

Desenho de Redes *Overlay* com Qualidade-de-serviço Utilizando Metaheurísticas

Sibelius Lellis Vieira

Resumo—A Internet ainda carece de um suporte adequado para aplicações de tempo-real com requisitos de qualidade-de-serviço. Em grande parte, isto é resultado do fato de que o tráfego fim-a-fim que atravessa os vários sistemas autônomos que compõem a rede requer um nível de cooperação entre estes que é difícil de se conseguir na arquitetura atual. Recentemente, as redes de serviço do tipo *overlay* têm sido consideradas como uma forma de obter a qualidade de serviço, evitando as dificuldades da estrutura da Internet. Neste trabalho, abordamos o problema de obtenção de uma topologia para uma rede *overlay* para a qual terminais (clientes) e nós provedores desta rede estão conectados através de provedores de serviço com garantias de banda. Expressamos este problema como um problema de otimização quadrático e investigamos como o uso de metaheurísticas pode auxiliar na obtenção de soluções próximas da solução ótima.

Palavras-Chave—Qualidade-de-Serviço, Redes *Overlay*, Análise de Redes.

Abstract—The Internet still lacks adequate support for QoS applications with real-time requirements. In great part, this is due to the fact that provisioning of end-to-end QoS to traffic that traverses multiple autonomous systems (ASs) requires a level of cooperation between ASs that is difficult to achieve in the current architecture. Recently, service overlay networks have been considered as an approach to QoS deployment that avoids these difficulties. In this study, we address the problem of the topological synthesis of a service overlay network, where endsystems and nodes of the overlay network (provider nodes) are connected through ISPs that supports bandwidth reservations. We express the topology design problem as an optimization problem. Even though the design problem is related to the (in general NP-hard) quadratic assignment problem, we are able to show that relatively simple heuristic algorithms can deliver results that are sometimes close to the optimal solution.

Keywords—Quality-of-service, Overlay Networks, Network Analysis.

I. INTRODUÇÃO

O suporte à qualidade de serviço (QoS) na Internet permanece uma tarefa desafiadora, apesar dos vários esforços recentes para melhorar o serviço básico de melhor esforço. Uma razão importante para esta dificuldade reside na própria estrutura da Internet, que é composta de um grande número de redes operadas independentemente (Sistemas autônomos ou SAs) [7], onde pontos de troca de tráfego fornecem a conexão dos sistemas autônomos na forma de uma estrutura única cooperativa [11]. A estrutura comercial de troca de tráfego torna o provisionamento de QoS fim-a-fim complicado. A maior parte dos contratos de troca de tráfego entre SAs são bilaterais e a QoS fim-a-fim é um esforço cooperativo de todos os SAs em um caminho fim-a-fim de um fluxo com garantias

de serviço. Embora um provedor de serviço (*Internet Service Provider* ou ISP) possa ter interesse no provisionamento de QoS dentro de seu sistema autônomo, não existe incentivo para dar o suporte a uma garantia similar para clientes de sistemas autônomos remotos [18].

Para solucionar estas questões, as redes *overlay* tem sido consideradas como um mecanismo de alto nível (acima da infraestrutura de nível de rede) que podem dar suporte a novos serviços para usuários sem requisitar mudanças na infraestrutura ou nas práticas de negócios [14]. Através destas redes, os serviços de rede têm sido propostos, abordando as necessidades das aplicações em tolerância à falhas [2], comunicação multicast [6], segurança [12], compartilhamento de arquivos [8] e QoS [18].

Neste sentido, consideramos um esquema no qual uma rede *overlay* de valor agregado que se localiza acima da infraestrutura dos ISP, denominada *rede provedora de QoS*, dá suporte à QoS fim-a-fim para os clientes. A rede provedora consiste de *nós provedores* e um conjunto de clientes, denominados *terminais*. Cada nó provedor e terminal tem potencialmente acesso à Internet através de um ou mais ISPs. Os nós provedores são conectados entre si e com os terminais através de ISPs. Dois nós provedores podem estabelecer um enlace na rede provedora se ambos estiverem associados ao mesmo ISP. De forma similar, um terminal pode acessar um dado nó provedor se ambos estão conectados ao mesmo ISP. Na Figura 1 ilustramos a relação entre os terminais, os nós provedores e os ISPs. A rede provedora adquire serviços dos ISPs, tais como uma banda garantida por enlace e de acordo com contratos pré-estabelecidos, fornece QoS para os terminais. Todos os enlaces seriam administrados pela rede provedora, que passa a oferecer o serviço para os terminais, mediante contrato.

