OTIMIZAÇÃO DE TRÁFEGO EM REDES IP UTILIZANDO MPLS*

Huds Sousa Costa[1]
Marcelo Lisboa Rocha[2]

RESUMO: Neste trabalho é apresentada uma técnica de otimização para o roteamento de tráfego em redes IP sobre MPLS. Aqui, a tarefa de roteamento de tráfego foi modelada como um problema de programação matemática e foi proposta uma heurística de colônia de formigas para resolver aproximadamente este difícil problema computacional. Os resultados experimentais demonstram que o algoritmo de roteamento baseado na heurística de colônia de formigas proposta apresenta melhor desempenho que os algoritmos padrões, com respeito à qualidade das soluções encontradas e a otimização de tráfego em redes IP utilizando MPLS.

Palavras-Chaves: Otimização. Roteamento em redes IP. MPLS.

ABSTRACT: In this work is presented an optimization technique to traffic routing on IP networks over MPLS. Here, the traffic routing task was modeled as a mathematical programming problem and was proposed an ant colony heuristic to obtain near-optimal solutions to this computational hard problem. Experimental results show that the routing algorithm base on the proposed ant colony heuristic presents better performance than the standard algorithms, with respect to the quality of solutions found and IP networks over MPLS.

Keywords: Optimization. Routing in IP networks. MPLS.

1 INTRODUÇÃO

A importância das redes de computadores em várias áreas vem aumentando progressivamente com o passar dos anos, tanto em número de usuários como na quantidade de aplicações e tipos diferentes de serviços que as utilizam. Grande parte destes serviços são baseados no protocolo IP. Para suprir estas necessidades foi desenvolvida uma aplicação que suporta múltiplos protocolos. O MPLS (*Multiprotocol Label Switching* – Comutação de Rótulos Multiprotocolo) é uma tecnologia capaz de viabilizar múltiplos serviços de rede sobre uma infraestrutura compartilhada, permitindo o aprovisionamento rápido de serviços e tornando-se um ponto de concentração para serviços novos e antigos (OSBORNE; SIMHA, 2002).

[1] Graduado em Ciências da Computação (UNIRG). Professor assistente e analista de sistemas da Faculdade Serra da Mesa. Email: hudsc@hotmail.com.
[2] Graduado em Ciências da Computação (UCP). Mestrado em Computação (UFF/RJ). Mestrado em Engenharia Elétrica (UFRJ). Doutorado em Engenharia Elétrica (UFRJ). Professor titular da Faculdade Serra da Mesa dentre outras. Email: marcelolisboarocha@gmail.com

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

Embora somente nesta década o MPLS começa a ser implantado nas redes de computadores para a criação de novos serviços, esta tecnologia não é recente. Iniciativas anteriores de protocolos baseados em métodos de engenharia de tráfego já utilizam essa tecnologia desde meados dos anos de 1990 (METRORED, 2007). A IETF (*Internet Engineering Task Force*) define-o como uma tecnologia de chaveamento de pacotes que proporciona o encaminhamento e a comutação eficiente de fluxos de tráfego através da rede, apresentando-se como uma solução para diminuir o processamento nos equipamentos de rede e interligar com maior eficiência redes de tecnologias distintas.

O termo *Multiprotocol* significa que esta tecnologia pode ser usada sob qualquer protocolo de rede (OSBORNE; SIMHA, 2002). Considerando a Internet e a importância de seus protocolos nas várias WAN's públicas e privadas, tem-se aplicado o estudo e a implementação do MPLS basicamente para redes IP. O protocolo MPLS abre um campo promissor para o roteamento baseado em QoS (WANG, 1999), onde a seleção dos caminhos/fluxos pode estar sujeita a restrições de QoS, expressas em termos de métricas como: largura de banda mínima, atraso e variação de atraso máximos e taxa de perda máxima de pacotes.

Sob a ótica da otimização, o problema de encontrar rotas que satisfaçam a esses requisitos pertence à categoria NP-difícil, o que significa que possui ordem de complexidade combinatória. Em outras palavras, o esforço computacional para a sua resolução cresce exponencialmente com o tamanho do problema, dado pelo número de nós.

