Otimização da Função de Roteamento para a Engenharia de Tráfego em Redes IP

Ronaldo Moreira Salles[1] **, Vitor Guerra Rolla**[1*]

[1]Departamento de Sistemas e Computação
Instituto Militar de Engenharia
Praça General Tibúrcio, 80 Praia Vermelha
22290-270 Rio de Janeiro, RJ

salles@ieee.org, vitor.rolla@ieee.org

Abstract. *This work presents an alternative analysis for the shortest path optimal routing problem and proposes novel heuristic methods to solve it. In practical terms, a solution for the shortest path optimal routing problem determines the link weights that optimizes a computer network operating under standard routing protocols (e.g. OSPF). We based our solution on traffic engineering techniques that respect the OSPF routing model. We considered the main objectives of traffic engineering, such as: load balancing, better use of available resources and capacity to support growing traffic demands. Our proposal presents better results than traditional approaches.*

Resumo. *Esse trabalho apresenta uma abordagem alternativa para o problema do roteamento ótimo de menor caminho e propõe novos algoritmos para a solução do problema. Em termos gerais, uma solução para o roteamento ótimo de menor caminho determina quais são os pesos ideais para os enlaces da rede de forma que protocolos de roteamento tradicionais (p.ex. OSPF) possam otimizar a distribuição do tráfego na rede. Foram considerados os objetivos principais da engenharia de tráfego: balanceamento de carga, otimização do uso dos recursos da rede e tolerância a demandas crescentes de tráfego. Resultados numéricos mostraram um desempenho bastante significativo para os algoritmos propostos.*

1. Introdução

A engenharia de tráfego abrange a aplicação de tecnologias e princípios científicos de medição, caracterização, modelagem, e controle do tráfego em uma rede de comunicação. O objetivo principal da engenharia de tráfego é otimizar o desempenho da rede por meio de operaçãoes de gerência da capacidade e/ou de gerência do tráfego [Awduche 2002]. A gerência do tráfego inclui funções de controle do tráfego tais como o condicionamento do tráfego e o controle das filas nos roteadores. Por outro lado, a gerência da capacidade inclui o planejamento de capacidade, a alocação dos recursos disponíveis, e o controle da função de roteamento que é o foco desse trabalho.

O protocolo Open Shortest Path First (OSPF)[Moy 1991] é o protocolo de roteamento intradomínio mais utilizado na Internet. O protocolo se tornou bastante utilizado

*Apoiado pelo Instituto Militar de Engenharia e CAPES.

por sua confiabilidade, escalabilidade e robustez. O OSPF fundamenta-se no estado do enlace e em algoritmos de menor caminho, tais como Dijkstra e Bellman-Ford. Os nós que fazem parte de um domínio OSPF trocam informações que proporcionam a cada nó a implementação de um banco de dados que represente a topologia da rede. O algoritmo de menor caminho é utilizado para construir uma árvore de menor caminho para todos os destinos, levando em conta o estado do enlace e seu peso (configurados pelos administradores da rede). Os fabricantes de roteadores sugerem que o peso atribuído a cada enlace deva ser $1/C$, onde C é a capacidade de transmissão do enlace de comunicação. Tal acertiva pode ser observada em [Thomas 1998].

Uma das dificuldades centrais do OSPF para a engenharia de tráfego deve-se a utilização do paradigma de encaminhamento baseado no destino. A origem do tráfego não tem controle e nem pode influenciar sobre o procedimento de seleção do caminho, a decisão de roteamento é efetuada em cada roteador. O algoritmo baseado no menor custo seleciona os caminhos mais curtos para preencher a tabela de roteamento. Caso os caminhos mais curtos estejam sobrecarregados, outros caminhos de maior comprimento não serão utilizados mesmo se estiverem subutilizados.

Em função das dificuldades encontradas pelo OSPF para a implementação de procedimentos destinados à engenharia de tráfego, outras soluções são comumente adotadas. A tecnologia MPLS [Rosen et al. 2001] vem sendo amplamente difundida como uma das principais ferramentas para atender aos requisitos da engenharia de tráfego [Srinivasan et al. 2004]. Além de muitos atrativos da tecnologia MPLS para essa aplicação, pode-se dizer em termos gerais que o MPLS fornece uma camada orientada à conexão para redes IP, o que facilita o estabelecimento de caminhos alternativos e os procedimentos gerais de roteamento baseado em restrições [Awduche 1999]. Apesar das aparentes vantagens da tecnologia MPLS, a maioria dos provedores de serviços empregam protocolos de roteamento tradicionais em função da experiência legada, da alta confiabilidade, escalabilidade e robustez desse protocolos.

Fortz e Thorup em [Fortz and Thorup 2002] estudaram o comportamento dos algoritmos de roteamento tradicionais no desempenho das funções de engenharia de tráfego. Os resultados obtidos mostraram que o desempenho dos algoritmos pode atingir praticamente os mesmos níveis do desempenho da tecnologia MPLS caso os algoritmos operem com um conjunto de pesos ideais nos enlaces de comunicação. A determinação dos pesos ideais envolve a solução do problema de roteamento ótimo de menor caminho, no entanto, esse problema é NP-completo [Fortz and Thorup 2000].

A motivação principal para a realização desse trabalho fundamenta-se na investigação de soluções alternativas para o problema do roteamento ótmio de menor caminho, de forma que os protocolos de roteamento legados possam ser empregados com eficiência nas operações de engenharia de tráfego. Nossas propostas são avaliadas de acordo com três aspectos principais: $i)$ distância dos pontos retornados em relação ao ótimo absoluto (roteamento ótimo com bifurcação), $ii)$ melhoria em relação às especificações dos fabricantes, e $iii)$ tempo de execução dos algoritmos.

Esse trabalho está organizado da seguinte forma. A Seção 2 descreve as categorias de problemas de roteamento ótimo, incluindo o estudo do problema de determinação do peso ótimo de um único enlace. Tais problemas ajudam a ilustrar e embasar a discussão e o entendimento das restrições impostas pelo problema de roteamento ótimo de menor caminho (problema do roteamento OSPF). A Seção 3 apresenta as propostas de solução para o problema. Os resultados obtidos da implementação das propostas são apresentados na Seção 4 e o trabalho concluído na Seção 5.

