [0, 1] \quad (3.17)$$

Quanto às funções objetivo, tem-se que F_1 representa o número de links utilizados para atender as requisições do conjunto R . Quanto menos links forem utilizados mais rapidamente os dados trafegarão da origem ao destino, logo este objetivo será minimizado. F_2 representa a quantidade de requisições de R que serão rejeitadas durante a otimização. Espera-se que este número também seja minimizado, de modo a atender o maior número possível de requisições. E F_3 diz respeito ao percentual de utilização dos links da rede, e como pretende-se balancear a carga, espera-se que este valor também seja minimizado para que a quantidade de dados trafegando nos links esteja equilibrada.

É possível observar os conflitos em atender simultaneamente a todas as funções objetivo, principalmente ao considerar F_1 e F_3 , uma vez que ao se otimizar uma delas a outra tende a ter seu resultado comprometido. Ao verificar a influência de F_2 em F_1 e F_3 observa-se também uma relação de compromisso. Ao minimizar F_2 , tanto F_1 quanto F_3 tendem a ter seus valores aumentados uma vez que a alocação de uma requisição na rede consumirá seus recursos. A não alocação de uma requisição privilegiará F_1 e F_3 mas não F_2 . Esta impossibilidade de se obter valores mínimos para as três funções objetivo que justificam a diversidade de soluções no conjunto Pareto-ótimo, existindo mais de uma solução ótima, que deverá ser escolhida em função do contexto de otimização.

Quanto às restrições, em (3.12) verifica-se que a quantidade de dados a trafegar em um link (i, j) não pode ultrapassar sua capacidade de banda B_{ij} . As restrições clássicas de conservação de fluxo estão presentes em (3.13), que determinam que o tráfego a entrar em um vértice deve ser o mesmo a sair, exceto na origem e no destino da requisição. A restrição (3.14) permite limitar o número de rejeições, o que pode ser de grande relevância em alguns ambientes. Finalmente as restrições (3.15), (3.16) e (3.17) limitam o espaço das variáveis de decisão envolvidas.

3.3.3 Alternativas de solução

A solução de problemas de otimização multi-objetivo é de elevada complexidade computacional. Isto porque, tanto o espaço das variáveis de decisão quanto o espaço dos objetivos é multi-dimensional, o que demanda um número maior de avaliações para obter todas as soluções do problema, aumentando sua complexidade.

Entre as alternativas de solução exata para os problemas de otimização multi-objetivo, duas técnicas se destacam por serem mais difundidas na literatura e também pela simplicidade da abordagem apresentada. São elas:

- **Ponderação dos objetivos (P_λ);** Esta técnica consiste em definir multiplicadores $\lambda_i \geq 0$, $i = \{1, \dots, M\}$ para cada função objetivo tal que $\sum_{i=1}^M \lambda_i = 1$. Então

as M funções objetivo do problema são multiplicadas por seus respectivos λ_i e somadas, transformando o problema multi-objetivo em um problema de otimização mono-objetivo. A solução de cada problema mono-objetivo resultará em um ponto no conjunto Pareto-ótimo. As desvantagens associadas à esta técnica é que ela não se aplica adequadamente a problemas não-convexos e também ao solucionar problemas com valores distintos de λ_i , o mesmo ponto do conjunto Pareto-ótimo pode ser obtido.

- **Problema ϵ -Restrito (P_ϵ);** Através da técnica P_ϵ , o problema de otimização multi-objetivo é também convertido em um problema de otimização mono-objetivo. Mas para isto $M - 1$ funções objetivo são consideradas como restrições do problema e apenas uma função objetivo será otimizada. Para cada função objetivo considerada como restrição será atribuído um valor ϵ_i , $i = \{1, \dots, (M - 1)\}$ que consistirá no limite da restrição. A variação dos valores de ϵ_i gerando novos problemas de otimização mono-objetivo resultarão em diferentes pontos do conjunto Pareto-ótimo, solucionando o problema multi-objetivo. Como desvantagem desta técnica é possível citar o aumento na complexidade do problema ao considerar outras $M - 1$ restrições na otimização e também a possibilidade de inserir valores inviáveis para os valores de ϵ_i nas novas restrições. No entanto, não existem limitações quanto a utilização desta técnica em problemas não-convexos.

Capítulo 4

Heurística - Algoritmo Genético multi-objetivo

Embora a solução para o problema de roteamento em redes MPLS utilizando a formulação matemática apresentada no capítulo 3 seja ótima, o tempo computacional necessário para obtê-la é elevado. Portanto, sua aplicação está limitada ao problema de roteamento estático, onde o processo de otimização não é influenciado por aspectos temporais como o surgimento de novas requisições ou mesmo mudança no comportamento da rede.

Para solucionar este problema de roteamento é necessário um procedimento que seja escalável e, sobretudo, eficiente. Estes requisitos sugerem a implementação de uma heurística que obtenha aproximações do conjunto de soluções ótimas para o problema com um baixo custo computacional. Neste contexto, foi implementado um Algoritmo Genético como heurística para realizar o roteamento.

Os Algoritmos Genéticos são técnicas de busca inspiradas em mecanismos de seleção e genética natural [Goldberg, 1989]. Para realizar a busca pelo espaço de soluções do problema, combina-se os conceitos de adaptação seletiva, troca de material genético e sobrevivência dos indivíduos mais aptos. A representação dos indivíduos é feita por cromossomos que são compostos pelas variáveis de decisão do problema. Para a evolução do material genético dos indivíduos são utilizadas estratégias de cruzamento e mutação. A população de indivíduos evolui ao longo das gerações para pontos no espaço de busca que privilegiem os valores de suas funções de aptidão.

Por este problema envolver diferentes funções objetivo, foi implementado um algoritmo genético multi-objetivo que se baseia na meta-heurística NSGA-II [Deb et al., 2000]. A aproximação dos indivíduos ao conjunto Pareto-ótimo é realizada a cada iteração, sendo que ao final das gerações é obtido um conjunto de soluções aproximadas para o problema.

No algoritmo, além daquelas características envolvidas no problema de roteamento estático, algumas particularidades são consideradas devido ao contexto dinâmico de otimização. A principal delas é a possibilidade de considerar diferentes classes de serviço na otimização das rotas na rede. Serão consideradas três classes de serviço com as seguintes características:

- **Classe A;** as requisições associadas a esta classe terão prioridade na alocação, podendo ser rejeitadas apenas se não houver banda disponível para sua alocação na rede.
- **Classe B;** uma característica principal associada a esta classe é que suas requisições devem ter menor atraso, no entanto, é possível rejeitar requisições caso as rotas não satisfaçam estas exigências.
- **Classe C;** esta classe estará associadas àquelas requisições que não apresentam exigências para serem alocadas na rede.

Caso a diferenciação de classes seja suportada pela rede, toda requisição estará associada a uma classe que será tratada de maneira distinta no processo de otimização. Esta característica proporcionará flexibilidade na definição das rotas LSP na rede, além de se aproximar mais dos cenários reais de comunicação nas redes.

Nas próximas seções deste capítulo é detalhada a implementação do algoritmo genético proposto. Inicialmente, a seção 4.1 descreverá a meta-heurística NSGA-II e então as etapas de implementação do algoritmo serão abordadas das seções 4.2 a 4.5, a saber: representação genética, população inicial, seleção, cruzamento e mutação. A seção 4.6 tratará da complexidade computacional do algoritmo e finalmente, a seção 4.7 mostrará seu desempenho verificando a qualidade das aproximações obtidas e sua eficiência.

4.1 NSGA-II

O NSGA-II apresenta duas contribuições relevantes com respeito à aproximação das soluções ao conjunto Pareto-ótimo. A primeira delas é a classificação dos indivíduos da população quanto à sua dominância. A relação de dominância é uma estratégia utilizada por outros algoritmos genéticos multi-objetivo para classificar os indivíduos da população, porém o NSGA-II sugere o método *fast-non-dominated-sorting* que garante eficiência nesta etapa do algoritmo.

No *fast-non-dominated-sorting* cada indivíduo p armazena duas entidades, n_p é o número de indivíduos que dominam p e S_p denota uma lista com todos os indivíduos

dominados por p . Assim, para calcular a relação de dominância de cada indivíduo na população são necessárias $O(MN)$ comparações, sendo M o número de funções objetivo do problema e N a quantidade de indivíduos na população. Conseqüentemente, a relação de dominância entre todos os indivíduos tem complexidade $O(MN^2)$.

Como aqueles indivíduos não dominados estão mais próximos do conjunto Pareto-ótimo, estes são priorizados para seguirem à próxima geração, portanto, os indivíduos com $n_p = 0$ seguem. Para definir os outros indivíduos da próxima geração, para cada indivíduo selecionado p com $n_p = 0$, percorre-se sua lista S_p e decrementa-se o valor de n_p daqueles indivíduos dominados por p . Assim, é possível considerar agora um novo grupo de indivíduos não-dominados, desconsiderando aqueles já selecionados para a próxima geração. Este processo define fronteiras de indivíduos que devem ser completamente inseridas na população da próxima geração.

Quando não é possível inserir uma fronteira por completo, a segunda contribuição apresentada pelo NSGA-II é o operador *crowding-distance-sorting* que é responsável por classificar os indivíduos quanto à sua distribuição no conjunto de soluções. Aquelles indivíduos que se encontrarem mais espalhados são priorizados, com o objetivo de obter uma aproximação do conjunto Pareto-ótimo mais distribuída, evitando-se a aglomeração de soluções no espaço de objetivos. A figura 4.1 foi retirada de Deb et al. [2000] e ilustra as etapas descritas no processo definição da próxima geração utilizando o NSGA-II.

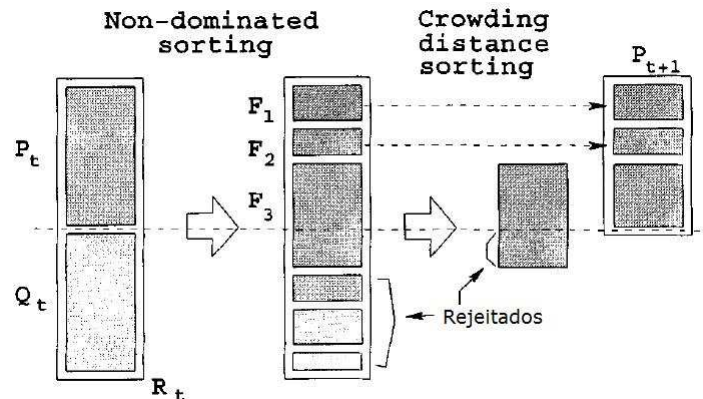


Figura 4.1. Definição dos indivíduos na próxima geração pelo NSGA-II

O algoritmo NSGA-II é fundamentado nos conceitos descritos, mas também conta com as etapas tradicionais dos algoritmos genéticos para obter os indivíduos e operá-los afim de obter melhorar as soluções para o problema. Sua entrada consiste de dois parâmetros, que são o tamanho da população (T_{pop}) e a quantidade de gerações (G_{max}). Como saída, tem-se o conjunto de indivíduos mais próximos da fronteira de Pareto (P) obtidos durante a execução do algoritmo.