Na prática, para problemas deste tipo raramente os resultados ótimos são encontrados, existem vários problemas classificados nesta classe NP-difícil, tais como: caixeiro viajando, roteamento de veículos, alocação, onde métodos de solução aplicados a instâncias reais são, em geral, heurísticos, isto é, tais métodos não asseguram que a solução final obtida seja a melhor existente. Dentre esses métodos heurísticos destacam-se as metaheurísticas, as quais, ao contrário das heurísticas convencionais, são providas de mecanismos para escapar de ótimos locais, além de serem de fácil implementação e permitirem incluir, com facilidade, múltiplas restrições no roteamento. (FREDERICO; JAMILSON, 2004).

A metaheurística baseada em otimização de colônia de formigas (ACO-*Ant Colony Optimization*) será utilizada para otimizar o tráfego em rede IP sobre MPLS.

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

A escolha do método se dá pela importância que este vem tendo nos problemas de otimização, pela sua capacidade adaptativa e pela necessidade de se ter um método que resolva o problema de forma eficiente.

Este artigo apresenta a solução de um problema de otimização de tráfego em uma topologia de rede similar àquelas encontradas em *backbones* de provedores de serviços de acesso à Internet. A solução consiste em minimizar o atraso fim-a-fim de todos os fluxos de dados que atravessam os enlaces da rede, restritos pelas larguras de banda respectivas. Como resultado, o congestionamento de forma global na rede é reduzido. As soluções do problema de otimização tenderão a oferecer os caminhos com menores atrasos de transmissão.

As demais seções do artigo são apresentadas a seguir. A seção 2 apresenta o problema de roteamento em questão e sua formulação matemática. A seção 3 descreve a abordagem de solução para o problema utilizando a heurística ACO. Resultados computacionais são apresentados e discutidos na seção 4, enquanto a última seção conclui o trabalho.

2 DESCRIÇÃO DO PROBLEMA

A topologia física da rede pode ser modelada como um grafo conexo $G = (V, E)$ direcionado, V é o conjunto de vértices do grafo, que em uma rede podem ser *hubs*, *switches*, roteadores, satélites, rádios-bases, entre outros; e E é o conjunto de arestas do grafo, que representam as possíveis conexões (*links*) entre os nós da rede. Ressalta-se que cada aresta ligando os vértices i e j possui dois valores associados: A_{ij} que é o atraso e C_{ij} que é a capacidade/banda da respectiva aresta.

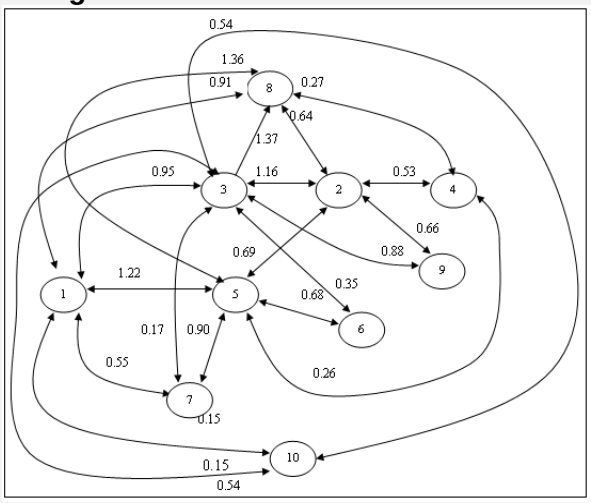
O objetivo é encontrar rotas otimizadas com o menor atraso possível dentre as várias existentes de forma a contornar as falhas existentes.

A Figura 1 apresenta um modelo de Grafo conexo proposto para o problema, onde este denota uma estrutura de redes de computadores aqui identificada como *inst1*. Esta possui 10 vértices e 20 arestas com capacidade máxima OC48, que é a banda passante para cada *link*, e um atraso (rótulo da aresta).

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

Como neste problema existem várias fontes e seus respectivos destinos com fluxos a serem roteados (cada qual com uma necessidade mínima de banda), o mesmo pode ser visto como um problema de multifluxos.

Figura 1: Instância *Inst1* de referência.



Fonte: Próprio autor

Ressalta-se que o problema de multifluxos, quando há a restrição de integralidade dos fluxos, se torna NP-difícil (EVENT et al., 1976) mostra que o problema de decisão com apenas 2 fluxos é NP-completo). Assim sendo, o problema pode ser modelado como o problema de programação matemática descrito a seguir. **Problema:** otimização do tráfego em rede IP sobre MPLS (*G, c, r*): Dados um Grafo orientado G = (V, E), atraso e banda disponível nas arestas $c : E \rightarrow R_+$, encontrar fluxos/rotas otimizadas com menor atraso satisfazendo o limite mínimo de banda passante.