Dada a conectividade entre os nós provedores e terminais ao conjunto de ISPs, como mostrado na Figura 1, o problema de desenhar uma topologia para a rede provedora consiste em atribuir a cada terminal um nó provedor e em atribuir conexões entre os nós provedores, de tal modo que todos os terminais possam trocar tráfego entre si, através de um caminho pela rede provedora. Como exemplo, na Figura 2, apresentamos uma topologia factível da rede provedora que corresponde a um conjunto de terminais e provedores interligados ilustrados na Figura 1.

Neste trabalho, apresentamos uma metodologia para prover o desenho topológico de uma rede *overlay* provedora. O propósito desta abordagem é estabelecer uma rede provedora com custo mínimo, levando em conta as interconexões dos nós e destes com os terminais. Formulamos o problema de busca desta topologia como um problema de otimização, que tem

Sibelius Lellis Vieira, Departamento de Computação, Universidade Católica de Goiás, Goiânia, Go, Brasil, E-mail: sibelius@ucg.br.

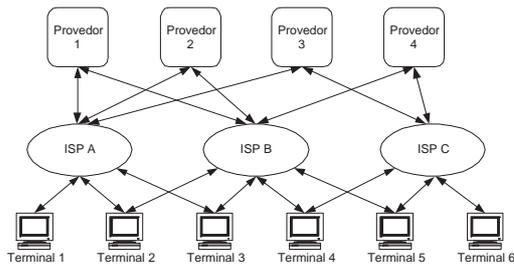


Fig. 1. Relação entre nós provedores, terminais e ISPs.

componentes lineares e quadráticos. Como tais problemas são, em geral, solúveis na prática apenas para pequenas instâncias ou casos especiais, nós investigamos o uso da metaheurística *Simulated Annealing* para encontrar boas soluções para o problema [5]. Também podemos mostrar que, em alguns casos especiais, soluções ótimas podem ser obtidas para redes maiores.

As redes *overlay* tem recebido uma grande atenção recentemente, uma vez que facilitam a implementação e disponibilização de novos serviços [2]. Vários modelos para *overlays* no nível de aplicação tem surgido, geralmente objetivando o fornecimento de serviços para aplicações específicas, tais como multicast [6], entrega de conteúdo [3] ou compartilhamento de arquivos [1]. Contudo, as questões relacionadas ao estabelecimento de topologia para redes de serviço tem recebido pouca atenção. Uma rede de serviço provê serviços genéricos que podem ser utilizados para uma variedade de aplicações [14]. Redes de serviço foram apresentadas como forma de prover serviços de valor agregado, incluindo QoS, baseados em requisições de usuário [7]. Por exemplo a abordagem OverQoS [18] propõe um serviço agregado baseado em infraestrutura de ISP que objetiva a garantia de serviços estatísticos. QUEST [9] é outra rede *overlay* que abrange o provisionamento de QoS e outros serviços. Uma rede comercial que é relativamente próxima do objeto deste trabalho é a rede Internap [10]. Uma diferença importante é que o acesso dos terminais é fornecido diretamente pela rede provedora, e não pelos ISPs.

Este artigo está estruturado de forma tal que na Seção II nós estabelecemos os parâmetros do problema de desenho da topologia e formulamos a síntese desta topologia como um problema de otimização. Na Seção III consideramos as condições para as quais o problema de otimização pode ser solucionado em tempo hábil. Na Seção IV apresentamos algoritmos heurísticos que podem resolver o problema de otimização para redes gerais. Na Seção V validamos nossos métodos em experimentos numéricos. Apresentamos as conclusões breves na Seção VI.