Quanto ao seu funcionamento, é definida uma população inicial P antes de iniciar o processo evolutivo do algoritmo. Após isto, o conjunto Q armazenará os indivíduos provenientes do cruzamento, e como o NSGA-II é um algoritmo elitista, o conjunto R representa todos os indivíduos do algoritmo, sendo a união dos conjuntos dos pais P e dos filhos Q . A mutação é realizada e então utiliza-se os operadores *fast-non-dominated-sorting* e *crowding-distance-sorting* para definir os melhores indivíduos que seguirão à próxima geração. O algoritmo NSGA-II é descrito a seguir:

Algoritmo 1: NSGA-II	
	Entrada: T_{pop}, G_{max}
	Saída: P
1	início
2	$P = populacaoInicial(T_{pop});$
3	para $g \leftarrow 1$ até G_{max} faça
4	$Q \leftarrow cruzamento(P);$
5	$R \leftarrow P \cup Q;$
6	$mutacao(R);$
7	$P \leftarrow \emptyset;$
8	$\chi \leftarrow fast - non - dominated - sorting(R);$
9	repita
10	$P \leftarrow \chi;$
11	$S \leftarrow (R - \chi);$
12	$\chi \leftarrow fast - non - dominated - sorting(S);$
13	até ($\ P\ < T_{pop}$) E ($\ \chi\ \leq (T_{pop} - \ P\)$)
14	$crowding - distance - sorting(\chi);$
15	enquanto $\ P\ \leq T_{pop}$ faça
16	$P \leftarrow P \cup retornaMelhorElemento(\chi);$
17	fim
18	fim
19	retorna $P;$
20	fim

4.2 Representação genética

Na representação genética cada indivíduo armazenará uma solução viável para o problema de roteamento. Assim, um indivíduo será composto por um vetor de rotas que ligam a origem ao destino de cada requisição na rede. Caso a requisição seja rejeitada, nenhum caminho será armazenado para aloca-la. O esquema da figura 4.2 ilustra a representação genética dos indivíduos no algoritmo genético proposto.

A inexistência de soluções inviáveis neste algoritmo pode limitar o seu espaço de busca, entretanto, a eficiência do algoritmo foi privilegiada. O tratamento de soluções inviáveis elevaria consideravelmente o tempo de processamento e, para a aplicação em um contexto dinâmico de otimização a eficiência do algoritmo é de grande relevância.

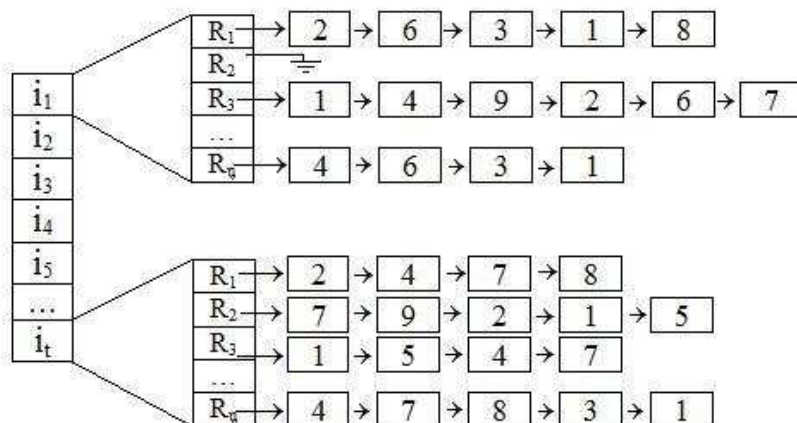


Figura 4.2. Representação genética do algoritmo

A figura 4.2 mostra como t indivíduos armazenam suas respectivas rotas. Cada indivíduo armazena um vetor de z requisições, sendo que cada uma possui uma lista de vértices que compõe a rota a ser seguida da sua origem ao destino. Quando não for possível atender uma dada requisição, esta armazenará um valor nulo indicando a inexistência da rota.

4.3 População inicial

Para gerar os indivíduos da população inicial o conceito de classes de serviço introduzido neste capítulo deve ser considerado. Caso exista diferenças de classes de serviço entre as requisições, aquelas pertencentes à classe A devem ser privilegiadas e atendidas antes das requisições das classes B e C.

Respeitando a restrição de classes de serviço, o algoritmo seleciona aleatoriamente uma requisição do conjunto R e sua rota é calculada na rede. Assim que uma requisição é atendida, a respectiva banda utilizada para atendê-la é retirada da rede, simulando o procedimento de alocação de rotas explícitas com reserva de banda. E quando não for possível atender uma requisição na rede, ela será considerada rejeitada.

Para determinar a rota que conecta a origem ao destino de cada requisição é utilizado o algoritmo de Dijkstra [1959] com diferentes atribuições de custos para os links da rede. Neste procedimento, a banda exigida pela requisição é retirada dos links da rede (supondo que aquela requisição poderá utilizar todos os links) e todos aqueles que continuaram com valor de banda residual positiva receberão um custo que será utilizado pelo algoritmo de Dijkstra no cálculo do caminho mínimo.

A atribuição de custos aos links também será feita em função das classes de serviço, caso estas sejam consideradas. Se a requisição a ser atendida pertence à classe B (que deve privilegiar o atraso no atendimento da requisição) cada aresta terá custo unitário e, portanto, o algoritmo de Dijkstra retornará o caminho com o menor número de saltos na rede. Caso a requisição seja das classes A ou C, ao custo de cada link será atribuído um valor entre 1 e 100, a ser escolhido aleatoriamente com o objetivo de aumentar a diversidade das soluções calculadas.

4.4 Cruzamento

A geração do conjunto Q no algoritmo NSGA-II é realizada através do cruzamento. São selecionados indivíduos do conjunto P aos pares e através do cruzamento são originados novos indivíduos que compõem o conjunto Q . A seleção dos indivíduos para o cruzamento baseia-se também no conceito de dominância apresentado na definição 1. É selecionado aleatoriamente um par de indivíduos e verifica-se a relação de dominância existente entre eles. Aquele indivíduo que domina o outro segue para o cruzamento, e caso não exista relação de dominância entre eles, ambos são cruzados. Este método de seleção é denominado *Torneio de Pareto*.

O processo de cruzamento procura agregar características dos indivíduos pais na geração do novo indivíduo filho. Suponha que os indivíduos i_1 e i_2 sejam os pais selecionados para o cruzamento. Ambos armazenam rotas viáveis para a requisição R_k , caso esta tenha sido atendida. Então, para compor a rota de R_k no indivíduo filho c_1 , o processo de cruzamento procura integrar o caminho de i_1 ao de i_2 gerando uma nova rota viável. A figura 4.3 ilustra o processo de cruzamento.

O cruzamento procura por um vértice v_c em comum nos caminhos de i_1 e i_2 para a requisição R_k , caso encontre este vértice será utilizado como intermediário para a geração do caminho do novo indivíduo. De o_k até v_c será herdado de i_1 e de v_c até d_k será obtido do indivíduo i_2 .

Caso não exista um vértice comum entre os caminhos de i_1 e i_2 , o cruzamento utilizará uma outra estratégia para a geração de um novo caminho. O algoritmo procurará

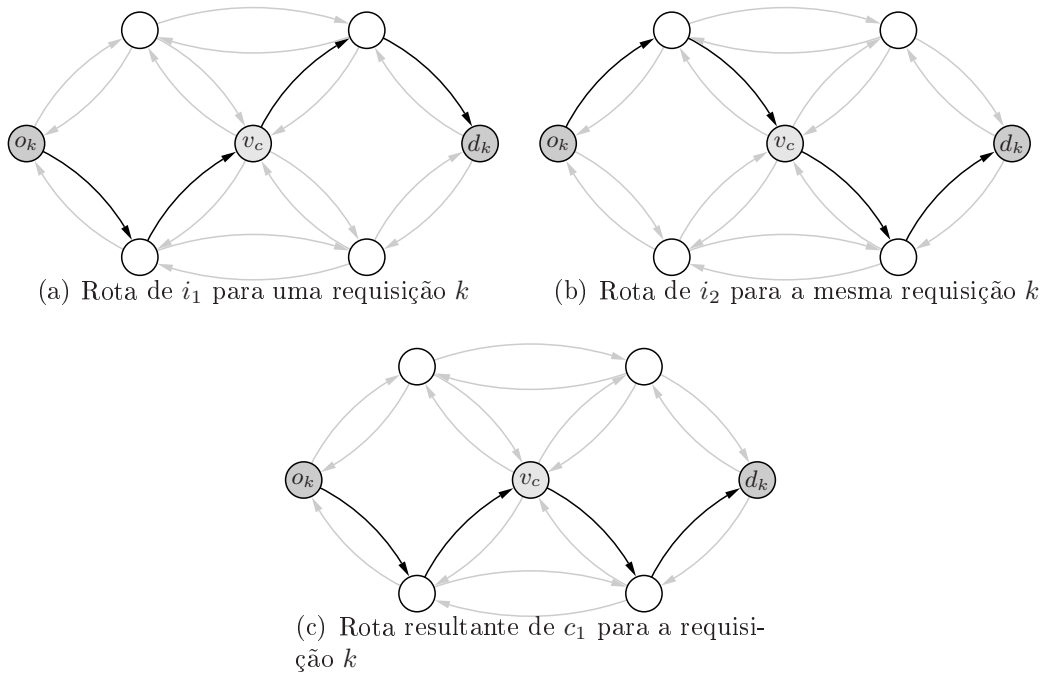


Figura 4.3. Processo de cruzamento no algoritmo genético multi-objetivo

por uma aresta a_c que conecte o caminho de i_1 a i_2 , ou vice-versa. Caso encontre, o caminho do novo indivíduo será composto pelas arestas de i_1 de o_k até a_c , pela própria aresta a_c e pelas arestas de i_2 até alcançar d_k . Este procedimento é mostrado na figura 4.4.