2. Categorias de Problemas de Roteamento

2.1. Problema Geral do Roteamento Ótimo

O problema geral do roteamento ótimo consiste em distribuir o tráfego de cada demanda OD (tráfego total entre um nó Origem e um nó Destino) entre os vários caminhos disponíveis na rede, de forma a minimizar uma certa função de custo global. Esse problema também é conhecido na literatura como problema do roteamento ótimo com bifurcação. A formulação matemática desse problema pode ser encontrada em várias referências. No entanto, utilizaremos a formulação encontrada em [Bertsekas and Gallager 1992] por ser simplificada:

$$\min \quad \sum_{(i,j)} D_{ij}(F_{ij}) \tag{1}$$

sujeito a

$$F_{ij} = \sum_{p \ni (i,j)} x_p \tag{2}$$

$$\sum_{p \in P_w} x_p = r_w, \quad \forall w \in W \tag{3}$$

$$x_p \geq 0, \quad \forall p \in P_w, w \in W \tag{4}$$

onde,

- $D_{ij}(F_{ij})$: é a função de custo associada ao enlace (i, j)
- F_{ij}: fluxo total que atravessa o enlace (i, j)
- p: um dado caminho fim-a-fim
- x_p: fluxo no caminho p
- W: conjunto de todos os pares OD
- P_w: conjunto de todos os caminhos da rede que conectam o par OD w
- r_w: demanda do par OD w

Cada termo do somatório (1) é dado por:

$$D_{ij}(F_{ij}) = \frac{F_{ij}}{C_{ij} - F_{ij}} \tag{5}$$

onde C_{ij} é capacidade do enlace (i, j). A equação (5) age como uma barreira para o enlace (i, j) – quanto mais próximo o valor de F_{ij} se aproximar de C_{ij}, D_{ij} crescerá sem limite.

A minimização dessa função de tráfego se dá roteando parcelas dos fluxos nos enlaces mais carregados para aqueles com menos carga, preservando as demandas r_w fim-a-fim. Pode-se notar que as funções de barreira têm uma interpretação física, elas representam o número médio de pacotes se cada enlace for modelado como uma fila

M/M/1. Assim, podemos vislumbrar o objetivo do problema como sendo distribuir as cargas das demandas de forma a minimizar o número médio de pacotes na rede.

A idéia básica do problema geral do roteamento ótimo é equilibrar as cargas das demandas. Um fluxo de um par OD pode ser distribuído por um, vários ou todos os caminhos existentes entre o par. Os fluxos nos enlaces são variáveis contínuas. Essa característica permite que o problema seja tratado matematicamente [Ramakrishnan and Rodrigues 2001].

O problema geral do roteamento ótimo é bem conhecido e já estudado pela comunidade acadêmica. As soluções mais recomendadas para o problema são: o método de desvio de fluxos (*flow deviation method*) [Kleinrock 1993] e o método de projeção do gradiente (*gradient projection method*) [Bertsekas and Gallager 1992]. Chamaremos o problema geral do roteamento ótimo utilizando a função de custo (5) de ORP-Delay.

É possível formular o problema geral do roteamento ótimo utilizando uma função objetivo diferente. No lugar da função (1) teremos:

$$\min \max_{(i,j)} \left\{ \frac{F_{ij}}{C_{ij}} \right\} \tag{6}$$

onde F_{ij}/C_{ij} representa a taxa de utilização do enlace (i, j) . A solução para esse novo problema nos retorna a melhor distribuição de carga que minimiza a utilização máxima na rede, caracterizando a otimização do balanceamento de carga em uma rede de comunicação. Chamaremos esse problema de ORP-Utilization.

A solução de ambos os problemas de roteamento acima apresentados são importantes para o estudo, visto que, representam limites às demais categorias de problemas de roteamento. As soluções fornecem a melhor forma de se rotear demandas em uma rede de comunicação. Por essas características, essas soluções serão utilizadas como parâmetros na avaliação do desempenho das soluções propostas para o problema do roteamento ótimo de menor caminho (roteamento OSPF).

2.2. Problema do Roteamento Ótimo de Caminho Único

O problema do roteamento ótimo de caminho único consiste em determinar, para cada demanda OD, um único caminho pelo qual a demanda passará integralmente, tal que a combinação de todos os caminhos selecionados (um para cada demanda) minimize uma certa função de custo global. Esse problema também é conhecido na literatura como problema do roteamento ótimo sem bifurcação. Ele está estritamente relacionado com o problema geral do roteamento ótimo e com o problema do roteamento ótimo de menor caminho, representando um meio termo entre esses dois últimos. Chamaremos o problema do roteamento ótimo de caminho único de Single-ORP-Delay para a formulação com a função objetivo $\min \sum_{(i,j)} D_{ij}(F_{ij})$. Desta forma, chamaremos de Single-ORP-Utilization a formulação com a função objetivo $\min \max_{(i,j)} \{F_{ij}/C_{ij}\}$. Segue abaixo a formulação matemática [Bertsekas and Gallager 1992] do problema Single-ORP-Delay:

$$\min \quad \sum_{(i,j)} D_{ij}(F_{ij}) \tag{7}$$

sujeito a

$$F_{ij} = \sum_{p \ni (i,j)} x_p \tag{8}$$

$$\sum_{p \in P_w} \delta_p x_p = r_w, \quad \forall w \in W \tag{9}$$

$$\sum_{p \in P_w} \delta_p = 1, \quad \delta_p = \{0, 1\} \tag{10}$$

$$x_p \geq 0, \quad \forall p \in P_w, w \in W \tag{11}$$

onde,

- $D_{ij}(F_{ij})$: é a função de custo associada ao enlace (i, j)
- F_{ij}: fluxo total que atravessa o enlace (i, j)
- p: um dado caminho fim-a-fim
- x_p: fluxo no caminho p
- W: conjunto de todos os pares OD
- P_w: conjunto de todos os caminhos da rede que conectam o par OD w
- r_w: demanda do par OD w
- δ_p indica se o caminho p é o caminho selecionado ($= 1$) ou não ($= 0$).