II. FORMULAÇÃO DO PROBLEMA DE DESENHO DA TOPOLOGIA

Nesta seção, formulamos o problema do desenho da topologia para a rede provedora em termos da solução de um problema de otimização. As entradas para este problema são os parâmetros da conectividade entre os terminais, nós provedores e os ISPs. Com esta informação, geramos a rede provedora,

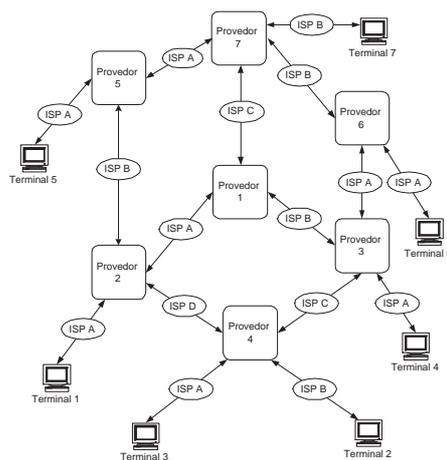


Fig. 2. Exemplo de solução para o problema de desenho da topologia.

conforme a Figura 2, de tal modo que a topologia resultante minimize uma dada métrica de custo que reflete o custo da rede provedora. Consideramos uma rede com M terminais e N nós provedores. Vamos nos referir ao terminal i como T_i e ao provedor j como PN_j .

Na rede provedora considerada aqui, cada terminal deve ser associado a exatamente um provedor. Um terminal pode estar associado a qualquer provedor com o qual compartilhe um ISP. Adotamos um custo constante α_{ij} para a reserva de uma unidade de banda (Mbps) entre o terminal T_i e o provedor PN_j . Este custo é denominado *custo de acesso*. Se não existe ISP comum a dois destes componentes, então T_i não pode ser atribuído à PN_j , e temos que $\alpha_{ij} = \infty$. Por outro lado, se T_i e PN_j podem ser conectados por mais de um ISP, então α_{ij} representa a conexão para o ISP com o custo mínimo. Portanto, α_{ij} implica na seleção dos ISP que conecta T_i e PN_j .

De forma equivalente, os nós provedores são também conectados entre si através de ISPs. Dizemos que existe um *enlace de transporte* entre dois provedores se ambos os nós tem pelo menos um ISP em comum. O custo de reservar uma unidade de banda entre PN_i e PN_j é l_{ij} . Se PN_i e PN_j não possuem ISP em comum teremos $l_{ij} = \infty$. Se dois provedores podem ser conectados por mais de um ISP, então l_{ij} é o custo através do ISP que minimiza este valor.

A rede provedora deve reservar banda nos enlaces de acesso e transporte para o tráfego entre os terminais. Supomos que a quantidade da banda reservada é dada por uma matriz de reserva $\Omega = \{\omega_{ij}\}$, onde ω_{ij} é a banda que é reservada para o tráfego de T_i para T_j , e naturalmente $\omega_{ii} = 0$. Estabelecemos que $\Omega_i = \sum_{j=1}^M \omega_{ij}$ denota a banda total reservada para o tráfego gerado em T_i .

Para obter a topologia da rede provedora, como dado na Figura 2, devemos resolver dois problemas. Primeiramente, para cada terminal, devemos selecionar um nó provedor para o qual atribuir este terminal. Depois, devemos selecionar os enlaces de transporte entre os nós provedores, de tal modo que qualquer provedor possa enviar o tráfego para qualquer outro, através de alguma rota. O custo total da rede provedora é o custo unitário dos enlaces de acesso e de transporte,

ponderados pela reserva de banda neste enlaces. O objetivo é determinar a topologia da rede provedora que minimize tal custo.

A construção da topologia da rede provedora é feita em 3 passos. No primeiro, consideramos apenas os nós provedores e seus enlaces de transporte e determinamos a rota entre pares de provedores, de modo a minimizar o custo total do transporte. Estas rotas são determinados de forma independente das atribuições dos terminais e da banda reservada. Dados PN_n e PN_m , o custo de transporte entre estes nós provedores é minimizado se o tráfego for enviado no caminho de menor custo conectando estes nós. Portanto, um enlace de transporte com custo l_{ij} é parte da topologia da rede provedora se este enlace está contido em alguma rota de menor custo entre qualquer par de provedores [16]. Denotando por r_{nm} a rota de menor custo entre PN_n e PN_m , e $(ij) \in r_{nm}$ se o enlace de transporte entre PN_i e PN_j for parte desta rota. A custo da rota de menor custo por unidade de banda reservada entre PN_n e PN_m , denotado por b_{nm} , é dado por $b_{nm} = \sum_{(ij) \in r_{nm}} l_{ij}$.