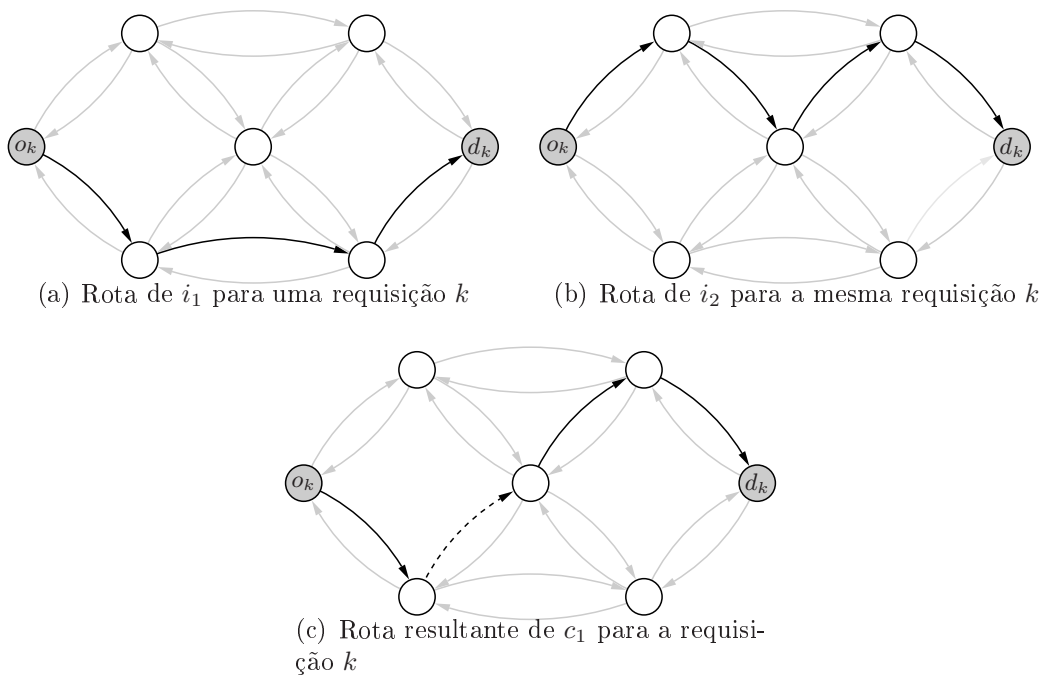


Figura 4.4. Procedimento alternativo para o cruzamento no algoritmo caso não sejam encontrados vértices em comum

Caso i_1 ou i_2 não apresentem uma rota para a requisição R_k , esta é considerada rejeitada. Assim, c_1 herdará completamente a rota daquele indivíduo que atende esta requisição. E se ambos os pais não atendem R_k , esta também será considerada rejeitada em c_1 .

4.5 Mutaç o

Na muta o uma pequena altera o   includida em um indiv duo a fim de mudar a sua posi o no espa o de busca. Neste algoritmo gen tico, a muta o consiste em desviar o fluxo de um link que estava sendo utilizado no roteamento das requisi es. Mais especificamente, o algoritmo procura pelo link com maior percentual de utiliza o na rede e procura por outros caminhos para as requisi es que o utilizam.

Ao definir uma caracter stica espec fica a ser privilegiada na muta o, como desviar o fluxo do link mais congestionado da rede,   poss vel que as solu es que sofram muta o apresentem um comportamento tendencioso, procurando por melhores solu es apenas com rela o a um dos objetivos. Esta opera o de muta o pode ser entendida como uma busca local com rela o ao objetivo de balanceamento de carga na rede.

A figura 4.5 ilustra o processo de muta o no algoritmo, em 4.5(a) observa-se que duas requisi es s o roteadas utilizando o link destacado em negrito que se encontra sobrecarregado. O resultado da muta o apresenta novas rotas para as requisi es, conforme pode ser visto em 4.5(b).

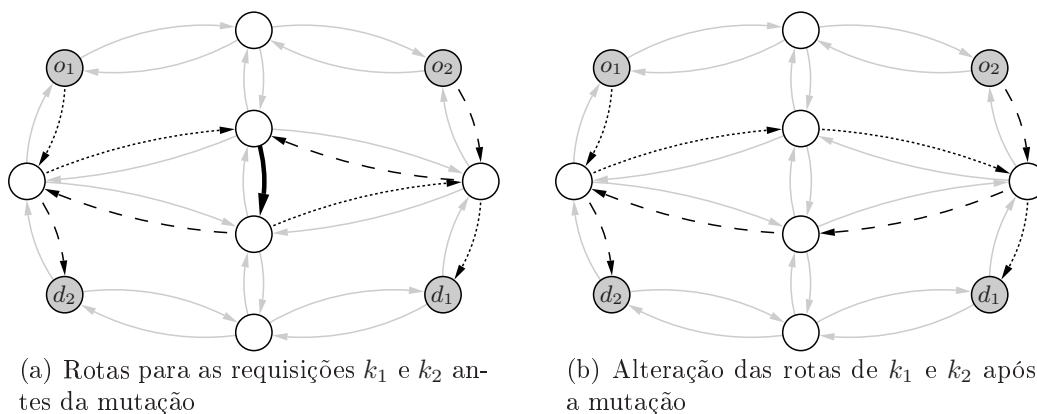


Figura 4.5. Exemplo de muta o no algoritmo

A taxa de muta o tamb m   um par metro a ser inserido pelo usu rio na execu o do algoritmo. Uma taxa de muta o elevada pode tornar inst vel o processo de busca por melhores solu es, enquanto uma taxa muito baixa pode dificultar o processo evolutivo do algoritmo.

4.6 Complexidade assintótica

Uma vez que o problema de roteamento em redes MPLS sujeito a mais de duas restrições de QoS é NP-Completo [Shao et al., 2006], esta heurística foi proposta com o intuito de realizar aproximações do conjunto Pareto-ótimo em tempo polinomial. Assim será possível utilizá-la em um contexto dinâmico de otimização, onde a eficiência das respostas é fator determinante no roteamento.

Para descrever a complexidade desta heurística, considera-se os parâmetros:

- o número de arestas (A) e o número de vértices (V) do grafo (que podem ser entendidos também como o número de links e roteadores da rede);
- a quantidade de requisições (R) a serem otimizadas simultaneamente;
- os parâmetros de configuração do algoritmo genético:
 - o número de gerações (G);
 - o tamanho da população (P);
 - a quantidade de funções objetivo do problema (M);

Ao longo do processo evolutivo do algoritmo genético, diferentes operações são realizadas para a obtenção das soluções. Cada uma destas operações contribui significativamente para a complexidade do algoritmo e por isto serão descritas separadamente, uma vez que a complexidade de uma independe da outra.

Para gerar a população inicial, cada indivíduo deverá armazenar rotas viáveis para satisfazer as R requisições na rede. As requisições têm exigência de banda a ser satisfeita e por outro lado os links têm limite de banda, portanto, assim que calcula-se o caminho para uma requisição deve-se atualizar o valor de banda residual nos links para que as próximas requisições não extrapolem este valor e haja perda de pacotes na rede.

Para calcular o caminho para uma requisição, são atribuídos pesos às arestas que estão em função da classe da requisição e então é utilizado o algoritmo de Dijkstra, cuja complexidade é dada por $O(A \log V)$ quando utiliza-se um heap binário em sua implementação. Para atualizar o grafo, no pior caso é possível visitar todas as arestas. Considerando que este procedimento deve ser realizado para todas as requisições, em todos os indivíduos da população, a complexidade assintótica para gerar a população inicial pode ser expressa como

$$O(PR(A + A \log V)).$$

No cruzamento, os indivíduos são escolhidos aos pares e para originar um herdeiro é necessário que para cada requisição o algoritmo percorra os caminhos armazenados nos indivíduos i_1 e i_2 afim de encontrar um vértice em comum que permita o cruzamento dos caminhos. O caminho de cada indivíduo para a requisição poderá ter no pior caso A arestas, portanto $2A$ operações.

Caso não existam vértices em comum entre os caminhos de i_1 e i_2 para a requisição, o algoritmo procurará por uma aresta que conecte os dois caminhos gerando um novo caminho para o indivíduo herdeiro. No pior caso, esta procura considerará todas as arestas do grafo, sendo necessárias A operações para concluí-la.

A procura por novos caminhos descrita acima é repetida para todas as requisições e originará novos P indivíduos herdeiros a cada geração do algoritmo. Portanto, a complexidade envolvida no cruzamento será dada no pior caso por

$$O(GPR(2A + A)) = O(GPR(3A)) = O(GPRA).$$

Para realizar a mutação é necessário inicialmente definir qual a aresta tem o maior percentual de utilização da rede, sendo necessárias A operações. Para eliminar a tráfego nesta aresta, são calculados novos caminhos para as requisições que não passem por ela, e novamente o algoritmo de Dijkstra é utilizado. Logo a complexidade assintótica da operação de mutação será

$$O(A + R(A \log V)) = O(R(A \log V)).$$

Por fim, conforme descrito na seção 4.1 a complexidade envolvida no NSGA-II para calcular a dominância dos indivíduos e definir quais passarão para a próxima geração é de $O(MP^2)$, sendo que M é o número de funções objetivo do problema que não se alterará neste problema, assumindo o valor constante 3. Como este cálculo será realizado a cada geração, tem-se que sua complexidade é

$$O(G(3P^2)) = O(GP^2).$$

Portanto, a complexidade assintótica do algoritmo será dada pela soma das complexidades envolvidas em cada etapa evolutiva sendo descrita como

$$O(PR(A + A \log V) + G(PRA + P^2) + R(A \log V)).$$

Como se pode observar, o desempenho computacional do algoritmo está em função de 4 parâmetros. Na próxima seção serão apresentados experimentos que verificarão seu comportamento em diferentes contextos de rede, atribuindo-se diferentes valores aos parâmetros.

4.7 Qualidade das aproximações e eficiência

O algoritmo genético multi-objetivo apresentado foi proposto com o intuito de obter aproximações do conjunto Pareto-ótimo para o problema de roteamento em redes MPLS. Quanto mais próximas estiverem as soluções do conjunto de Pareto, considera-se maior sua qualidade. No entanto, as aproximações devem ser obtidas eficientemente, pois no contexto dinâmico de otimização a eficiência das respostas é um fator de grande relevância.

Nesta seção, o desempenho da heurística será inicialmente avaliado ao comparar os resultados obtidos pelo algoritmo genético com aqueles obtidos ao solucionar o problema de forma exata, utilizando a formulação matemática proposta no capítulo 3 e o software de otimização CPLEX [Ilog Incorporation, 2006] versão 10.2. Posteriormente, será avaliada a influência dos parâmetros utilizados pela heurística, a saber o tamanho da população e o número de gerações, na qualidade das soluções e na eficiência do algoritmo.

Para realizar os experimentos, foram utilizadas as mesmas topologias de rede adotadas em Oliveira [2005], conforme pode ser visto na figura 4.6. Estas topologias permitem avaliar o comportamento do algoritmo em diferentes cenários, permitindo melhor validá-lo.