A característica que difere o problema geral do roteamento ótimo e o problema do roteamento ótimo de caminho único é que no último existe a obrigatoriedade do tráfego de cada demanda passar integralmente por um único caminho entre o nó origem e o nó destino. Com essa característica o problema passa a ser combinatório e NP-difícil. Em [Wang and Wang 1999] pode-se encontrar uma heurística proposta pelos autores para resolver esse problema. A heurística basicamente executa a solução do ORP-Utilization e re-roteia as demandas que foram segmentadas (sofreram bifurcação) não permitindo a bifurcação de forma a minizar o enlace com maior utilização.

2.3. Problema do Roteamento Ótimo de Menor Caminho

O problema do roteamento ótimo de menor caminho ou problema do roteamento OSPF consiste em encontrar os pesos ótimos para os enlaces de uma rede de comunicação que possibilite ao roteamento OSPF minimizar uma certa função de custo global. Chamaremos o problema do roteamento ótmio de menor caminho de Short-ORP. A principal característica do problema Short-ORP é que as variáveis desse problema são os pesos que devem ser configurados em cada enlace de comunicação. Tal característica torna a formulação matemática do problema Short-ORP muito complexa. Segue abaixo a formulação matemática do problema de acordo com [Harmatos 2001]:

$$\max \quad \sum_e b_e * (y_e - \sum_d \sum_j v_{edj} * x_{dj}(w)) \tag{12}$$

sujeito a

$$\sum_j x_{dj}(w) = h_d \quad d = 0, 1, 2, 3, ..., D \tag{13}$$

$$\sum_d \sum_j v_{edj} * x_{dj}(w) \leq y_e \quad e = 0, 1, 2, 3, ..., E \tag{14}$$

$$1 \leq w_e \leq K \quad e = 0, 1, 2, 3, ..., E \tag{15}$$

onde,

- h_d: é o volume mínimo da demanda d;
- v_{edj}: 1 se o enlace pertencer ao caminho j contendo a demanda d e 0 caso contrário.
- y_e: é a capacidade do enlace e;
- b_e: valor de uma unidade de capacidade ociosa no enlace e;
- K: limite para os pesos;
- x_{dj}: fluxo que contém a demanda d no caminho j implicado pelo sistema de pesos
- w_e: peso do enlace e
- a_e: capacidade alocada no enlace e

A função (12) tem como objetivo maximizar a capacidade residual da rede. As restrições garantem que todas as demandas sejam respeitadas, que as capacidades dos enlaces não sejam ultrapassados e que os pesos sejam inteiros e positivos, respectivamente. A resolução deste problema com softwares de otimização é inviável para redes com mais de 12 nós e 25 enlaces. O tempo de execução pode ultrapassar 30 horas para chegar a uma solução ótima [Harmatos 2001].

Um aspecto importante que devemos levar em consideração no problema Short-ORP é o acoplamento. O conceito de acoplamento pode ser definido da seguinte forma: se dois caminhos têm interseção em dois nós, o subcaminho entre os dois nós deve ser idêntico [Ramakrishnan and Rodrigues 2001]. Por exemplo, se no grafo (rede de comunicação) apresentado na Fig.1 o menor caminho entre os nós *A* e *D* for *ABCD*, obrigatoriamente o menor caminho entre os nós *X* e *Z* deve ser *XBCDZ*. Podemos observar que o subcaminho entre os nós *B* e *D* tem que ser o mesmo nos dois caminhos.

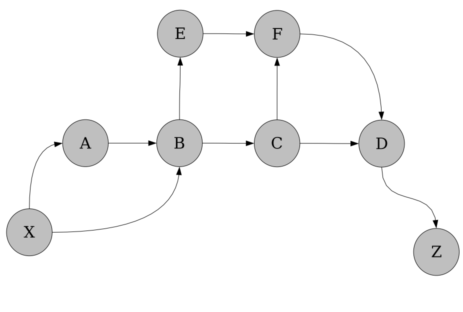

Figure 1. Exemplo de Acoplamento

Em [Fortz and Thorup 2000] os autores provam que o problema Short-ORP é NP-completo. Por esse motivo os estudos que procuram soluções para o problema normalmente propõem soluções heurísticas.

2.4. Problema de Determinação do Peso Ótimo de Um Enlace

Como já mencionamos anteriormente, o Short-ORP é um problema NP-difícil. A complexidade de tal problema nos motivou a estudar e formular um outro problema com menos restrições. Denominamos o problema de determinação do peso ótimo de um único enlace da rede de One-ORP.

O problema One-ORP tem como objetivo determinar o peso ótimo de um determinado enlace da rede, dados os pesos de todos os outros enlaces. Segue abaixo a formulação matemática do problema:

$$\min \quad \sum_{(i,j)} D_{ij}(F_{ij}) \tag{16}$$

sujeito a

$$n_w \begin{cases} < m_w & \text{, se } x \text{ absorver a demanda } w \\ > m_w & \text{, se } x \text{ repelir a demanda } w \\ = m_w & \text{, se houver 2 menores caminhos} \end{cases} \quad , \quad \forall w \in W \tag{17}$$

$$p_x > 0 \tag{18}$$

onde,

- x: é o enlace cujo o peso ótimo deve ser determinado
- p_x: é o peso ótimo a ser determinado para o enlace x
- W: conjunto de todos os pares OD
- n_w: é o menor caminho entre o par OD $w \in W$ que utiliza o enlace x
- m_w: é o menor caminho entre o par OD $w \in W$ que não utiliza o enlace x
- $D_{ij}(F_{ij})$: é a função de custo associada ao enlace (i, j)
- F_{ij}: fluxo total que atravessa o enlace (i, j)

A primeira restrição do problema determina as faixas de valores relevantes para p_x. Isto é, os limites entre as diferentes soluções possíveis para o problema. A segunda restrição garante que p_x é positivo.