Minimize

$$\sum_{i=1}^{k} \sum_{j=1}^{n} \sum_{z \in v_j} A_{jz} y_{jzi} + A_{zj} y_{zji} \quad (1)$$

Sujeito a

$$\sum_{z \in v_j} x_{zji} - \sum_{z \in v_j} x_{jzi} = \begin{cases} 0, se\ i\ for\ nó\ de\ passagem \\ -B_i, se\ for\ nó\ de\ origem, \forall_{i,j} \\ B_i, se\ j\ for\ nó\ de\ destino \end{cases} \quad (2)$$

$$\sum_{z \in v_j} x_{zji} \leq C_{zj}\ ,\ \forall_{i,j} \quad (3)$$

$$x_{zji} - M y_{zji} \leq 0\ ,\ \forall_{i,j,z} \quad (4)$$

$$y_{jzi} \in \{0,1\}, \forall_{i,j,z}$$

Onde:

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

k = nº de fluxos e n = nº de vértices.

V_j = Conjunto de vértices vizinhos/adjacentes ao vértice j.

B_i = banda necessária para rotear o fluxo i.

A_{jz} = Atraso da aresta j-z.

y_{jzi} = indica se a aresta j-z está sendo (1) ou não (0) utilizada no fluxo i.

x_{zji} = Quantidade de fluxo que passa pelo link/atesta z-j no fluxo i.

C_{zj} = Capacidade do link/aresta z-j.

M = Valor numérico grande.

Na seqüência se tem a explicação de cada item da formulação matemática do respectivo problema, onde:

(1) Minimizar a soma dos atrasos das aresta/links utilizados em todos os fluxos

(2) Restrição de equilíbrio de fluxo, onde:

- O fluxo resultante deve ser igual a zero (0) nos nós de passagem.
- O fluxo resultante deve ser igual a B_i nos nós de destino i.
- O fluxo resultante deve ser igual a $-B_i$ nos nós de origem i.

(3) Restrição de capacidade da aresta/link: a soma de todos os fluxos que passam pela aresta/link z-i deve ser menor ou igual a capacidade C_{zi} da respectiva aresta.

(4) Restrição de utilização: indica que se passou algum fluxo pela aresta z-j no caminho i a aresta j-z está sendo utilizada no fluxo i (y_{jzi}).

3 A HEURÍSTICA ACO PROPOSTA

Este trabalho aplica a metaheurística da colônia de formigas proposta por Dorigo e Caro (1999) para a solução de problemas de otimização combinatória, como o problema do caixeiro viajante.

A metaheurística da colônia de formigas foi inspirada na observação das colônias de formigas reais, em particular em como elas encontram o menor caminho entre a fonte de alimentos e o formigueiro.

Para a obtenção do alimento para o formigueiro, a colônia resolve um interessante problema de otimização. Inicialmente, as formigas percorrem de modo aleatório a região próxima ao formigueiro em busca do alimento. Cada formiga,

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

enquanto percorre o seu caminho, deposita sobre o solo uma substância chamada feromônio, formando um caminho ou rastro de feromônio. As formigas subsequentes detectam o a presença desta substância e tendem a escolher o caminho marcado com a maior concentração de feromônio. O feromônio, portanto, além de possibilitar a formação de um caminho de volta para a formiga, também tem a função de informar as outras formigas sobre quais os melhores caminhos até o alimento. Depois de algum tempo, os caminhos mais eficientes – ou de menor distância percorrida até o alimento – acumulam uma quantidade maior de feromônio. Inversamente, os caminhos menos eficientes – ou de maior distância percorrida até o alimento – apresentam uma pequena concentração de feromônio, devido ao menor número de formigas que passaram por ele e ao processo de evaporação natural do feromônio. No problema de otimização que o formigueiro se defronta, cada formiga é capaz de construir uma solução completa do problema; contudo, a melhor solução só é obtida mediante cruzamento das diversas soluções encontradas.