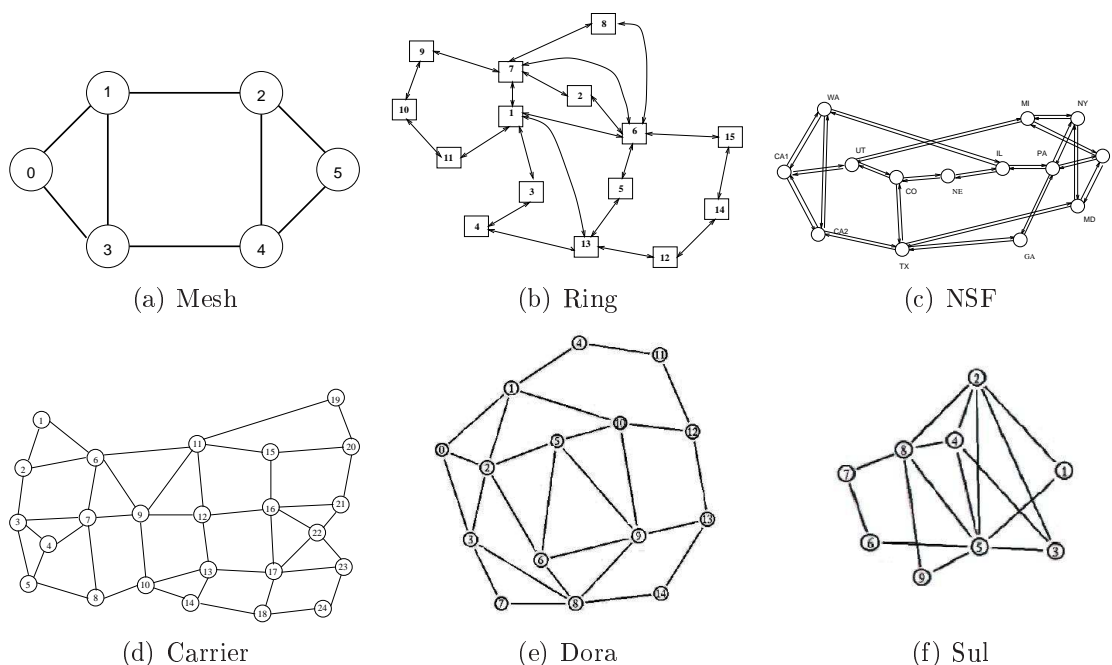


Figura 4.6. Topologias de rede utilizadas nos experimentos

As topologias Ring, NSF e Carrier representam cenários reais de redes, como redes metropolitanas ou *backbones* de Internet. Já as topologias Mesh, Dora e Sul são

utilizadas em trabalhos correlatos para propósitos de validação dos algoritmos e também serão adotadas neste trabalho. A tabela 4.1 apresenta algumas características das topologias utilizadas nos experimentos.

Topologia	Num. vértices	Num. links	Grau médio dos vértices
Mesh	6	16	2.67
Ring	15	42	2.8
NSF	14	38	2.71
Carrier	24	43	3.58
Dora	15	52	3.46
Sul	9	32	3.55

Tabela 4.1. Características das topologias utilizadas nos experimentos

Comparar os resultados de problemas de otimização multi-objetivo é definitivamente uma tarefa não-trivial. A complexidade envolvida para mensurar a qualidade de conjuntos de soluções é um desafio inexistente em problemas de otimização mono-objetivo onde a qualidade das soluções é definida verificando-se o valor mínimo (ou máximo) obtido.

Nesta seção serão adotadas duas métricas para mensurar a qualidade das aproximações obtidas pelo algoritmo genético multi-objetivo, a primeira delas será a solução utópica apresentada na Definição 3. Os valores apresentados por esta solução definem valores ótimos para as diferentes funções objetivo do problema, e servirá como referência para os valores obtidos pelo algoritmo genético.

Para obter a solução utópica, foi utilizado o software CPLEX [Ilog Incorporation, 2006] que solucionou a formulação matemática apresentada no capítulo 3 considerando-se uma função objetivo por vez. Assim, um único problema de roteamento multi-objetivo será transformado em três problemas mono-objetivo, cada um resultando em uma coordenada da solução utópica.

Na tabela 4.2 são exibidos os valores de solução utópica para o roteamento multi-objetivo considerando as topologias da figura 4.6. Quanto às requisições a serem otimizadas, seleciona-se aleatoriamente o par (origem, destino) e a respectiva banda necessária para sua alocação varia também aleatoriamente no intervalo de 200K a 400K. A capacidade de banda para é de 1024K para todos os links.

Para mensurar a qualidade das aproximações obtidas pelo algoritmo genético, será utilizada uma aproximação da solução utópica. Esta solução será representada por um ponto cujas coordenadas terão o melhor valor de cada função objetivo obtido ao se executar o algoritmo genético multi-objetivo. Os resultados das aproximações são mostrados na tabela 4.3 para o mesmo conjunto de topologias e requisições considerados para o cálculo da solução utópica, na tabela 4.2.

topologia	solução utópica (<i>CPLEX</i>)								
	10 requisições			30 requisições			50 requisições		
	obj F_1	obj F_2	obj F_3	obj F_1	obj F_2	obj F_3	obj F_1	obj F_2	obj F_3
Mesh	8	2	0.84	8	11	0.85	11	19	0.82
Ring	15	2	0.81	23	6	0.89	21	16	0.86
NSF	15	1	0.86	19	8	0.88	20	15	0.85
Carrier	19	1	0.88	40	2	0.87	41	9	0.85
Dora	17	0	0.72	30	4	0.73	44	8	0.74
Sul	6	5	0.78	15	8	0.83	17	17	0.85

Tabela 4.2. Solução utópica para problemas de roteamento multi-objetivo

topologia	aproximação da solução utópica								
	10 requisições			30 requisições			50 requisições		
	obj F_1	obj F_2	obj F_3	obj F_1	obj F_2	obj F_3	obj F_1	obj F_2	obj F_3
Mesh	8	2	0.84	9	11	0.85	12	19	0.84
Ring	17	2	0.81	23	6	0.89	22	16	0.86
NSF	16	1	0.86	23	8	0.92	23	15	0.88
Carrier	20	1	0.88	49	2	0.93	52	9	0.88
Dora	17	0	0.72	32	4	0.73	46	8	0.77
Sul	7	5	0.78	17	8	0.83	17	17	0.85

Tabela 4.3. Aproximação das soluções utópicas através do algoritmo genético

Ao comparar os resultados das tabelas 4.2 e 4.3 é possível observar que as aproximações da solução utópica obtidas pelo algoritmo genético estão próximas das soluções calculadas na otimalidade, inclusive atingindo o valor ótimo em algumas instâncias. Mesmo com o aumento do número de requisições, as aproximações obtidas pelo algoritmo genético apresentam uma boa qualidade, devido à proximidade das soluções exatas.

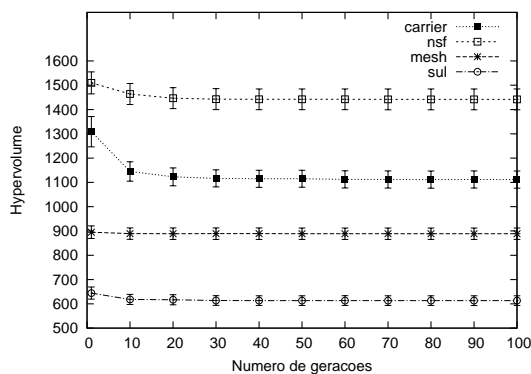
A segunda estratégia para mensurar a qualidade das aproximações foi a utilização do *hypervolume*. Esta medida consiste em avaliar a qualidade das soluções calculando o volume entre um ponto de referência e os pontos do conjunto de soluções aproximadas. Como o problema multi-objetivo apresentado se trata de um problema de minimização, o ponto de referência será a origem $(0, 0, 0)$ e quanto menores os valores do *hypervolume*, maior a qualidade das soluções.

Diferentes algoritmos foram propostos para calcular o *hypervolume*. Neste trabalho foi implementado o algoritmo de Inclusão-Exclusão [Wu e Azarm, 2001]. Dado um conjunto de pontos, este algoritmo consiste em avaliar todos seus subconjuntos, somando ao valor aqueles que contribuem com o *hypervolume*, e subtraindo os valores que foram somados como intercessão. O algoritmo de Inclusão-Exclusão é descrito a seguir:

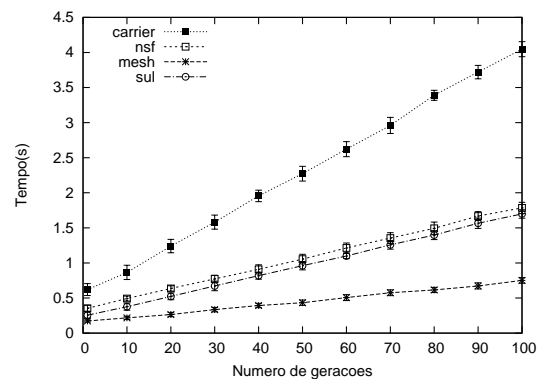
Algoritmo 2: Hypervolume por Inclusão-Exclusão	
Entrada: S	
Saída: $volume$	
1	início
2	$volume = 0$
3	para Cada Subconjunto S' não-vazio de S faça
4	$volume = volume + intercessao(S') * (-1)^{\ S'\ +1}$
5	fim
6	retorna $volume$
	<i>/* intercessao(S') retorna o volume do maior hyper-cubóide dominado pelos pontos de S' */</i>
7	fim

Assim, o *hypervolume* será utilizado para mensurar o desempenho do algoritmo genético considerando seus dois principais parâmetros: o tamanho da população e o número de gerações. Quanto mais indivíduos em uma população, maior a possibilidade de se obter soluções de qualidade. E também à medida em que os indivíduos da população evoluem de geração, a qualidade das soluções tende a aumentar. Entretanto, ambos têm como consequência o aumento no tempo de execução do algoritmo.

O experimento a seguir avalia o comportamento do algoritmo genético quanto ao parâmetro número de gerações. Os resultados são ilustrados na figura 4.7, onde são avaliadas a melhoria na qualidade das aproximações 4.7(a) e também a eficiência computacional do algoritmo 4.7(b). Neste cenário são avaliadas as topologias Carrier, NSF, Mesh e Sul contendo 50 requisições definidas aleatoriamente. A população tem tamanho fixo de 50 indivíduos e a taxa de mutação é 0.02 para todas as instâncias do problema.



(a) Variação do volume com relação ao número de gerações do algoritmo



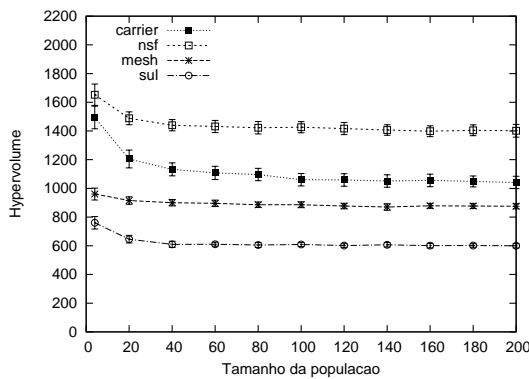
(b) Crescimento do tempo computacional com o aumento no número de gerações

Figura 4.7. Influência do parâmetro número de gerações no algoritmo genético

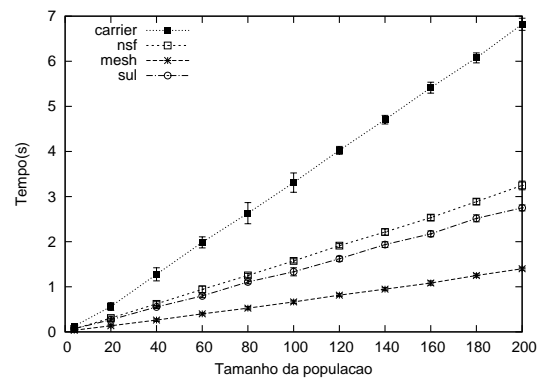
Observa-se que a influência do número de gerações na qualidade das aproximações é significativa apenas nas primeiras gerações do algoritmo, sendo que quando o número de gerações se torna elevado, as soluções tendem a não apresentar melhorias de qualidade. Este comportamento sugere que a operação de mutação pode ser melhorada, para que consiga retirar as soluções de mínimos locais e fazer com que o algoritmo genético explore melhor o espaço de busca das soluções.