A complexidade de pior caso do problema One-ORP é $O(2 * (n^2 - n))$ ou $O(w)$, pois cada par OD w que possuir pelo menos um caminho viável que passe pelo enlace x vai gerar duas faixas de valores a serem testadas. A solução força bruta para esse problema pode ser efetuada através do teste das faixas de valores viáveis que p_x pode tomar.

3. Otimizando o Peso dos Enlaces

Nessa seção apresentamos três propostas heurísticas para resolver o problema Short-ORP. O motivo de apresentarmos mais de uma proposta de solução para o problema é fundamentado pela evolução da compreensão do problema Short-ORP e principalmente pelo desempenho de otimização que os algoritmos propostos apresentam para diferentes parâmetros.

Em nossas heurísticas optamos por utilizar operações determinísticas. Tais operações evitam a utilização de processos randômicos. Heurísticas determinísticas são aquelas que consideram a teoria operacional juntamente com as restrições e procedimentos específicos do problema a ser resolvido. Acreditamos que os algoritmos desenvolvidos com operações determinísticas têm a capacidade de prover soluções melhores que as meta-heurísticas conhecidas: Simulated Annealing, Simulated Allocation e Algoritmos Evolucionários.

3.1. HeurCoupling

A solução HeurCoupling baseia-se principalmente na característica de acoplamento do problema Shor-ORP. Ela difere das demais soluções estudadas pelo fato de dividir o problema em duas etapas distintas. A primeira etapa consiste em estabelecer os caminhos já acoplados, de forma a minimizar o custo de roteamento pelos quais as demandas devem ser roteadas. A segunda etapa consiste em determinar os pesos aos enlaces da rede que reflitam os caminhos estabelecidos na primeira etapa.

3.1.1. Primeira Etapa: Estabelecer Sistemas de Caminhos Acoplados

A idéia básica da primeira etapa do HeurCoupling é rotear as demandas uma a uma, da maior para a menor, com o objetivo de criar o sistema de caminhos acoplados. Utilizamos o conceito de quantidade de banda disponível para representar o custo de se tráfegar por cada enlace. O custo do enlace l é $c_l = \frac{1}{A_l}$, onde A_l é igual a capacidade do enlace l menos a carga submetida a esse enlace. Portanto, roteamos as demandas por caminhos menos custosos. A justificativa por trás desse modelo é forçar que as grandes demandas trafeguem sobre uma rede menos congestionada e utilizem os enlaces com mais recursos disponíveis (custos c_l baixos). Esse procedimento procura reduzir o impacto das grandes demandas sobre as funções de barreira (5), além de diminuir a incidência de caminhos acoplados.

Segue abaixo o pseudo código da primeira etapa do algoritmo HeurCoupling, onde: W é o conjunto de demandas, r_w são as demanadas a serem roteadas, L é o conjunto de enlaces da rede, C_l a capacidade do enlace l, A_l a largura disponível no enlace e c_l o custo de se trafegar pelo enlace.

HeurCoupling: Finding Paths

0. para todo $l \in L$: $c_l \leftarrow 1/C_l$
1. encontre $r_j \in W$ tal que r_j é máximo
2. rotear r_j através do caminho mais curto p_j (Dijkstra's)
3. se p_j está acoplado **vá para 4.**
senão recompute p_j
4. atualize todos os custos nos enlaces: $c_l \leftarrow 1/A_l$
5. $W \leftarrow W - \{r_j\}$
6. se $W = \varnothing$: **FIM**
senão vá para 1.

No passo (0) configura-se os custos dos enlaces com a recomendação padrão $\frac{1}{C_l}$. Neste momento nenhuma demanda ainda foi roteada, portanto não existem cargas nos enlaces. Nos passos (1) e (2), procura-se a maior demanda ainda não roteada e determina-se o caminho menos custoso p_j, respectivamente. Tal caminho pode ser alterado no passo (3) para se acoplar a caminhos previamente determinados. No passo (4) os custos dos enlaces são atualizados para refletir as demandas que foram roteadas e consumiram recursos dos enlaces. No passo (5) o conjunto W é atualizado e o procedimento termina no passo (6) caso não existam mais demandas a serem roteadas.

A complexidade de pior caso da primeira etapa do algoritmo HeurCoupling, se considerarmos w como sendo o número de demandas OD do conjunto W e n como sendo o número de roteadores na rede, é $O\left(wn^2\right)$ ou $O(n^4)$, pois cada demanda é roteada utilizando-se o algoritmo de Dijkstra.

3.1.2. Segunda Etapa: Programação Linear

A segunda etapa consiste em resolver o problema do menor caminho inverso determinando os pesos finais aos enlaces da rede que reflitam os caminhos estabelecidos na primeira etapa. Formalizando o problema, temos: dado um conjunto de caminhos, conjunto este que respeite a regra do acoplamento, o problema consite em atribuir pesos às arestas do grafo (enlaces da rede), tal que o roteamento OSPF reflita o conjunto de caminhos estabelecidos na primeira etapa.