No passo seguinte, determinamos como atribuir os terminais aos nós provedores. Para exprimir a atribuição entre os terminais e nós provedores como um problema de otimização, introduzimos as variáveis de decisão binárias y_{ij} . Consideramos $y_{ij} = 1$ se ES_i for atribuído para o provedor PN_j , e $y_{ij} = 0$ em caso contrário. A formulação do problema é dada a seguir:

$$\begin{aligned} \text{Minimize} \quad & \sum_{i=1}^M \sum_{k=1}^N \Omega_i \alpha_{ik} y_{ik} + \sum_{i=1}^M \sum_{j=1}^N \sum_{k=1}^M \sum_{l=1}^N y_{ij} y_{kl} \omega_{ik} b_{jl} \\ & + \sum_{j=1}^M \sum_{l=1}^N \Omega_j \alpha_{jl} y_{jl} \\ \text{sujeito à} \quad & \sum_{j=1}^N y_{ij} = 1 \text{ for } i = 1, \dots, M \end{aligned} \quad (1)$$

O primeiro termo na função objetivo expressa o custo total de acesso para o tráfego entrante na rede. O segundo termo expressa o custo de transporte da rede provedora e o terceiro termos expressa o custo total de acesso de saída. As condições de contorno asseguram que cada terminal é atribuído a um e apenas um nó provedor. A complexidade deste procedimento é dominada pela atribuição dos terminais aos nós provedores.

O problema de otimização da Eq. (1) é uma variante do problema de atribuição quadrático (QAP) no qual atribui-se um item a um recurso de tal forma que este item é atribuído a apenas um recurso, e cada recurso tem um item atribuído a si [15]. Por fim, no terceiro passo, baseado no resultado da otimização, eliminamos todos os nós provedores que não tem terminais atribuídos a si e que não façam parte de uma rota de menor custo entre outros dois nós provedores.

III. FORMULAÇÃO MATRICIAL PARA O PROBLEMA DA ATRIBUIÇÃO

Vamos expressar o problema de otimização como um problema na forma matricial. Esta representação permite expressar a estrutura combinatorial do problema de forma mais adequada

do que a dada pela Eq. (1). Observando os custo de transporte e acesso na forma matricial, podemos identificar mais rapidamente algumas condições para as quais a topologia da rede provedora pode ser obtida de forma eficiente.

Sejam as matrizes $\Omega = \{\omega_{ij}\}$, $B = \{b_{ij}\}$ e $\alpha = \{\alpha_{ij}\}$ representando, respectivamente, os requisitos de banda, de custo de transporte e de acesso. Seja u um mapeamento de $i \in \{1, \dots, M\}$ tal que $u(i) = j$, onde $j \in \{1, \dots, N\}$. Nos termos da Eq. (1), temos que $u(i) = l$ se $y_{il} = 1$. Note que o vetor $\underline{u} = (u(1), u(2), \dots, u(M))$, com $u(i) \leq N$ para $i = 1, \dots, M$ fornece uma atribuição factível de terminais aos provedores. Se para algum T_i tivermos $\alpha_{ik} = \infty$, então $u(i) = k$ não é factível.

A função objetivo da Eq. (1) pode ser reescrita como

$$Z(\underline{u}) = \sum_{i=1}^M \sum_{j=1}^M \omega_{ij} (\alpha_{iu(i)} + b_{u(i)u(j)} + \alpha_{ju(j)}) \quad (2)$$

Pode-se verificar que a função $Z(\underline{u})$ é a função objetivo do problema original. Uma minimização sobre todos os vetores \underline{u} produz uma solução para o problema de desenho da topologia. Note que as condições de restrição estão implicitamente embutidas na definição dos $u(i)$'s.

Em geral, a otimização reformulada na Eq. (2) não é mais simples do que o problema original. Contudo, existem casos especiais onde a relação entre as matrizes B e α levam a uma solução com complexidade linear. Tomemos v_i tal que α_{iv_i} é o menor valor entre os α_{ij} , ou seja, $\alpha_{iv_i} = \min_j \{\alpha_{ij}\}$. Portanto, a complexidade de resolver o problema de otimização pode ser reduzida se as condições seguintes ocorrerem:

$$(C1) \quad b_{ij} \leq b_{ik} + b_{kj} \text{ for all } i, j, k \leq N.$$

$$(C2) \quad \alpha_{ij} \geq \alpha_{iv_i} + b_{v_i j} \text{ for all } i \leq M \text{ and } j, v_i \leq N.$$

A condição (C1) sempre acontece, pois os elementos da matriz B são baseados nas rotas de menor custo. Portanto, a desigualdade triangular dada é obtida por construção da matriz. A condição (C2) é satisfeita se a estrutura dos custos for tal que o custo de acesso é bem maior do que o custo de transporte, por unidade de reserva. Em tal cenário, o custo total é minimizado quando se atribui o terminal T_i ao provedor que oferece o menor custo de acesso para este, PN_{v_i} .