O Sistema de Formigas, primeira metaheurística de otimização de colônia de formigas proposta por Dorigo (1992), quando aplicado ao problema do caixeiro viajante, inicia-se com cada formiga construindo uma solução a partir de um dos nós da rede do problema. Cada formiga *k* constrói o seu caminho movendo-se através de uma seqüência de locais vizinhos, onde os movimentos são selecionados segundo uma distribuição de probabilidades dada por:

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} . [n_{ij}]^{\beta}}{\sum [\tau_{ik}(t)]^{\alpha} . [n_{ik}]^{\beta}} \\ 0, \text{outros casos} \end{cases} \qquad (5)$$

Na equação (5) $P_{ij}^{k}(t)$ é a probabilidade da formiga *k*, que se encontra em *i*, escolher o nó *j* como próximo nó a ser visitado no tempo (iteração) *t*, os valores dos parâmetros α e β são ajustados heuristicamente. A variável α é a ponderação do feromônio ($0 \leq \alpha \leq 1$) e β é a ponderação da informação heurística ($0 \leq \beta \leq 1$), n ij=1/dij é a visibilidade entre a variável j a variável i e vice-versa; dij é a distância Euclidiana entre i e j; τ ij é a intensidade da trilha no caminho (i,j) no tempo *t* (quando *t*=0 a

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

intensidade da trilha é gerada aleatoriamente, geralmente com distribuição uniforme). Ao longo da trilha de *i* até *j* a formiga deposita na trilha uma substância (feromônio) definida por:

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k} \end{cases} \qquad (6)$$

$\Delta\tau_{ij}^{k}$ possui este valor se (i, j) pertence a rota construída pela melhor formiga da k-ésima iteração e 0 caso contrário (usar Q = 1) e L_k é o comprimento da *tour* da *k*-ésima formiga. Após um certo número *m* de formigas terem finalizado suas rotas, a quantidade de feromônio é atualizada em cada arco de modo a reforçar o caminho obtido pela melhor formiga. Assim, para cada arco (*i*,*j*) da rede, adiciona-se uma quantidade de feromônio proporcional ao tamanho da rota obtida:

$$\tau_{ij}^{k}(t) = (1-\rho).\tau_{ij}^{k}(t-1) + \rho.\Delta\tau_{ij}^{k} \qquad (7)$$

O primeiro termo da equação (7) é responsável pela evaporação do feromônio, onde ρ é um parâmetro que determina a velocidade da evaporação. O segundo termo é responsável por aumentar a concentração de feromônio apenas nos arcos visitados pela melhor formiga.

A estratégia de solução proposta neste trabalho para o problema de otimização de tráfego em redes IP sobe MPLS é ligeiramente diferente da descrita anteriormente para o problema do caixeiro viajante. No caso do problema tratado neste trabalho, para obtenção de soluções aproximadas eficientes, a ordem de alocação dos fluxos influencia no resultado final. De modo a este problema poder ser resolvido de forma semelhante ao problema do caixeiro viajante, foi atribuído um número a cada fluxo (cidade) e o que equivaleria a distância entre as cidades é o valor do atraso calculado entre a alocação de um fluxo e outro dependendo da ordem. Assim sendo, a única diferença básica é a não necessidade de calcular a distância (atraso) da última cidade (fluxo) para a primeira.

4 RESULTADOS COMPUTACIONAIS

Serão abordados nesta seção, os testes e resultados computacionais realizados sobre o problema proposto. A seguir, serão apresentadas as instâncias

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

utilizadas para testes e o desempenho do algoritmo ACO tanto sobre a qualidade das soluções obtidas como o tempo de execução, será realizado análises e comparações destes como os métodos de solução B&B (*Branch-and-Bound*) e Caminho Mínimo (CM).

4.1 Geração das Instâncias

Houve a necessidade de desenvolver um programa com o propósito de gerar as instâncias utilizadas para coletar os resultados computacionais onde estas representam grafos conexos, simulando uma estrutura de redes de computadores. Cada instância gerada é um grafo que representa enlaces bidirecionais e as arestas têm capacidade de 622 (OC12) ou 2488 (OC48) unidades de banda, atribuídas aleatoriamente, com a mesma probabilidade, como também um atraso gerado aleatoriamente entre 0.01 e 2.0.

A Tabela 1 apresenta as instâncias utilizadas para os testes computacionais geradas. Estas possuem quantidades de vértices que variam de 10 a 63, e arestas que variam de 28 a 978. Na Tabela 1 também são especificados os números de caminhos para cada instância.

Tabela 1: Instâncias utilizadas.