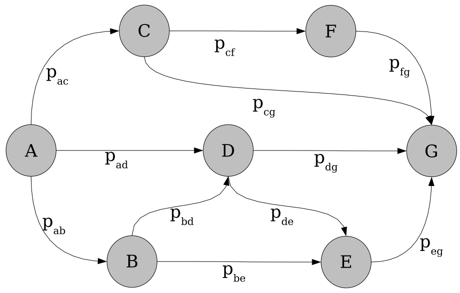

Figure 2. Diminuindo Restrições

Uma solução para o problema seria modelar um sistema de restrições lineares que garanta que o peso total do caminho selecionado para um par OD deva ser menor que os pesos dos outros caminhos (não selecionados) entre o par OD. Vamos considerar o mesmo exemplo proposto no artigo [Mahajan et al. 2002] onde p_{od} representa o peso do enlace com origem em O e destino em D. Na Fig.2, se assumirmos que o menor caminho para o par com origem em A e destino em G seja ADG. Obrigatoriamente teremos o conjunto de restrições abaixo para garantir que ADG seja o menor caminho:

1)$p_{ad} + p_{dg} < p_{ab} + p_{be} + p_{eg}$

2)$p_{ad} + p_{dg} < p_{ac} + p_{cg}$

3)$p_{ad} + p_{dg} < p_{ab} + p_{bd} + p_{dg}$

4)$p_{ad} + p_{dg} < p_{ab} + p_{bd} + p_{de} + p_{eg}$

5)$p_{ad} + p_{dg} < p_{ac} + p_{cf} + p_{fg}$

6)$p_{ad} + p_{dg} < p_{ad} + p_{de} + p_{eg}$

A solução clássica produz um número exponencial de restrições, muitas delas redundantes. Para cada caminho não selecionado teremos uma restrição. Porém, no artigo [Mahajan et al. 2002], os autores propõem um modelo simples e eficaz para reduzir o número de restrições, que vem a ser o seguinte.

Assumindo-se S_{od} como o caminho selecionado entre os vértices O e D (caminho que pertence ao conjunto dos caminhos já acoplados gerados na primeira etapa) e como $P\left(S_{od}\right)$ o peso para esse caminho. Sendo $A_{oi} : A_{id}$ um caminho alternativo que passa no vértice intermadiário I, logo $P\left(A_{oi} : A_{id}\right)$ é o peso desse caminho alternativo. O critério criado pelos autores do artigo [Mahajan et al. 2002] afirma que a restrição $P\left(S_{od}\right) < P\left(A_{oi} : A_{id}\right)$ é redundante se pelo menos uma das condições abaixo, para qualquer vértice intermediário, for verdadeira:

i) A_{oi} não é o caminho selecionado entre os vértices O e I;

ii) A_{id} não é o caminho selecionado entre os vértices I e D.

Adotando esse simples critério o sistema de restrições lineares passa a ter um número polinomial de restrições no número de vértices. Para o exemplo da Fig.2, é fácil de observar que as restrições 3, 4, 5 e 6 passam a ser redundante se aplicarmos o critério e assumirmos que o caminho selecionado entre B e G é BEG.

3.2. HeuRoSa

A HeuRoSa difere da HeurCoupling pelo fato de propor uma solução de alteração direta dos pesos. A idéia básica da HeuRoSa é efetuar uma única alteração de peso a cada iteração do algoritmo. A alteração do peso resultará no re-roteamento do caminho de uma única demanda OD. Essa característica do algoritmo possibilita um maior controle sobre o processo evolutivo da solução, caracterizando um procedimento determinístico.

Para que possamos re-rotear uma única demanda a cada interação do algoritmo precisamos calcular o valor δ. Esse valor representa o quanto devemos somar ($\delta_{superior}$) ou diminuir ($\delta_{inferior}$) do valor do peso de um determinado enlace para que somente uma demanda w, que trafegue por esse enlace, seja re-roteada.

O algoritmo executa duas funções distintas. A primeira função efetua a diminuição dos valores dos pesos dos enlaces menos utilizados e a segunda efetua aumento dos valores dos pesos dos enlaces mais utilizados. O algoritmo só para de ser executado quando as duas funções não conseguem mais evoluir no objetivo de diminuir o custo. Isto é, toda vez que uma das funções conseguir melhorar o custo, através da alteração de um peso, a outra função será re-executada. Esse modelo permite que o algoritmo se torne mais flexível com o objetivo de possibilitar que a solução caminhe o mais próximo em direção do ótimo global, driblando os mínimos locais.

O primeiro passo do algoritmo é configurar os pesos dos enlaces de acordo com a proposta padrão dos fabricantes $\frac{1}{C}$. Portanto, o custo inicial $\sum_{ij} D_{ij}\left(F_{ij}\right)$ é o custo recomendado pelos fabricantes. Segue abaixo o pseudocódigo da função $\delta_{superior}$ do algoritmo e em seguida uma explanação a respeito das funções $\delta_{superior}$ e $\delta_{inferior}$.

HeuRoSa

0. para todo $l \in L$: $w_l \leftarrow \frac{1}{C_l}$
1. encontre $l \in L$ tal que a utilização de l é máxima
2. calcule $\delta_{superior}$ para o enlace l
3. $w_l \leftarrow w_l + \delta_{superior}$
4. calcule $novocusto$ de roteamento
5. se $novocusto < custo$: $custo \leftarrow novocusto$ e volte para o **Passo 1**
senão $L \leftarrow L \setminus \{l\}$
6. se $L = \varnothing$: **FIM**
senão volte para o **Passo 1**

A função de aumento do valor dos pesos procura o enlace mais utilizado da rede, calcula o $\delta_{superior}$ deste enlace, atualiza o peso deste enlace: $w_l + \delta_{superior}$ (re-roteando um único caminho que trafega nesse enlace). A função computa o custo $\sum_{l \in L} D_l(F_l)$ da nova solução com o peso alterado. Caso o custo da nova solução seja melhor que o custo encontrado até o momento, a alteração do valor do peso é mantida e a função é re-executada. Caso o custo da nova solução seja pior, a alteração do valor do peso não é mantida. Então, a função procura o segundo enlace mais utilizado e efetua os mesmos passos e assim sucessivamente até que todos os enlaces sejam testados.

A função de diminuição do valor dos pesos é inversa a função de aumento já abordada. Ela procura o enlace menos utilizado da rede, calcula o $\delta_{inferior}$ deste enlace, atualiza o peso deste enlace: $w_l - \delta_{inferior}$ (re-roteando um único caminho para esse enlace). A função computa o custo $\sum_{l \in L} D_l(F_l)$ da nova solução com o peso alterado. Caso o custo da nova solução seja melhor que o custo encontrado até o momento, a alteração do valor do peso é mantida e a função é re-executada. Caso o custo da nova solução seja pior, a alteração do valor do peso não é mantida. Então, a função procura o segundo enlace menos utilizado e efetua os mesmos passos e assim sucessivamente até que todos os enlaces sejam testados.