Portanto, podemos dizer que a função objetivo $Z(\underline{u})$ é minimizada para o mapeamento $u(i) = v_i$ se as matrizes α and B são tais que as condições (C1) e (C2) forem satisfeitas, onde $\alpha_{iv_i} = \min_j \{\alpha_{ij}\}$. Para simplificar a notação, usamos $u(i)$ para nos referirmos à v_i .

IV. ABORDAGENS METAHEURÍSTICAS

Se a rede não satisfaz as condições (C1) e (C2) dadas na seção precedente, o esforço computacional para resolver o problema de otimização impede a obtenção de soluções exatas para redes grandes. Espera-se que topologias realistas devem ser encontradas para redes com milhares de terminais e nós provedores. Contudo, soluções exatas para o problema de atribuição quadrático podem ser obtidas para problemas com até cerca de 30 terminais e nós [5]. Portanto, para resolver o problema do desenho da rede provedora, precisamos nos voltar para métodos metaheurísticos. Existe um grande número de metaheurísticas que auxiliam na busca de soluções para os

problemas combinatoriais, tais como o problema de atribuição quadrático. Entre as metaheurísticas mais utilizadas, temos a busca tabu, o *simulated annealing* e a computação evolutiva [5]. O *simulated annealing* foi selecionado neste trabalho por apresentar um bom desempenho na solução de problemas de natureza combinatorial, tanto linear quanto quadrática [4], [17]. Discutimos também algumas heurísticas simples que estabelecem a atribuição baseando-se apenas nos custos de acesso, e referimo-nos a tais heurísticas como ambiciosas.

A. Simulated Annealing

A metaheurística de *simulated annealing* emprega uma analogia entre os problemas de física estatística e problemas combinatoriais [13]. O procedimento considera um sistema em equilíbrio térmico a uma temperatura t com um nível de Energia E_k . A partir daí, uma perturbação aleatória é aplicada no sistema, correspondendo a uma mudança no nível de energia. Se o novo nível de energia E_j é menor do que E_k , a perturbação é aceita e o sistema evolui para um novo estado. Se o nível de energia aumenta, o sistema pode evoluir para um novo estado com uma probabilidade que é proporcional à $e^{\frac{E_i - E_k}{t}}$. Depois de um número razoavelmente grande de estados terem sido gerados e avaliados, a temperatura é decrementada e o processo é repetido. A medida que a temperatura diminui, a probabilidade de aceitar perturbações que aumentem o nível de energia do estado corrente também diminui. Isto implica em aceitar, depois de um longo tempo, apenas as perturbações que melhorem a solução. O algoritmo termina com algum critério de parada determinado, tal como o número de estados avaliados e/ou uma temperatura relativamente baixa.

A solução inicial é denotada por S_0 e nos referimos à S_{melhor} , S_{atual} e S_{novo} para representar a melhor solução, a corrente e a nova solução obtida a partir da solução corrente. Os valores das funções objetivo para as soluções inicial, nova, melhor e corrente são denotadas por Z_0 , Z_{novo} , Z_{melhor} e Z_{atual} , respectivamente. A partir da solução corrente S_{atual} pode-se obter uma nova solução S_{novo} através de uma busca aleatória na vizinhança da solução corrente, denominada $BUSCA(S_{atual})$. A nova solução pode ser obtida a partir da solução corrente através da troca de posição de dois bits. Supondo que a mudança é tal que o terminal T_k é escolhido e atribuído ao provedor PN_l , a função objetivo é alterada através da expressão $\Delta(Z_{novo}, Z_{atual}) = \sum_{j=1, j \neq k}^M \omega_{kj} (\alpha_{kl} + b_{lu_j} - \alpha_{ku_k} - b_{u_k u_j}) + \sum_{i=1, i \neq k}^M \omega_{ik} (\alpha_{kl} + b_{u_i l} - \alpha_{ku_k} - b_{u_i u_k})$. A nova solução será aceita se $\Delta(Z_{novo}, Z_{atual})$ for negativo. Em tal caso, é feita uma verificação para identificar se a nova solução é a melhor encontrada até o momento. Por outro lado, se $\Delta(Z_{novo}, Z_{atual})$ for não-negativo, a nova solução será aceita com uma probabilidade que diminui com a temperatura corrente t . Um número uniformemente distribuído no intervalo $[0,1)$ é gerado para decidir se a nova solução será aceita. Na Figura 3, ilustramos algoritmicamente o método.