Por outro lado, o tempo de execução cresce linearmente com o aumento no número de gerações. Este comportamento era esperado considerando-se a complexidade assintótica do algoritmo, uma vez que o número de gerações é um parâmetro que tem comportamento linear. Os resultados da figura 4.7 sugerem que este parâmetro não deve assumir valores elevados, pois a relação de melhoria da qualidade por tempo se apresenta melhor com um número de gerações reduzido.

Considerando agora o desempenho do algoritmo em função do tamanho da população, ao utilizar um número fixo de 30 gerações e uma taxa de mutação de 0.02 os resultados são exibidos nos gráficos da figura 4.8. A influência do tamanho da população no valor do *hypervolume* é mais acentuada, conforme pode ser visto no gráfico 4.8(a). Já a eficiência do algoritmo considerando o tamanho da população é apresentada no gráfico 4.8(b).



(a) Valor do *hypervolume* ao se variar o tamanho da população



(b) Crescimento do tempo computacional em função do tamanho da população

Figura 4.8. Influência do parâmetro tamanho da população no algoritmo genético

A maior quantidade de indivíduos na população faz com que o espaço de busca das soluções seja melhor explorado e, conseqüentemente, o algoritmo armazene aquelas mais próximas ao conjunto Pareto-ótimo. Mas ainda assim, a melhoria da qualidade tende a diminuir quando o número de indivíduos se torna muito elevado. Para a instância que utiliza a topologia mesh, observa-se que a melhoria é pouco significativa, mas este fato pode ser justificado pela existência de poucas opções de roteamento, uma

vez que esta topologia tem menos links e nós roteadores entre todas.

O crescimento do tempo de execução do algoritmo é linear e a justificativa para este comportamento pode ser dada ao observar sua complexidade assintótica descrita na seção 4.6

$$O(PR(A + A \log V) + G(PRA + P^2) + R(A \log V)).$$

O parâmetro P que denota o tamanho da população terá um comportamento linear enquanto $RA > P$. Isto significa que se a desigualdade é satisfeita e o comportamento deste parâmetro será determinado por $O(GPRA)$, caso o tamanho da população seja maior que o número de requisições R multiplicado pelo número de arestas do grafo A , a influência do parâmetro P terá um comportamento quadrático, determinado por $O(GP^2)$.

Como em todas as instâncias deste experimento a desigualdade $RA > P$ é satisfeita, o crescimento do tempo computacional do algoritmo em função do tamanho da população apresenta um comportamento linear, como pode ser visto no gráfico 4.8(b).

Os experimentos mostram que o algoritmo genético proposto obtém boas aproximações para o problema de roteamento em redes MPLS. A proximidade da solução utópica para o problema além da eficiência computacional do algoritmo sugerem que sua aplicação em um contexto dinâmico de otimização será bem sucedida. Portanto, o próximo capítulo mostra os resultados obtidos utilizando-se o algoritmo apresentado como alternativa para determinar as rotas LSP em redes MPLS dinamicamente, usando para isto cenários de simulação.

Capítulo 5

Resultados de simulações

Este capítulo se dedica à experimentação do algoritmo de otimização multi-objetivo proposto neste trabalho aplicado ao problema de roteamento em redes MPLS em um contexto dinâmico de otimização. Assim será possível avaliar o desempenho do algoritmo ao definir rotas LSP dinamicamente na rede.

Para isto é utilizado o ambiente de simulação NS-2 (*Network Simulator 2*). Assim, será possível observar o comportamento do algoritmo considerando diferentes indicadores de QoS na rede. Particularmente, serão considerados como indicadores de QoS o atraso médio das requisições na rede, o balanceamento de carga e a disponibilidade do serviço.

O atraso medirá o tempo médio em que um pacote demora para sair da origem e alcançar o destino. Nos cenários de simulação utilizados o atraso para atravessar todos os links da rede têm o mesmo valor, sendo assim o número de links utilizados para atender uma requisição é um dos principais fatores que influenciam o atraso.

Já o balanceamento de carga será avaliado através do percentual de carga no link mais utilizado da rede. Se o tráfego de uma rede está concentrado em um ou mais links, eles tenderão a apresentar maiores dificuldades de transmissão dos dados o que pode ocasionar problemas na comunicação na rede.

A disponibilidade do serviço será medida calculando-se quantas requisições surgiram mas não puderam ser atendidas. Entre os motivos da rejeição pode-se citar a indisponibilidade de banda para alocar a rota LSP ou mesmo opção de roteamento, quando o otimizador decide rejeitar uma requisição para priorizar os valores de QoS de outras na rede.

Um grande desafio encontrado na alocação dinâmica das rotas LSP na rede foi o processo de tomada de decisão. A execução do algoritmo genético multi-objetivo resulta em uma aproximação para o conjunto de Pareto, portanto, um conjunto de soluções para o problema. A escolha de qual solução é a mais adequada para o roteamento é

uma decisão difícil de ser tomada.

Para o processo de decisão foi utilizado o método AHP (*Analytic Hierarchy Process*) [Saaty, 1980]. Neste método o decisor define a relação de prioridade existente entre as funções objetivo do problema, obtendo uma matriz de preferências. Em um problema contendo n funções objetivo, a matriz de preferências terá dimensões $n \times n$. Suponha um problema com 3 funções objetivo, sua matriz de preferências seria dada como

$$\begin{array}{c|ccc} & f_1 & f_2 & f_3 \\ \hline f_1 & 1 & a & b \\ f_2 & \frac{1}{a} & 1 & c \\ f_3 & \frac{1}{b} & \frac{1}{c} & 1 \end{array}$$

onde os valores a , b e c representam a prioridade da matriz f_1 sobre f_2 , f_1 sobre f_3 e f_2 sobre f_3 , respectivamente. A diagonal principal é 1, pois não é possível definir a prioridade de uma função objetivo sobre ela mesma.

O AHP utiliza então operações vetoriais e matriciais através de vetores denominados *eigen vectors* para definir valores que permitam classificar cada solução do problema multi-objetivo com relação às prioridades atribuídas. Assim, é escolhida aquela solução que apresente melhores valores para as funções objetivo, segundo as prioridades da matriz de preferências.

As topologias de rede utilizadas para compor os cenários de simulação são aquelas mesmas apresentadas no capítulo 4, figura 4.6. Para mensurar o desempenho da estratégia multi-objetivo no roteamento dinâmico, diferentes cenários de simulação foram utilizados. São propostos 5 cenários que objetivam avaliar o desempenho da estratégia multi-objetivo proposta sob diferentes condições.

Os cenários de simulação 1 e 2 comparam o desempenho das abordagens de otimização mono-objetivo e multi-objetivo no problema de roteamento dinâmico, sendo que no cenário de simulação 2 as requisições são geradas para realizar comunicação em *broadcast* na rede. No cenário de simulação 3 é verificada a flexibilidade do roteamento através da abordagem multi-objetivo ao utilizar diferentes matrizes de preferências para realizar o roteamento, obtendo diferentes resultados sem alterar o procedimento de otimização. O cenário de simulação 4 compara as abordagens de roteamento estática e dinâmica, enquanto no último cenário a estratégia de diferenciação de classes é avaliada.

5.1 Cenário de simulação - 1

O primeiro cenário de simulação avalia o desempenho da técnica de otimização multi-objetivo comparando-a com duas técnicas tradicionalmente utilizadas para o roteamento em redes MPLS. A primeira delas é a estratégia de caminho mínimo, onde as requisições seguirão o menor caminho com banda disponível da origem ao destino. Já a segunda técnica de roteamento aloca as rotas LSP procurando maximizar o balanceamento de carga na rede, o que pode ser entendido também como minimizar o percentual de carga no link mais utilizado da rede.

Na primeira simulação é considerada a topologia de rede NSF, onde as requisições surgem aleatoriamente seguindo uma distribuição de Poisson, e têm sua permanência na rede definidas por uma distribuição Normal. As três técnicas de otimização são utilizadas para alocar as requisições na rede assim que elas surgem. A tomada de decisão utilizando a estratégia multi-objetivo será feita utilizando uma matriz de preferências do AHP que privilegia soluções sem rejeição e que busquem um equilíbrio entre o percentual de carga no link mais utilizado da rede e o número de links para atender as requisições.

Os indicadores de QoS avaliados são o atraso médio das requisições e o balanceamento de carga na rede e são exibidos na figura 5.1 variando-se o parâmetro λ da distribuição de Poisson e fixando-se os valores de média e variância para a distribuição Normal como 50 e 5 segundos respectivamente. O tempo de simulação foi definido como 120 segundos, pois nos experimentos se mostrou um tempo suficiente para verificar o comportamento das estratégias de otimização.

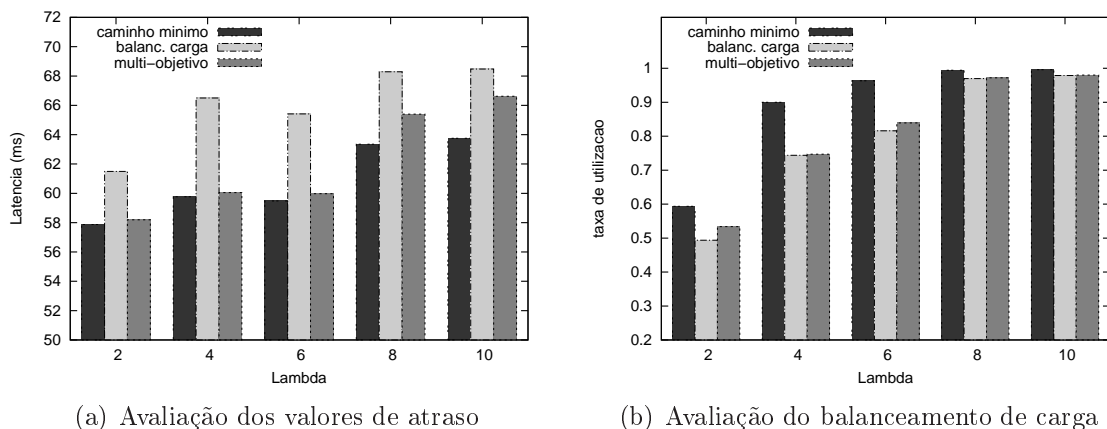


Figura 5.1. Comparação da abordagem de otimização multi-objetivo com as estratégias de caminho mínimo e balanceamento de carga, considerando a topologia NSF

Conforme mostram os gráficos 5.1(a) e 5.1(b), ao utilizar a estratégia mono-objetivo de caminho mínimo os resultados com relação ao atraso médio das requisições são mínimos, no entanto, o balanceamento de carga da rede não é privilegiado. Por outro lado, ao definir as rotas LSP em função apenas do balanceamento de carga na rede, os valores de atraso apresentados são elevados. Isto acontece porque ambos se tratam de técnicas de otimização mono-objetivo, que alocam as rotas privilegiando apenas um objetivo específico.