3.3. HeurOne

A proposta de solução HeurOne utiliza o problema One-ORP previamente apresentado na seção 2.4. Essa proposta também baseia-se na atualização direta dos pesos. A cada iteração, o algoritmo procura o enlace com maior utilização da rede e resolve o problema One-ORP para tal enlace. Determinado o peso ótimo para o enlace, recalcula-se o custo $\sum_{l \in L} D_l(F_l)$. Caso o custo tenha diminuido, a alteração do peso é efetivada e a procura pelo enlace com maior utilização recomeça. Caso o custo não tenha diminuido, o algoritmo procura o segundo enlace com maior utilização na rede e assim sucessivamente até que o custo seja otimizado ou que todos os enlaces tenham sido testados.

O primeiro passo do algoritmo é configurar os pesos dos enlaces de acordo com a proposta padrão dos fabricantes $\frac{1}{C}$. Portanto, o custo inicial $\sum_{l \in L} D_l(F_l)$ é o custo recomendado pelos fabricantes. Segue abaixo o pseudocódigo do algoritmo, onde: L é o conjunto de enlaces, C_l a capacidade do enlace l e w_l o seu peso.

HeurOne

0. para todo $l \in L : w_l \leftarrow \frac{1}{C_l}$
1. calcule $custo$ de roteamento
2. encontre $l \in L$ tal que a utilização de l é máxima
3. resolva One–ORP para l (retorna novo w_l)
4. calcule $novocusto$ de roteamento
5. se $novocusto < custo$: $custo \leftarrow novocusto$ e volte para o **Passo 2**
senão $L \leftarrow L \setminus \{l\}$
6. se $L = \varnothing$: **FIM**
senão volte para o **Passo 2**

4. Análise de Desempenho

Os testes foram executados em oito topologias de rede diferentes, no entanto apresentaremos somente os resultados de 3 dessas redes. Em cada topologia foram especificadas dez matrizes de tráfego (dez cenários distintos) com diferentes intensidades de carga. Consideramos o cenário 0 como sendo de baixa intensidade de carga, enquanto o cenário 9 é o de máxima intensidade de carga.

Os parâmetros de comparação foram: número médio de pacotes nas filas dos roteadores $\sum_{ij} D_{ij}(F_{ij})$ (custo de roteamento), enlace de máxima utilização da rede e tempo de execução do algoritmo.

Os problemas ORP-Delay e ORP-Utilization foram modelados em AMPL – linguagem que permite a modelagem de problemas de otimização em larga-escala ou de programação matemática. Tal linguagem é compatível com os principais softwares de otimização conhecidos comercialmente e na comunidade acadêmica. Para facilitar a geração dos modelos desenvolvemos uma ferramenta em linguagem C onde a topologia da rede e a matriz de tráfego são os dados de entrada, e a saída é um arquivo com o problema modelado em AMPL.

Na resolução dos modelos ORP-Delay e ORP-Utilization utilizamos o software de otimização KNITRO no NEOS Server [Czyzyk et al. 1998]. O KNITRO é uma ferramenta desenvolvida para resolver problemas de otimização de grande escala.

As ferramentas $1/C$ (padrão recomendado pelos fabricantes), NrHops (solução número de saltos), HeuRoSa e HeurOne foram desenvolvidas em linguagem C padrão. Enquanto a ferramenta HeurCoupling foi desenvolvida em linguagem C juntamente com o pacote de otimização LP-Solve [Berkelaar 1996].

As figuras 3, 5 e 7 representam o custo de roteamento $\sum_{ij} D_{ij}(F_{ij})$ das redes de 16, 20, e 30 nós respectivamente. Nesses gráficos podemos reparar que para cenários de baixa carga (0 ao 3) o desempenho das soluções comparadas são praticamente iguais em todas as redes. As figuras 4, 6 e 8 representam os resultados dos enlaces de maior utilização das redes de 16, 20, e 30 nós respectivamente.

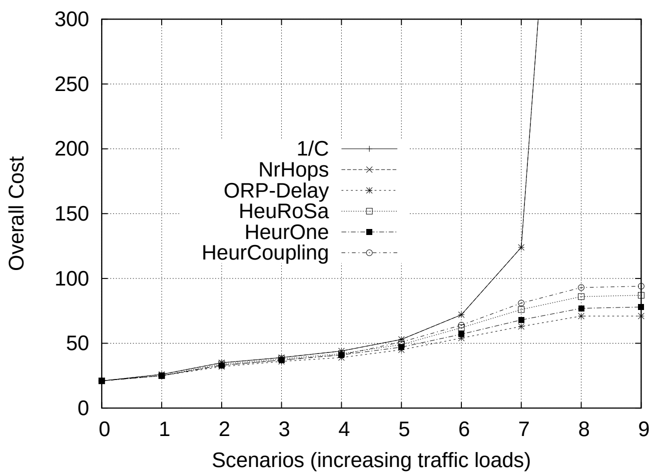

Figure 3. Custo de Roteamento – Rede de 16 nós

Na rede de 16 nós, cujos resultados de custo de roteamento são apresentados na figura 3, observamos que a partir do cenário 6 as propostas NrHops e $1/C$ passam a ter custos bastante elevados. As demais soluções acompanham de perto os resultados do roteamento ótimo (ORP-Delay). A proposta HeurOne é em média 6% pior que a solução ótima e 603% melhor que o padrão $1/C$. As propostas HeuRoSa e HeurCoupling são em média 13% e 18% piores em relação ao ótimo e 556% e 530% melhores em relação ao padrão $1/C$, respectivamente.

Na figura 4 temos os resultados de enlace com máxima utilização na rede de 16 nós. Nesse parâmetro de comparação, a proposta HeurOne é em média 1% pior que a solução ótima e 16% melhor que o padrão $1/C$. As propostas HeuRoSa e HeurCoupling são em média 6% piores em relação ao ótimo e 10% melhores em relação ao padrão $1/C$.