A temperatura t , com valor inicial t_0 , é o parâmetro que controla a evolução do algoritmo. O mecanismo de decréscimo de temperatura utiliza geralmente uma escala geométrica, na qual a temperatura decresce em progressão geométrica $t_{k+1} =$

Algoritmo Simulated Annealing (S_0, Z_0, t_0) ;

```

begin
     $S_{melhor} \leftarrow S_{atual} \leftarrow S_0$ 
     $Z_{melhor} \leftarrow Z_{atual} \leftarrow Z_0$ 
     $t = t_0$ 
    repeat
        for  $i = 1$  to  $Rep_{max}$  do
             $S_{novo} \leftarrow BUSCA(S_{atual})$ 
            if  $\Delta(Z_{atual}, Z_{novo}) < 0$ 
                 $S_{novo} \leftarrow S_{atual}$ 
                if  $Z_{novo} < Z_{melhor}$ 
                     $S_{melhor} \leftarrow S_{novo}$ 
                     $Z_{melhor} \leftarrow Z_{novo}$ 
                endif
            else
                if  $e^{-\Delta(Z_{novo}, Z_{atual})/t} > Rand(0, 1)$ 
                     $S_{novo} \leftarrow S_{atual}$ 
                endif
            endif
        endfor
         $t \leftarrow r_c \cdot t$ 
    until temperatura muito baixa
    return  $S_{melhor}, Z_{melhor}$ 
end
    
```

Fig. 3. Simulated Annealing.

$r_c \cdot t_k$ com $0 < r_c < 1$). A cada nível de temperatura, um número normalmente fixo de soluções são geradas e avaliadas. Este número é denominado fator de repetição e denotado como Rep_{max} . O valor desta repetição deve ser suficientemente alto, para permitir que boas soluções sejam encontradas em cada nível de temperatura. O processo continua até que a temperatura atinja um valor tão baixo que efetivamente impeça que a solução vai mudar. Neste ponto o algoritmo retorna com a melhor solução encontrada e o valor da função objetiva para esta solução.

B. Algoritmo Ambicioso

Motivados pelo caso especial apresentado na Seção III, apresentamos agora um algoritmo simples para a atribuição de terminais à provedores denominado algoritmo ambicioso. Para este algoritmo, atribuímos cada terminal ao nó provedor que apresenta o menor custo de acesso, ignorando portanto os custos de transporte nesta atribuição. Usando a notação previamente apresentada, se escolhermos v_i tal que o custo de acesso α_{iv_i} seja o menor, ou de outra forma, $\alpha_{iv_i} = \min_j \{\alpha_{ij}\}$, então o algoritmo ambicioso atribui ao terminal i o nó provedor v_i .

Esta estratégia ambiciosa pode ser computada rapidamente, pois tem complexidade linear. O algoritmo certamente apresenta um bom desempenho quando o custo de acesso é menor do que o custo de transporte. Ainda considerando o caso especial apresentado na Seção III, quando as condições (C1) e (C2) ocorrem, esta estratégia permite encontrar a solução exata. Esta exatidão segue do fato de que se T_i and T_k forem atribuídos à PN_{u_i} e PN_{u_k} respectivamente, o custo do caminho de T_i to T_k é dado por $\alpha_{iu_i} + b_{u_i u_k} + \alpha_{ku_k}$. Pela

TABELA I
 AVALIAÇÃO DO FATOR DE REPETIÇÃO PARA REDES PEQUENAS
 ($M = N = 9$).

Fator de repetição Rep_{max}	Desvio máximo do ótimo	Número de soluções ótimas encontradas
10	6.59%	1
20	4.44%	3
30	1.41%	4
40	0.02%	7
50	0.02%	9

condição (C2), $\alpha_{iu_i} \geq \alpha_{iv_i} + b_{v_iu_i}$ e $\alpha_{ku_k} \geq \alpha_{kv_k} + b_{v_ku_k}$ e pela condição (C1), $b_{v_iu_i} + b_{u_iu_k} + b_{v_ku_k} \geq b_{v_iv_k}$. Portanto, o caminho de T_i para T_k tem custo mínimo se T_i e T_k forem atribuídos aos provedores usando este algoritmo ambicioso. Mostraremos posteriormente que, mesmo se a condição (C2) não for satisfeita para todos os nós provedores, ou seja $\alpha_{ik} \leq \alpha_{iv_i} + b_{v_ik}$ para algum k , a solução fornecida pelo algoritmo ambicioso pode fornecer bons resultados.