Instâncias	Nº de Vértices	Nº de Arestas	Nº de Caminhos
inst1	10	20	3
inst2	15	45	2
inst3	20	89	4
inst4	26	153	5
inst5	30	209	3
inst6	35	286	3
inst7	40	372	2
inst8	45	490	4
inst9	47	540	3
inst10	49	587	2
inst11	53	698	5
inst12	56	779	4
inst13	63	978	3

Fonte: Próprio autor

 ISSN 2238-9547

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

Na Tabela 1 estão todas as instâncias utilizadas para os testes computacionais, os vértices referente a cada uma, as arestas e os caminhos percorrido em cada instância.

4.2 Ambiente Computacional Utilizado

Os testes computacionais foram realizados em um microcomputador com a configuração especificada a seguir, para todas as instâncias sobre os algoritmos analisados: Processador *Pentium* 4 de 2.8 GHz com e 1.00 GB de memória RAM. O sistema operacional utilizado foi o Windows XP, e todos os programas foram codificados na linguagem “C”.

Na seção a seguir serão apresentados, os resultados das soluções (soma dos atrasos dos caminhos) das instâncias analisadas bem como os tempos computacionais (em Segundos de CPU) apresentados pelos algoritmos CM, B&B e ACO sobre estas.

4.3 Testes Computacionais

Nesta Subseção, serão feitas as análises e comparações dos resultados do método proposto com outros métodos clássicos, tais como CM e B&B.

4.3.1 Análises dos Resultados

Para a execução desta análise, foi considerada a utilização de 3 (*três*) caminhos ou rotas. Cada um dos 3 caminhos possui a seguinte configuração: o primeiro caminho tendo origem 9 e destino 10, o segundo possuindo origem 1 e destino 4, já o terceiro tem origem 6 e destino 8, sendo todos com necessidade de banda passante de 1680 Mbit/s.

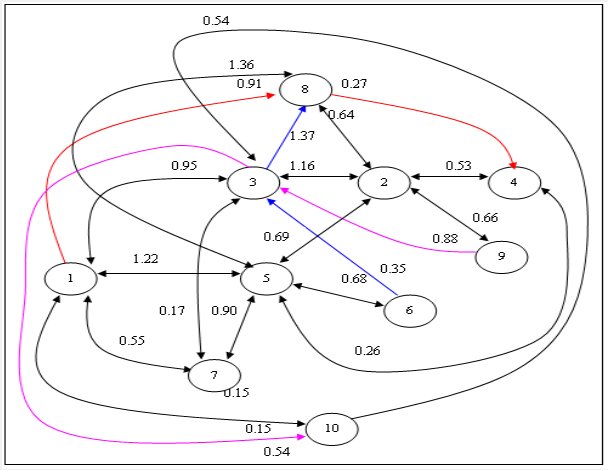
A Figura 2 apresenta a topologia de rede gerada após a execução do Algoritmo Caminho Mínimo. Para completar, as três rotas utilizam a seguinte seqüência de vértices intermediários:

- Caminho1: 9–3–10 – trajeto de cor rosa.
- Caminho2: 1–8–4 – trajeto de cor vermelha.
- Caminho3: 6–3–8 – trajeto de cor azul.

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

A soma do atraso dos caminhos percorrido pelos três rotas resulta em 4.32

Figura 2: Solução do CM na instância *Inst1*.



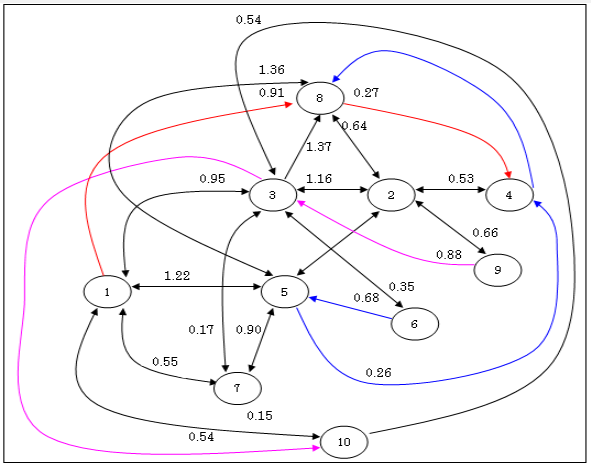
Fonte: Próprio autor

Aplicando agora o método B&B à instância *Inst1* para os 3 caminhos já citados, tem-se a solução ilustrada na Figura 3. A seguir, são apresentados todos os vértices e arestas pertencentes aos 3 caminhos.