Ao avaliar a técnica de otimização multi-objetivo, observa-se que os valores obtidos para os dois indicadores de QoS avaliados se aproximam dos valores mínimos obtidos. E isto consiste na principal vantagem apresentada pela estratégia multi-objetivo neste cenário.

Para verificar o comportamento das diferentes técnicas de otimização utilizadas anteriormente em diferentes topologias, a mesma simulação foi realizada novamente considerando agora todas seis topologias da figura 4.6. Os parâmetros utilizados são os mesmos, exceto o λ da distribuição de Poisson que agora assume valor fixo 4. Os resultados podem ser vistos na tabela 5.1.

topologia	Caminho mínimo		Balanc. de Carga		Multi-objetivo	
	atraso	% de carga	atraso	% de carga	atraso	% de carga
Mesh	52.2ms	70%	56.5ms	65%	52.2ms	69%
Ring	69.1ms	74%	69.3ms	71%	69.1ms	71%
NSF	59.7ms	90%	66.5ms	74%	60.0ms	75%
Carrier	65.1ms	62%	66.3ms	48%	66.0ms	53%
Dora	54.6ms	50%	56.0ms	46%	54.7ms	46%
Sul	56.7ms	93%	58.4ms	67%	56.9ms	78%
Média	59.6ms	73%	62.2ms	62%	59.8ms	65%

Tabela 5.1. Indicadores de QoS nas redes utilizando diferentes estratégias de roteamento

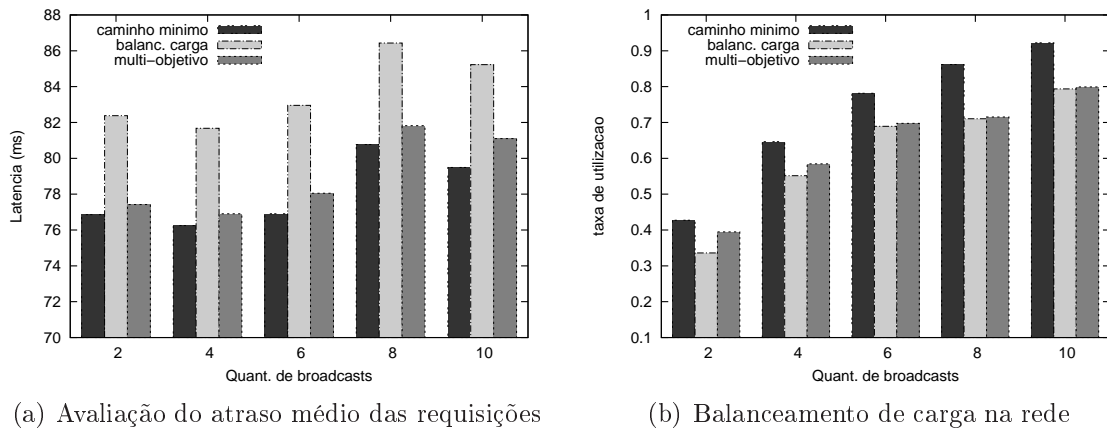
Os valores apresentados pela tabela 5.1 confirmam o comportamento apresentado pela otimização multi-objetivo também para as demais topologias. Observa-se uma tendência que os dois indicadores de QoS apresentem valores próximos dos ótimos quando utilizada a estratégia multi-objetivo para o roteamento, embora nas topologias Ring e Sul a diferença entre os valores de todas as estratégias estejam muito próximas.

Acredita-se que em topologias de rede contendo um número maior de roteadores e links esta diferença seja ainda mais visível, quando a diferença entre os valores de QoS entre as estratégias de caminho mínimo e balanceamento de carga sejam maiores, e a otimização multi-objetivo por sua vez apresentará resultados ainda mais expressivos, com valores aproximando-se dos ótimos para os diferentes indicadores de QoS.

5.2 Cenário de simulação - 2

O segundo cenário de simulação é muito semelhante ao primeiro, apresentando as mesmas características exceto o comportamento das requisições. Neste cenário, as requisições surgem para atender a uma comunicação em *broadcast* na rede e têm uma exigência de banda menor do que aquelas do primeiro cenário de simulação. Assim, será possível comparar a estratégia de otimização multi-objetivo com as mono-objetivo em um outro contexto de rede.

Em uma rede com n roteadores, um *broadcast* consistirá em $n - 1$ requisições a serem alocadas simultaneamente. À medida em que os *broadcasts* acontecem, as requisições devem ser alocadas utilizando as estratégias de otimização de caminho mínimo, balanceamento de carga e multi-objetivo. Os gráficos da figura 5.2 mostram os valores obtidos para os indicadores de QoS atraso das requisições e percentual de carga no link mais utilizado, quando o número de *broadcasts* varia, considerando a topologia Carrier.



(a) Avaliação do atraso médio das requisições

(b) Balanceamento de carga na rede

Figura 5.2. Diferentes abordagens de otimização no roteamento de requisições de broadcast, considerando a topologia Carrier

As soluções obtidas pelas diferentes estratégias de roteamento para requisições de *broadcast* apresentam um comportamento semelhante àquele observado quando as requisições surgem aleatoriamente na rede. Isto sugere que a estratégia de otimização multi-objetivo pode se desempenhar bem em diferentes contextos de otimização, variando-se o comportamento das topologias e também das requisições.

A próxima tabela ilustra os valores obtidos para os indicadores de QoS atraso das requisições e balanceamento de carga para todas as topologias, considerando requisições de *broadcast*. Durante o período de simulação 4 *broadcasts* são realizados na rede, e a exigência de banda para a alocação das rotas LSP é baixa, de modo que não exista rejeição e todas as requisições sejam atendidas.

topologia	Caminho mínimo		Balanc. de Carga		Multi-objetivo	
	atraso	% de carga	atraso	% de carga	atraso	% de carga
Mesh	49.4ms	28%	53.3ms	20%	50.5ms	24%
Ring	66.1ms	81%	69.1ms	58%	66.1ms	76%
NSF	63.7ms	52%	67.3ms	46%	65.2ms	47%
Carrier	76.2ms	64%	81.6ms	55%	76.9ms	58%
Dora	58.0ms	44%	58.8ms	44%	58.4ms	44%
Sul	51.1ms	38%	53.0ms	36%	51.4ms	37%
Média	60.7ms	52%	63.9ms	43%	61.4ms	47%

Tabela 5.2. Indicadores de QoS nas redes utilizando diferentes estratégias de roteamento para atender requisições de broadcast

Os resultados apresentados na tabela 5.2 confirmam as expectativas e mostram que a estratégia de otimização multi-objetivo apresenta um bom desempenho também para as demais topologias quando realiza-se *broadcasts* na rede. Assim confirma-se a eficiência desta abordagem na otimização simultânea de diferentes indicadores de QoS.

5.3 Cenário de simulação - 3

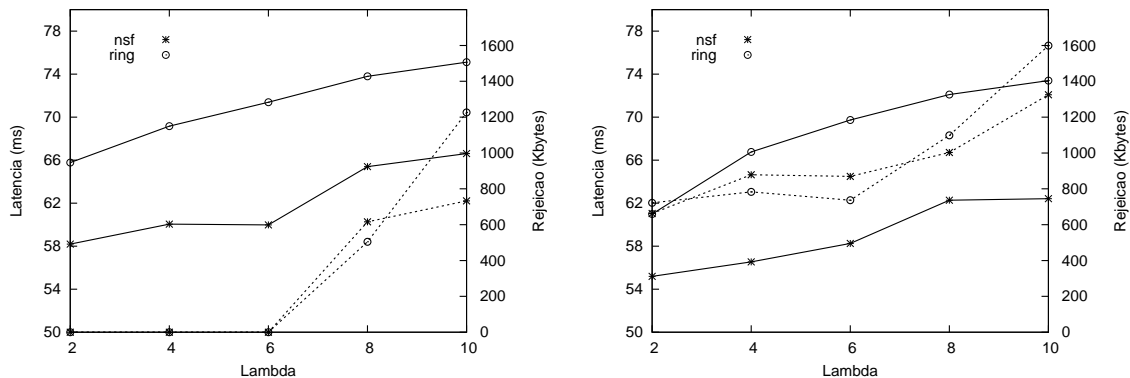
Além da possibilidade de considerar diferentes indicadores de QoS simultaneamente no processo de otimização, a estratégia de roteamento multi-objetivo apresenta também como vantagem a flexibilidade. A solução do problema de roteamento multi-objetivo fornece como resultado um conjunto de soluções viáveis e a escolha da solução mais adequada será definida no processo de tomada de decisão.

É possível privilegiar as soluções quanto ao atraso, ao balanceamento de carga ou mesmo à quantidade de rejeições na rede apenas escolhendo-se soluções adequadas do conjunto. Ao utilizar o método AHP, é possível alterar os valores de prioridade das funções objetivo na matriz de preferências e mudar o esquema de roteamento das rotas LSP, sem a necessidade de formular outro problema de otimização.

Neste contexto, o próximo cenário de simulação foi formulado para demonstrar a flexibilidade da otimização multi-objetivo no roteamento dinâmico em redes MPLS. O mesmo procedimento de otimização será utilizado neste cenário considerando duas matrizes de preferências distintas para o processo de tomada de decisão. A figura 5.3 apresenta os resultados obtidos, sendo que o gráfico 5.3(a) utiliza uma matriz de preferências que privilegia a não-rejeição de requisições. Já o gráfico 5.3(b) utiliza uma matriz de preferências que privilegia o atraso, mesmo que seja necessário rejeitar requisições.

Para a interpretação dos gráficos é importante ressaltar que o eixo Y apresenta diferentes escalas. A linha tracejada diz respeito à quantidade de Kbytes rejeitados

no roteamento e está relacionada ao eixo Y da direita, enquanto a linha contínua está relacionada ao atraso médio na rede e seus valores devem ser verificados no eixo Y à esquerda. Quanto aos parâmetros da simulação, as requisições surgem aleatoriamente seguindo uma distribuição de Poisson, e sua duração é definida por uma distribuição Normal, com valor de média 50 segundos e variância 5. As topologias adotadas foram NSF e Ring.



(a) Resultados utilizando uma matriz de preferências que evita rejeições

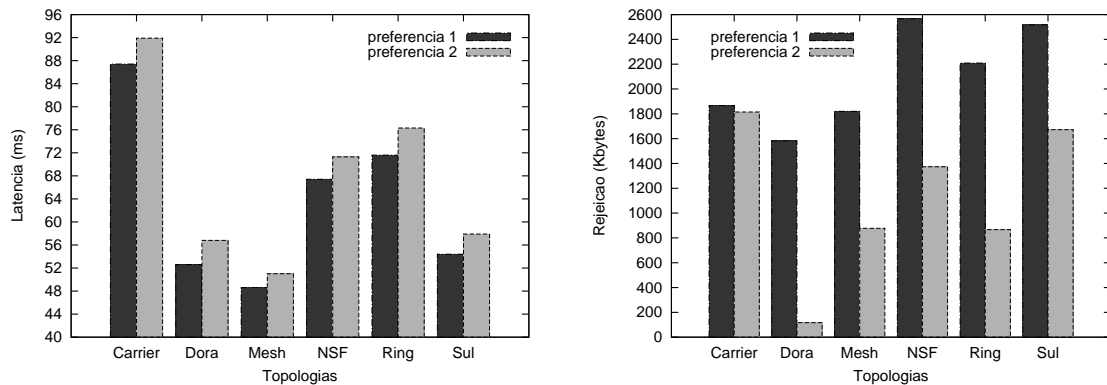
(b) Matriz de preferência que privilegia o atraso, mesmo que seja necessário rejeitar requisições

Figura 5.3. Flexibilidade da otimização multi-objetivo no problema de roteamento em redes MPLS

A comparação dos resultados ilustrados nos gráficos 5.3(a) e 5.3(b) permite concluir que o procedimento de otimização multi-objetivo fornece diferentes soluções ao mesmo problema de roteamento, apenas alterando a prioridade entre as funções objetivo. A diferença nos valores nos valores de atraso aproxima-se dos 10%, enquanto os valores de rejeição apresentam uma diferença ainda maior.