V. AVALIAÇÃO NUMÉRICA

Nesta seção, avaliamos as abordagens para criar as topologias descritas anteriormente. Neste sentido, tentaremos avaliar o quão bem as heurísticas apresentadas, a do *simulated annealing* e a da abordagem ambiciosa, podem se aproximar da solução exata. Para esta avaliação, geramos uma rede de nós provedores usando o Modelo de Topologia de Rede da Georgia Tech (GT-ITM) [19]. Produzimos um grafo aleatório através do modelo ‘Pure Random’ representando a conectividade entre os provedores (não usamos o GT-ITM para simular a topologia da Internet subjacente). Um arco no grafo indica que dois provedores compartilham um ISP. O modelo Pure Random insere um arco com probabilidade P , onde P é um parâmetro de entrada denominado *probabilidade do arco*. O custo do transporte entre dois provedores é obtido de uma distribuição uniforme na faixa [5, 50] (para alguma métrica arbitrária), e $l_{ij} = \infty$ se GT-ITM não inserir um arco entre os provedores PN_i e PN_j . O custo de acesso α_{ij} também é obtido de uma distribuição uniforme na faixa [5, 50]. Supomos que cada terminal pode ser conectado a um ou mais nós provedores. Cada um deles pode se conectar aleatoriamente a um conjunto selecionado de $p_\alpha \cdot 100\%$ nós provedores, onde $0 \leq p_\alpha \leq 1$ é um parâmetro. A matriz de reserva tem os coeficientes ω_{ij} distribuídos uniformemente na faixa de [10, 20] Mbps. Quando mostramos o custo total de uma rede provedora, este custo é apresentado como uma média de 100 cálculos, e para cada cálculo empregamos uma rodada de execução.

A. Ajuste do Simulated Annealing

Primeiramente, avaliamos o desempenho da heurística *simulated annealing*, comparando os resultados dados por este com a solução exata da Eq. (1). Para redes pequenas, podemos resolver a Eq. (1) usando métodos enumerativos. Se a rede é grande, podemos determinar a solução exata para os casos especiais descritos na Seção III.

Comparando o custo mínimo da Eq. (1) com os resultados obtidos pelo *simulated annealing* com $M = 9$ e $N = 9$, $p_\alpha = 1$ e $P = 0.5$, na Tabela I, os resultados do *simulated annealing*

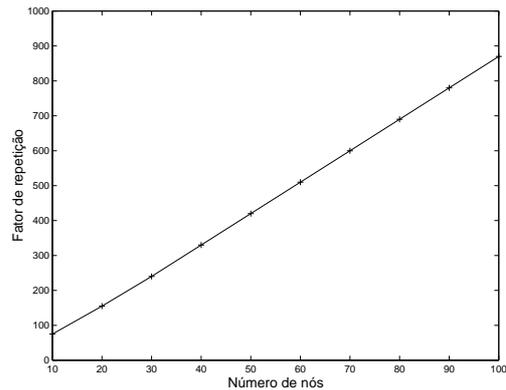


Fig. 4. Ajuste do fator de repetição para casos especiais.

são mostrados para valores diferentes de Rep_{max} . A primeira coluna apresenta o valor do fator de repetição e a segunda o desvio médio da solução ótima. A terceira coluna indica o quão frequentemente o algoritmo encontra a solução ótima. Os resultados indicam que para um fator de repetição maior do que 40, o *simulated annealing* encontra a solução ótima em várias rodadas, o que permite extrair este valor executando o procedimento algumas vezes.

Para redes maiores, a comparação do *simulated annealing* com a solução ótima só é possível com a garantia da validade das condições (C1) e (C2) da Seção III. Para este experimento, geramos entre 10 e 100 provedores e terminais (com $M = N$). Utilizamos os parâmetros $p_\alpha = 1$ e $P = 0.5$ e para atender a condição (C2), selecionamos o custo de acesso de tal forma que para cada terminal T_i , escolhemos aleatoriamente um provedor PN_{v_i} e selecionamos α_{iv_i} também aleatoriamente no intervalo [5, 50]. Portanto, para todos os nós provedores, temos que $\alpha_{ij} = \alpha_{iv_i} + b_{v_ij}$.