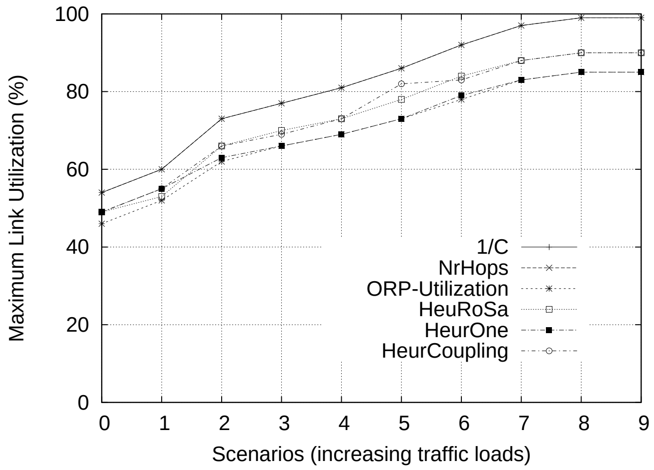

Figure 4. Enlace com Máxima Utilização – Rede de 16 nós

Na rede de 20 nós, cujos resultados são apresentados na figura 5, observamos que a partir dos cenários 1 e 5 as propostas NrHops e $1/C$ passam a ter custos de roteamento exacerbados se comparados com as demais propostas. As propostas HeurCoupling e HeurOne são em média 8% piores em relaçao ao ótimo e 196% melhores em relação ao padrão $1/C$. Enquanto a proposta HeuRoSa está é em média 16% pior em relação ao ótimo e 175% melhor em relação ao padrão $1/C$.

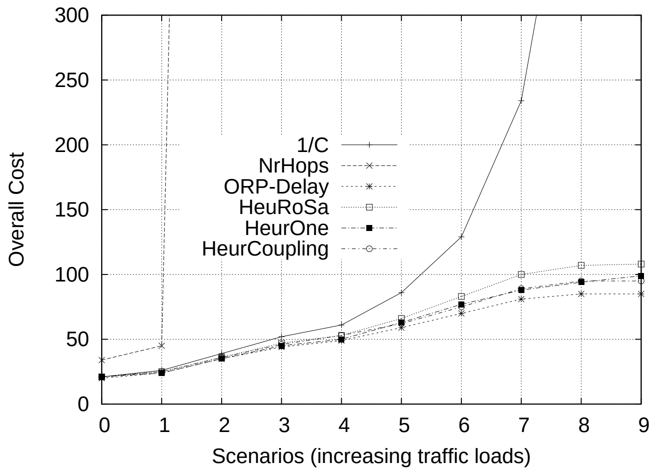

Figure 5. Custo de Roteamento – Rede de 20 nós

Podemos verificar na figura 6 os resultados de enlace com máxima utilização da rede de 20 nós. Nesse parâmetro de comparação, a proposta HeurOne é em média 26% pior que a solução ótima e 21% melhor que o padrão $1/C$. As propostas HeuRoSa e HeurCoupling são em média a 33% e 31% piores em relação ao ótimo e 15% e 17% melhores em relação ao padrão $1/C$, respectivamente.

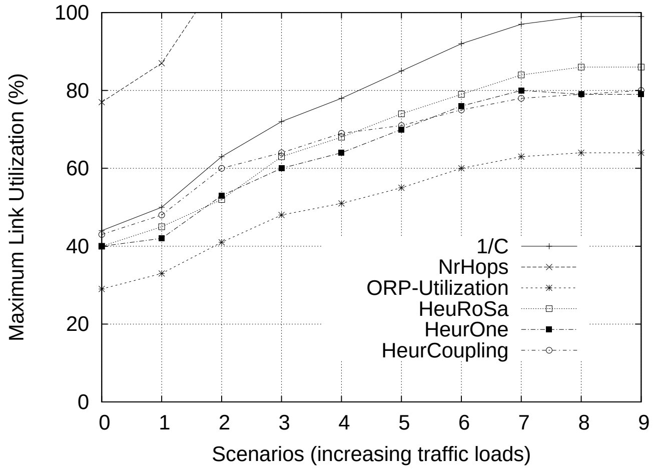

Figure 6. Enlace com Máxima Utilização – Rede de 20 nós

Na rede de 30 nós, cujos resultados são apresentados na figura 7, podemos observar que as propostas NrHops e $1/C$ extrapolam o custo de roteamento a partir do cenário 6. As demais propostas são em média 7% piores em relação ao roteamento ótimo e 340% melhores que o padrão.

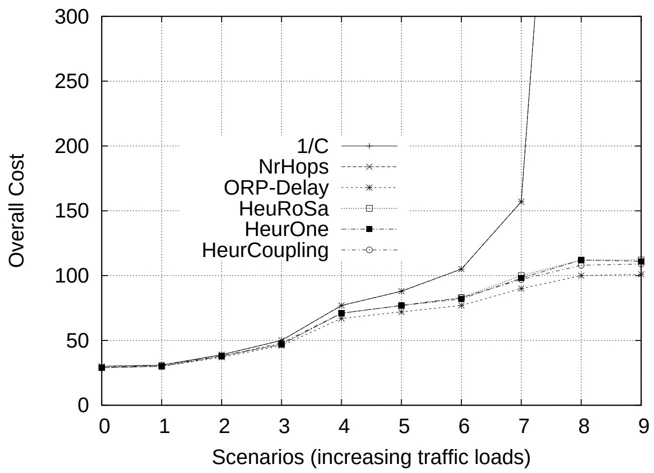

Figure 7. Custo de Roteamento – Rede de 30 nós

Na figura 8 temos os resultados de enlace com máxima utilização na rede de 30 nós. Nesse parâmetro de comparação, as propostas HeurCoupling e HeurOne são em média 4% piores que a solução ótima e 7% melhores que o padrão. A proposta HeuRoSa é em média 6% pior em relação ao ótimo e 5% melhor em relação ao padrão.