O próximo gráfico exibe os resultados das simulações feitas utilizando as mesmas duas matrizes de preferências, mas considerando agora todas as topologias de rede. As requisições têm o mesmo comportamento, no entanto, a distribuição de Poisson agora tem seu parâmetro fixo, $\lambda = 4$. O rótulo ‘preferencia 1’ identifica a utilização da matriz de preferências que prioriza baixos valores de atraso na rede, enquanto o rótulo ‘preferencia 2’ diz respeito à matriz de preferências que evita rejeições. Os resultados podem ser vistos na figura 5.4.

A melhoria obtida nos valores de atraso em contraposição à quantidade de rejeições observada nos gráficos 5.4(a) e 5.4(b) pode ser útil em alguns contextos de rede, por exemplo em aplicações de video-conferência ou VoIP. Diversas outras particularidades podem ser observadas em outros contextos de rede, e a otimização multi-objetivo proporciona flexibilidade para lidar com elas, sendo necessário apenas alguns ajustes no modelo de decisão para que novos resultados sejam obtidos.



(a) Valores de atraso para a otimização utilizando diferentes matrizes de preferência (b) Quantidade de Kbytes rejeitados, para realizar o roteamento na rede

Figura 5.4. Variação dos valores de atraso e balanceamento de carga nas redes, ao se utilizar diferentes matrizes de preferência

5.4 Cenário de simulação - 4

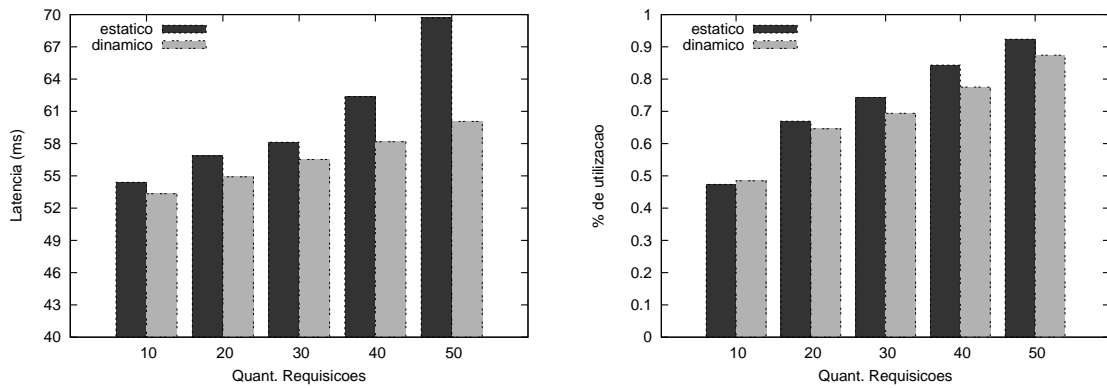
Este cenário de simulação tem a finalidade de realizar uma comparação entre as abordagens estática e dinâmica para o roteamento em redes MPLS através da otimização multi-objetivo. No roteamento estático assume-se que todas as requisições a serem alocadas são previamente conhecidas e então procura-se por rotas LSP que atendam as requisições durante todo o tempo de atividade da rede. Já no roteamento dinâmico as requisições surgem em função do tempo e consomem os recursos da rede apenas enquanto estão ativas.

Nesta simulação é definido um grupo de requisições que devem ser otimizadas estaticamente, ou seja, antes dos dados começarem a trafegar na rede. E para comparar com a abordagem dinâmica, as requisições deste mesmo grupo surgirão durante o período de simulação para serem alocadas na rede seguindo uma distribuição de Poisson, até que o número total seja atingido.

A figura 5.5 contém dois gráficos com os resultados comparativos entre a estratégia estática e a dinâmica considerando a topologia Dora. No gráfico 5.5(a) estão os valores de atraso médio na rede ao se utilizar ambas estratégias enquanto em 5.5(b) são mostrados os valores de balanceamento de carga.

Na otimização estática existe a vantagem de se conhecer previamente todas as requisições, sendo possível fazer um planejamento global mais eficiente, no entanto, deve-se reservar uma rota LSP para cada requisição desde o início de operação da rede até o fim, mesmo que a requisição não esteja ativa por todo este tempo, havendo um desperdício de recursos.

Já a otimização dinâmica não consegue visualizar todas as requisições de uma só



(a) Comparação dos valores de atraso entre as estratégias (b) Valores de balanceamento de carga, através do percentual de utilização do link mais saturado da rede

Figura 5.5. Comparação dos resultados obtidos pelas estratégias de roteamento estática e dinâmica

vez, pois estas surgem em função do tempo, porém a rota LSP é alocada apenas quando a requisição está ativa, e seus recursos são liberados quando a requisição termina. Esta característica torna o roteamento dinâmico mais adequado às necessidades de operação das redes.

5.5 Cenário de simulação - 5

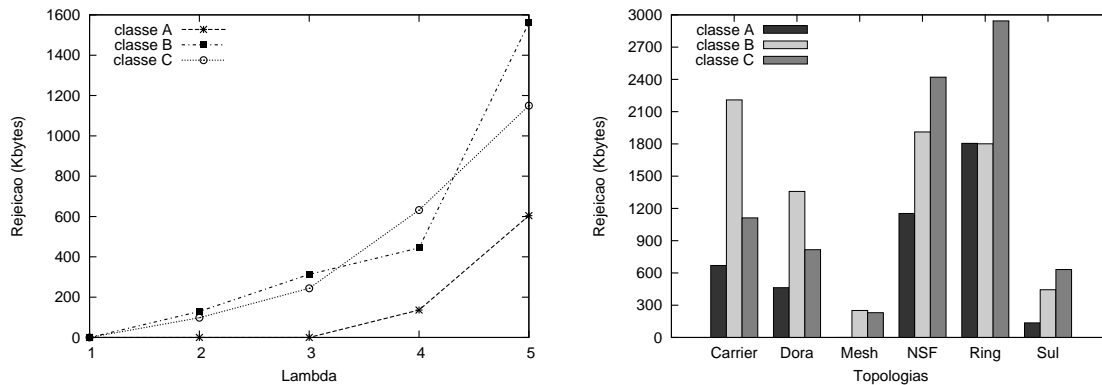
O tráfego de informação nas redes é gerado por diferentes tipos de aplicações. É possível realizar atividades desde o compartilhamento de arquivos até a realização de *streaming* de audio e video. Assim, as aplicações que utilizam seus recursos têm um caráter muito heterogêneo: algumas precisam utilizar os recursos com uma prioridade elevada enquanto outras têm uma demanda menos restrita.

Neste contexto, foi proposto neste trabalho uma abordagem para agrupar as requisições em classes e tratá-las de maneira diferente, procurando adequar o desempenho das requisições de uma determinada classe às suas necessidades. Esta abordagem foi descrita no capítulo 4 e consiste na definição das 3 classes de serviço: A, B e C.

Para avaliar o desempenho da abordagem de diferenciação de classes no roteamento dos LSPs, será atribuída aleatoriamente à cada requisição que surgir na rede uma classe de serviço, assim o procedimento de otimização multi-objetivo procurará atribuir rotas LSP às requisições que mais se adequem às necessidades de sua classe.

Os gráficos da figura 5.6 contêm os valores obtidos para a rejeição das requisições. O gráfico 5.6(a) mostra os valores de rejeição para as respectivas classes considerando a topologia Sul. As requisições surgem seguindo uma distribuição de Poisson, e o valor de λ varia de 1 a 5. Avaliando-se todas as topologias, para um valor de $\lambda = 4$, os

resultados são expressos no gráfico 5.6(b) .



(a) Rejeição das requisições para a topologia Sul (b) Avaliação da rejeição considerando-se todas as topologias

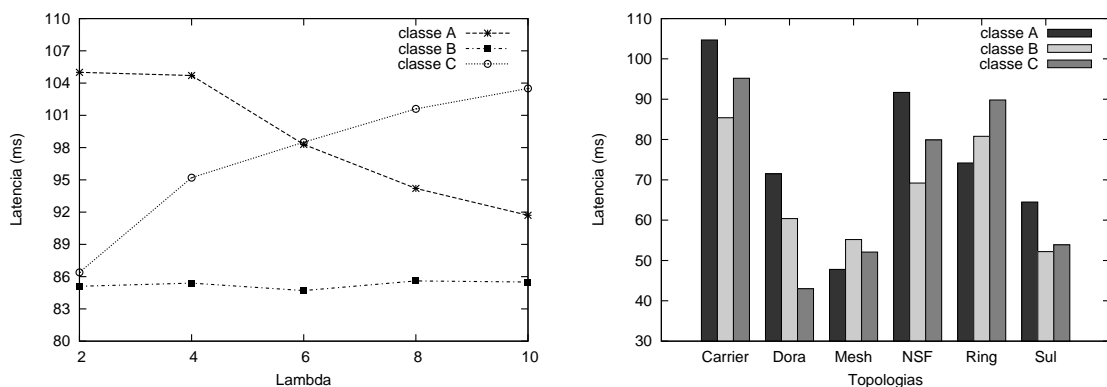
Figura 5.6. Comparação dos valores de rejeição para as diferentes classes de serviço

Os resultados apresentados pelos gráficos mostram que a quantidade de rejeição das requisições da classe A é menor quando comparado com as classes B e C. Uma vez que o algoritmo genético aloca primeiramente as requisições da classe A, elas serão rejeitadas apenas se não houverem recursos disponíveis para alocá-las. As requisições das classes B e C apresentam um índice de rejeição maior por serem alocadas posteriormente na rede, estando mais sujeitas às limitações dos recursos.

Analisando agora o valor de atraso observado na rede para as requisições pertencentes às diferentes classes, os resultados podem ser vistos na figura 5.7. A otimização dinâmica multi-objetivo com diferenciação de classes foi utilizada para o roteamento na topologia Carrier, e o aparecimento das requisições é definido através de uma distribuição de Poisson, com o λ variando em duas unidades de 2 a 10. Os resultados são exibidos no gráfico 5.7(a). Já o gráfico 5.7(b) ilustra o desempenho da abordagem de diferenciação de classes para todas as topologias, considerando o parâmetro da distribuição de Poisson $\lambda = 4$.