- Caminho1: 9–3–10 – trajeto de cor rosa.
- Caminho2: 1–8–4 –trajeto de cor vermelha.
- Caminho3: 6–5–4–8 –trajeto de cor azul.

A soma do atraso dos caminhos percorrido pelos três rotas resulta em 3.81.

Figura 3: Solução do B&B na instância *Inst1*.



Fonte: Próprio autor

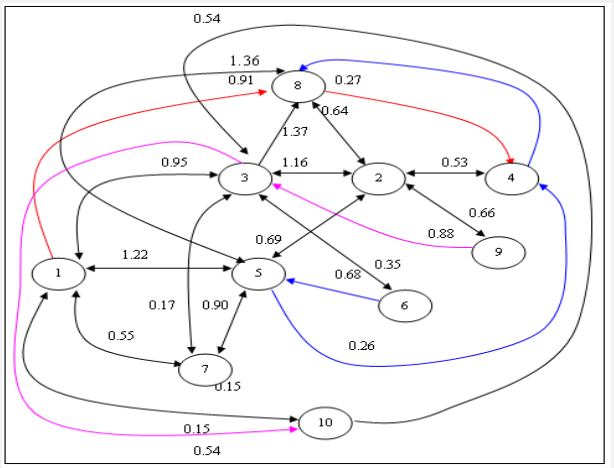
Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

Agora será apresentada a solução do ACO, na Figura 4, para a instância Inst1 com 3 caminhos já especificados. A seguir, são apresentados os vértices e as respectivas arestas de cada um dos caminhos encontrados:

- Caminho1: 9–3–10 – trajeto de cor rosa.
- Caminho2: 1–8–4 – trajeto de cor vermelha.
- Caminho3: 6–5–4–8 – trajeto de cor azul.

A soma do atraso dos caminhos percorrido pelas três rotas resulta em 3.81.

Figura 4: Solução do ACO na instância *Inst1*.



Fonte: Próprio autor

4.3.2 Comparação dos Resultados

Nesta subseção, serão apresentados e comparados os resultados computacionais das 3 técnicas aqui implementadas.

Na Tabela 2 estão as soluções (soma dos respectivos atrasos de todos os caminhos a serem roteados) dos métodos CM / B&B / ACO e o tempos computacionais em segundos de CPU. Deve-se citar que neste trabalho, devido ao alto tempo de execução do método B&B para o problema em questão, o tempo máximo de execução do mesmo foi limitado ou ao tempo de execução do ACO para a respectiva instância ou até a obtenção da primeira solução válida. Vale ressaltar que os tempos computacionais são diretamente proporcionais ao tamanho das instâncias o que influência diretamente nos resultados, como descrito na próxima subseção.

De acordo com os dados apresentados na Tabela 2, pode-se observar:

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

- método do caminho mínimo (CM) é o mais rápido, possuindo baixíssimo tempo computacional. Por outro lado, é o que apresenta as soluções de pior qualidade. Isto pois, do ponto de vista de QoS, nem sempre o caminho mais curto é o caminho que apresenta o melhor conjunto de recursos necessários a uma aplicação.
- método branch-and-bound (B&B) apresentou soluções de boa qualidade. Contudo, a um custo computacional (tempo de execução) geralmente bem mais alto que os métodos do CM e ACO. Isto se deve a alta complexidade do problema em questão.
- método de colônia de formigas (ACO), comparado aos outros dois (CM e B&B) é o que apresenta a melhor relação custo-benefício. Isto se deve ao fato de apresentar as melhores soluções com custo computacional (tempo de execução) razoável, mesmo para as instâncias de maiores dimensões.

Tabela 2: Soluções e tempo de execução dos métodos CM / B&B / ACO.