Na tabela da figura 9 observamos o tempo médio que cada algoritmo levou para resolver os 10 cenários de cada rede de comunicação. Como verificamos anteriormente a proposta HeurOne apresentou um desempenho melhor nos parâmetros custo de roteamento e enlace com máxima utilização da rede. No entanto, no parâmetro tempo de execução essa proposta tem a maior média dentre as três propostas heurísticas. Tais resultados comprovam a capacidade que tal algoritmo tem de “driblar” os mínimos locais em busca do mínimo global, levando mais tempo para se aproximar do ótimo.

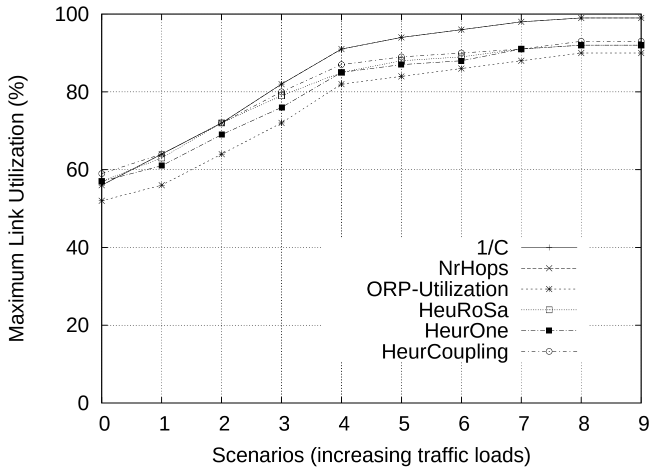

Figure 8. Enlace com Máxima Utilização – Rede de 30 nós

	ORP-Delay	HeuRoSa	HeurOne	HeurCoupling
Rede de 16 nós	2,651 seg	0,285 seg	2,577 seg	0,099 seg
Rede de 20 nós	1200,321 seg	1,659 seg	183,521 seg	0,134 seg
Rede de 30 nós	1589,356 seg	704,222 seg	1504,222 seg	0,309 seg

Figure 9. Tempo Médio de Execução dos Algoritmos Propostos

A proposta HeuRoSa demonstra os resultados mais interessantes. Observamos resultados mais equilibrados nos três parâmetros analisados, melhor custo-benefício. A proposta HeuCoupling apresenta excelentes resultados em termos de tempo de execução. No entanto, essa proposta não aparenta ter desempenho equivalente às outras duas no parâmetros custo de roteamento e enlace com máxima utilização da rede.

5. Conclusão

Nesse artigo estudamos os problemas de roteamento ótimo e suas aplicações para a engenharia de tráfego. Foram propostas três novas heurísticas para a solução do problema do roteamento ótimo de menor caminho: HeurCoupling, HeuRoSa e HeurOne. Os pesos retornados pelas heurísticas possibilitam aos algoritmos de roteamento tradicionais (p.ex. OSPF) realizarem de forma eficiente o balanceamento de tráfego na rede, a otimização do uso dos recursos, e o condicionamento da rede para suportar aumentos repentinos no tráfego de entrada.

Os resultados numéricos confirmaram um excelente desempenho de nossas propostas quando comparadas à recomendação dos fabricantes de roteadores e a proximidade da solução ideal do roteamento ótimo com bifurcação.

Em trabalhos futuros pretendemos integrar os algoritmos propostos em uma *toolbox* a fim de criar um pacote específico para a engenharia de tráfego em redes IP. Tal pacote poderá ser utilizado pelos administradores de redes como ferramenta de suporte aos procedimentos de otimização do desempenho das redes de comunicação.

References

Awduche, D. (1999). Requirements for traffic engineering over mpls. In *RFC 2702*. IETF Network–WG.

Awduche, D. (2002). Overview and principles of internet traffic engineering. In *RFC 3272*. IETF Network–WG.

Berkelaar, M. (1996). Lp solve: Linear programming code. In *http://www.cs.sunysb.edu/algorith/implement/lpsolve/implement.shtml*.

Bertsekas, D. and Gallager, R. (1992). *Data Networks*. Prentice–Hall, 2nd edition.

Czyzyk, J., Mesnier, M. P., and More, J. J. (1998). The neos server. In *IEEE Computational Science and Engineering*, volume 5, pages 68–75.

Fortz, B. and Thorup, M. (2000). Internet traffic engineering by optimizing ospf weigths. In *Proc. of the 19th IEEE Conference on Computer Communications*, volume 20, pages 519–528. INFOCOM00.

Fortz, B. and Thorup, M. (2002). Traffic engineering with traditional ip routing protocols. In *IEEE Communications Magazine*, volume 40, pages 118–124.

Harmatos, J. (2001). A heuristic algorithm for solving the static weight assignment optimisation problem in ospf networks. In *Proceedings of Global Internet Conference*.

Kleinrock, L. (1993). On the modeling and analysis of computer networks. In *Proc.of the IEEE*, volume 81, pages 1179–1191.

Mahajan, R., Spring, N., Wetherall, D., and Aderson, T. (2002). Inferring link weights using end–to–end measurements. In *Proc. of ACM SIGCOMM – Internet Measurement Workshop*.

Moy, J. (1991). Ospf version 2. In *RFC 1247*. IETF Network–WG.

Ramakrishnan, K. G. and Rodrigues, M. A. (2001). Optimal routing in shortest–path data networks. In *Bell Labs Technical Journal*, volume 6, pages 117–138.

Rosen, E., Viswanathan, A., and Callon, R. (2001). Multiprotocol label switching architecture. In *RFC 3031*. IETF Network–WG.

Srinivasan, C., Viswanathan, A., and Nadeau, T. (2004). Multiprotocol label switching (mpls) traffic engineering (te) management information base (mib). In *RFC 3812*. IETF Network–WG.

Thomas, T. M. (1998). *OSPF Network Design Solutions*. Cisco Press.

Wang, Y. and Wang, Z. (1999). Explicit routing algorithms for internet traffic engineering. In *Proc. of the 8th Computer Communications and Networks*.