Ao avaliar o atraso médio obtido para as requisições da classe B, observa-se que os valores foram em sua maioria melhores do que aqueles apresentados pelas classes A e C. O gráfico 5.7(a) mostra uma oscilação dos valores de atraso para as requisições da classe A e C, enquanto os valores para a classe B permanecem menores em todas as simulações.

Ao verificar o comportamento nas demais topologias, os valores de atraso obtidos para a classe B mostram um bom desempenho da abordagem, no entanto, o atraso obtido para esta classe não é menor para todas as topologias. Nas topologias Ring e Dora, os valores de atraso para as classes A e C, respectivamente, são menores do que



(a) Variação do atraso em função do λ considerando a topologia Carrier (b) Atraso médio observada nas diferentes topologias para as classes A, B e C

Figura 5.7. Avaliação da abordagem de diferenciação de classes, quanto ao parâmetro de QoS atraso das requisições

aqueles obtidos para a classe B. E ainda, ao considerar a topologia Mesh verifica-se que o valor de atraso obtido para as requisições da classe B é pior que o de ambas as classes A e C.

Uma justificativa para este comportamento é que as requisições da classe A são alocadas primeiro, pois não devem ser rejeitadas, e isto pode influenciar no cálculo das rotas para as requisições da classe B, que são calculadas posteriormente. Uma alternativa de solução para este problema é implementar um procedimento de re-roteamento no algoritmo, realocando as requisições de modo a melhorar os valores de atraso obtidos para a classe B.

5.6 Conclusão

Após avaliar todos os cenários de simulação é possível concluir que a otimização multi-objetivo apresentou bons resultados para o problema de roteamento dinâmico em redes MPLS. Através desta abordagem é possível obter rotas LSP na rede que privilegiem mais de um indicador de QoS, o que não é possível ao utilizar a otimização mono-objetivo no roteamento.

O roteamento utilizando a otimização multi-objetivo obteve simultaneamente valores de atraso médio e balanceamento de carga próximos dos ótimos. Os resultados obtidos apresentaram o mesmo comportamento tanto para o surgimento de requisições aleatórias, como requisições de *broadcast* e em diferentes topologias, demonstrando a eficácia desta abordagem em diferentes contextos.

Além da otimização simultânea de mais de um indicador de QoS, a abordagem multi-objetivo mostrou flexibilidade na alocação das rotas LSP na rede. Apenas al-

terando os parâmetros do procedimento de tomada de decisão, foi possível realizar o roteamento para contextos de rede completamente distintos.

Uma outra característica estudada neste trabalho e avaliada nos cenários de simulação é o tratamento das requisições em diferentes classes de serviço. Ao agrupar as requisições em classes que demandam prioridades distintas no roteamento é possível que o roteamento seja feito mais adequadamente, atribuindo rotas LSP às requisições dentro de suas necessidades. Os resultados comprovaram a eficiência desta alternativa, embora melhorias ainda possam ser realizadas para melhorar os valores obtidos.

Capítulo 6

Considerações finais

Este trabalho apresentou uma abordagem multi-objetivo para a alocação de rotas LSP em redes MPLS em um cenário dinâmico de otimização. Foi proposta uma formulação matemática para a solução do problema, mas devido à sua complexidade computacional elevada uma heurística foi implementada para obter soluções aproximadas eficientemente. Esta heurística se trata de um algoritmo genético multi-objetivo.

As soluções obtidas pelo algoritmo genético estão próximas do conjunto de soluções ótimas do problema, denominado conjunto Pareto-ótimo, conforme mostraram os resultados computacionais apresentados. A eficiência apresentada pelo algoritmo viabilizou sua utilização em um contexto dinâmico de otimização, onde as respostas para alocação das rotas LSP precisam ser imediatas.

Devido à otimização multi-objetivo fornecer como resultado um conjunto de soluções não-dominadas, foi necessário implementar um procedimento para tomada de decisão para escolher entre as soluções deste conjunto aquela mais adequada para solucionar o problema. O método AHP foi escolhido para o processo de tomada de decisão e permitiu que as soluções fossem escolhidas baseando-se nas preferências do decisor.

Com o intuito de validar a proposta de otimização multi-objetivo para as redes MPLS, foram conduzidas simulações utilizando software NS-2. Diferentes topologias de rede foram adotadas, bem como diferentes estratégias para o surgimento das requisições na rede, o que permitiu a formulação de diversos cenários de simulação para avaliar a estratégia proposta.

Quanto aos resultados apresentados, conclui-se que a otimização multi-objetivo obteve sucesso no problema de roteamento dinâmico em redes MPLS. Diferentes indicadores de QoS puderam ser otimizados simultaneamente, o que proporciona o desempenho global da rede ao avaliar diferentes indicadores.

Além da otimização simultânea de diferentes indicadores de QoS, com os cenários de simulação foi possível mostrar a flexibilidade obtida ao se realizar o roteamento

nas redes MPLS através da otimização multi-objetivo. Apenas alterando os valores de prioridade das funções objetivo no método de decisão AHP é possível obter soluções que se adequem a diferentes cenários de rede, sem a necessidade de alterar o procedimento de otimização.

É proposta também uma estratégia para agrupar as requisições em classes de serviço com diferentes características. As classes estão associadas a diferentes características que devem ser privilegiadas durante o roteamento. Foram propostos cenários de simulação para avaliar o desempenho desta proposta no ambiente de redes MPLS, e os resultados apresentados foram satisfatórios, embora algumas ajustes possam ser realizados para melhorar a qualidade da proposta.

Como possibilidades de trabalhos futuros poderão ser conduzidas melhorias no algoritmo, principalmente relativas ao tratamento diferenciado de classes afim de melhorar os resultados obtidos. Também poderá ser implementada uma outra heurística para aproximar os resultados do problema multi-objetivo, de modo que seus resultados possam ser comparados àqueles obtidos com o algoritmo genético apresentado. O processo de tomada de decisão poderá ser reavaliado e outras estratégias podem ser implementadas, para que esse processo de tomada de decisão se torne mais automático e menos dependente das preferências do decisor.

Também como continuidade do trabalho, espera-se adotar mecanismos de predição de tráfego que permitam estimar o comportamento da rede em um momento futuro, de modo que as decisões no presente possam utilizar este conhecimento. Assim, o processo de tomada de decisão se tornaria mais adequado ao tráfego da rede durante um período de operação.

Referências Bibliográficas

- Awduche, D. (1999). MPLS and traffic engineering in IP networks. *IEEE Communications Magazine*, 37(12).
- Deb, K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester.
- Deb, K.; Agrawal, S.; Pratab, A. e Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pp. 849–858, Paris, France.
- Dias, R.; Camponogara, E.; Farines, J.-M.; Willrich, R. e Campestrini, A. (2003). Otimização Lagrangeana em engenharia de tráfego para redes IP sobre MPLS. *XXI Simpósio Brasileiro de Redes de Computadores*.
- Dias, R.; Farines, J.-M.; Camponogara, E.; Willrich, R. e Campestrini, A. (2005). Engenharia de tráfego dinâmica em redes IP sobre tecnologia MPLS: Otimização baseada em heurísticas. *XXIII Simpósio Brasileiro de Redes de Computadores*.
- Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numer. Math.*, 1:269–271.
- Erbas, S. C. e Erbas, C. (2003). A multiobjective off-line routing model for MPLS networks. In *18th International Teletraffic Congress*, pp. 471–480, Berlin. Elsevier.
- Evans, J. e Filsfil, C. (2007). *Deploying IP and MPLS QoS for multiservice networks*. Elsevier, San Francisco.
- Floyd, R. W. (1962). Algorithm 97: Shortest path. *Communications of the ACM*, 5(6):345.
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional.

- Hedrick, C. (1988). Routing information protocol. *The Internet Engineering Task Force - IETF*, RFC 1058.
- Ilog Incorporation (2006). Solver CPLEX v10. www.ilog.com/products/cplex/ (accesado em 06/06/2008).
- 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*, 18:2566–2579.
- Knowles, J. e Corne, D. (2000a). M-PAES: A memetic algorithm for multiobjective optimization. In *Proceedings of the 2000 Congress on Evolutionary Computation CEC00*, pp. 325–332, California, USA. IEEE Press.
- Knowles, J.; Oates, M. e Corne, D. (2000). Advanced multi-objective evolutionary algorithms applied to two problems in telecommunications. *BT Technology Journal*, 18(4):51–65.
- Knowles, J. D. e Corne, D. W. (2000b). Approximating the nondominated front using the Pareto archived evolution strategy. *Evolutionary Computation*, 8(2):149–172.
- Koyama, A.; Barolli, L.; Matsumoto, K. e Apduhan, B. O. (2004). A GA-based multi-purpose optimization algorithm for QoS routing. In *AINA '04: Proceedings of the 18th International Conference on Advanced Information Networking and Applications*, p. 23, Washington, DC, USA. IEEE Computer Society.
- Moy, J. (1991). OSPF version 2. *The Internet Engineering Task Force - IETF*, RFC 1247.
- Oliveira, A. (2005). *Um arcabouço para engenharia de tráfego em redes MPLS*. PhD thesis, Universidade Federal de Minas Gerais.
- Oliveira, A. e Mateus, G. R. (2006). Using genetic algorithms to LSP setup in MPLS networks. *XXIV Simpósio Brasileiro de Redes de Computadores*.
- Oran, D. (1990). Osi is-is intra-domain routing protocol. *The Internet Engineering Task Force - IETF*, RFC 1142.
- Papadimitriou, C. H. (1997). Np-completeness: A retrospective. In *ICALP '97: Proceedings of the 24th International Colloquium on Automata, Languages and Programming*, pp. 2–6, London, UK. Springer-Verlag.
- Rosen, E.; Viswanathan, A. e Callon, R. (2001). Multiprotocol Label Switching Architecture. *RFC Editor*, RFC 3031.

- Saaty, T. (1980). *The Analytic Hierarchy Process, Planning, Priority Setting, Resource Allocation*. McGraw-Hill, New York.
- Shao, H.; Chen, X. e Wang, W. (2006). A multiobjective optimization algorithm for LSP setup in diffserv and MPLS networks. *First International Conference on Communications and Networking*.
- Steuer, R. E. (1985). *Multiple Criteria Optimization: Theory, Computation and Application*. John Wiley and Sons, New York.
- Takahashi, R. H. C. (2004). Otimização escalar e vetorial - notas de aula. Technical Report 1, UFMG - Belo Horizonte.
- Tanenbaum, A. S. (2002). *Computer Networks*. Prentice Hall.
- Tanezine, A. B. e dos Santos, R. M. (2006). Rede privada virtual utilizando a arquitetura MPLS em redes IP. Technical report. Monografia de Especialização em Especialização em Telecomunicações.
- Ulungu, E. e Teghem, J. (1994). Multi-objective combinatorial optimization problems: A survey. *Journal of Multi-Criteria Decision Analysis*, 3:83–104.
- Wu, J. e Azarm, S. (2001). Metrics for quality assessment of a multiobjective design optimization solution set. *Journal of Mechanical Design*, 123(1):18–25.