Instância	Sol. CM	Sol. B&B	Sol. ACO	Tempo CM	Tempo B&B	Tempo ACO
inst1	4.32	3.81	3.81	0.01	10.20	10.12
inst2	2.41	2.06	1.23	0.01	19.03	19.05
inst3	3.38	4.41	2.54	0.01	19.10	19.01
inst4	4.19	4.40	4.06	0.01	21.50	20.81
inst5	1.19	1.19	1.13	0.01	30.96	30.27
inst6	1.62	1.61	1.33	0.14	50.12	45.31
inst7	0.80	0.59	0.59	0.14	52.32	50.16
inst8	2.52	3.51	1.68	0.15	59.11	54.90
inst9	1.51	1.25	1.00	0.16	60.01	59.09
inst10	0.91	0.91	0.91	0.18	63.45	41.59
inst11	1.75	2.13	1.87	0.21	65.78	50.10
inst12	1.65	1.41	1.05	0.25	69.09	52.01
inst13	1.06	4.70	0.89	0.27	71.01	55.72

Fonte: Próprio autor

 ISSN 2238-9547

Revista Fasem Ciências | Vol. 1, n. 1, jan.-jun./2012

5 CONCLUSÕES E RECOMENDAÇÕES

Neste trabalho foi estudado o problema de otimização de roteamento em redes IP sobre MPLS, como contribuição deste trabalho, desenvolveu-se um algoritmo utilizando a meta-heurística colônia de formigas para o problema em questão. Embora simples de descrever e entender, o problema de otimização de roteamento em redes IP é de difícil resolução. Como visto anteriormente, este problema está classificado como um problema NP-difícil o que limita o uso de técnicas exatas para encontrar a solução para instâncias grandes.

O algoritmo proposto foi testado em 13 instâncias geradas aleatoriamente de tamanho compatível com os existentes nos *backbones* das redes óticas das empresas de telecomunicações. O ACO proposto obteve um bom desempenho em termos de qualidade das soluções geradas, principalmente em instâncias de maiores dimensões. Outro aspecto relevante foram os tempos computacionais satisfatórios.

O objetivo deste trabalho é encontrar boas soluções num tempo computacional razoável. O objetivo foi plenamente alcançado com a aplicação da metaheurística ACO, portanto, conclui se que os resultados obtidos são animadores, boas soluções em uma quantidade de tempo computacional razoável.

Este problema foi projetado para operar de forma estática e em instâncias próximas a saturação do enlaces, tendo aplicação direta em planejamento de capacidade de “backbones” de operadoras de telecomunicações. Não foram definidas garantias rígidas de qualidade de serviços a serem oferecidas às aplicações. A garantia de que a capacidade de largura de banda dos enlaces não seja violada, entretanto, melhora de forma global a QoS da rede.

Apesar dos bons resultados obtidos pelo método de colônia de formigas (ACO) aqui implementado para o problema em questão, ainda existem atividades a serem realizadas que possibilitam a obtenção de resultados ainda melhores, tanto no tocante ao tempo computacional quanto à qualidade da solução. Assim, tem-se as seguintes recomendações:

- Para melhorar o desempenho do ACO, desenvolver versões paralelas do código desenvolvido, tanto para máquinas de memória compartilhada (vários núcleos) quanto para máquinas de memória distribuída (*clusters* ou *grids*).

- Utilizar outros métodos exatos, tais como *branch-and-cut* e *branch-and-price*, de modo a verificar a possibilidade de obtenção de tempos computacionais menores para as mesmas instâncias ou outras ainda maiores para comparação com o ACO.

REFERÊNCIAS

DORIGO, M.; CARO, G. DI. The Ant Colony Optimization Meta-Heuristic. In: **New Ideas in Optimization**, McGraw-Hill, 1999.

DORIGO, M. **Optimization, Learning and Natural Algorithms**. Ph.D.Thesis, Politecnico di Milano, Italy, 1992.

EVEN, S.; ITAI, A.; SHAMIR, A. **On the complexity of timetable and multicommodity flow problems.** SIAM Journal on Computing, 5(4):691–703, Dezembro 1976.

FREDERICO, C. M.; JAMILSON, M. F. S; SUMIKA, H. S. **Uma metodologia Heurística baseada em GRASP, VND e VNS Para a resolução do Problema de Dimensionamento de Redes IP.** XXXVI– SBPO, 2004.

METRORED. Disponível em: http://www.metrored.com.br/artigos/artigo_mpls_em_redes.php Acessado em: 22/03/2007

OSBORNE, E.; SIMHA, A. **Engenharia de tráfego com MPLS**. Ed. Campus, 2002.

WANG, Z. **On the complexity of Quality of Service Routing**. Information Processing Letters 69 (1999) 111-114. Elsevier Science, 1999.