Apresentamos a comparação em termos do fator de repetição Rep_{max} necessário para obter soluções ótimas 99% das rodadas. Os resultados são ilustrados na Figura 4. Esta figura mostra que o algoritmo *simulated annealing* conseguiu determinar a solução ótima grande parte das vezes para o fator de repetição ilustrado. Este valor em geral aumenta linearmente com o tamanho do problema, e esta característica já foi observada em experimentos com o *simulated annealing* para problemas de atribuição quadrática [4].

B. Comparação de Desempenho das Heurísticas

Consideramos agora as redes para as quais a condição (C2) não vale e vamos comparar o custo das topologias geradas tanto pelo *Simulated Annealing* quanto pelo algoritmo ambicioso. Como um *benchmark*, incluímos também o resultado da atribuição aleatória de terminais. A avaliação é feita para terminais e nós provedores em número variando de 10 à 100, com $M = N$. Empregamos probabilidade de ocorrência do arco com valores $P = 0.1, 0.5$ e 0.9 e selecionamos $p_\alpha = 0.9$. Em relação ao custo de acesso, para cada terminal T_i , selecionamos aleatoriamente um provedor PN_{v_i} e o valor de α_{iv_i} , este último no intervalo [5, 50]. A partir daí, para todos os outros provedores $j \neq v_i$ selecionamos aleatoriamente os valores de α_{ij} (para $i = 1, 2, \dots, M$),

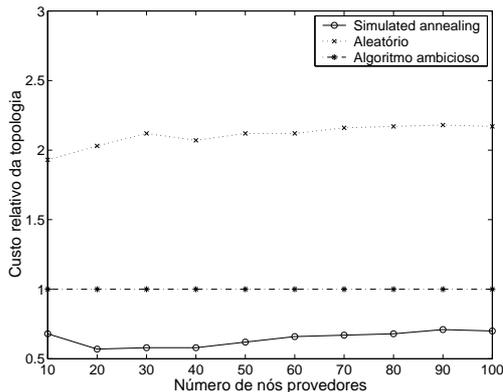


Fig. 5. Custo da topologia para as heurísticas com $P = 0.1$.

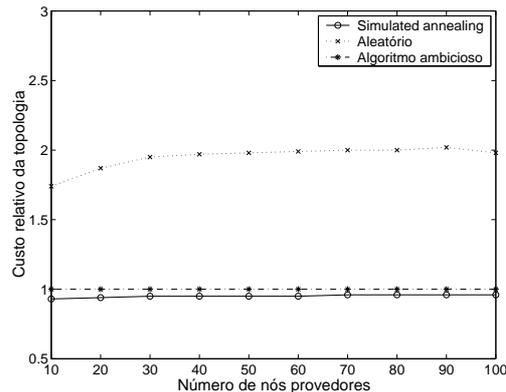


Fig. 7. Custo da topologia para as heurísticas com $P = 0.9$.

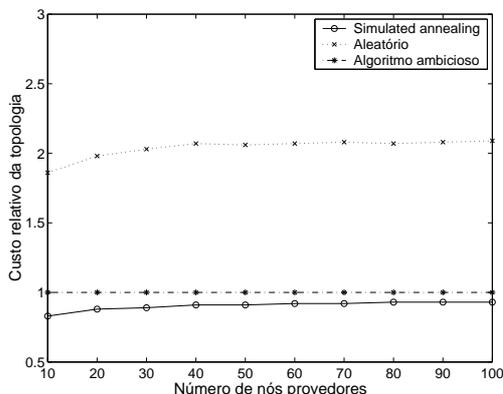


Fig. 6. Custo da topologia para as heurísticas com $P = 0.5$.

também no intervalo $[5, 50]$, onde forçamos a ocorrência de $\alpha_{iv_i} \leq \alpha_{ij} \leq \alpha_{iv_i} + b_{v_i,j}$. Estas restrições asseguram que a condição (C2) não ocorra.

Os resultados das Figuras 5–7 mostram os valores do custo normalizados pelos valores obtidos com o algoritmo ambicioso. Vemos que o *simulated annealing* e o algoritmo ambicioso fornece resultados similares para $P = 0.5$ e $P = 0.9$, e o *simulated annealing* tem melhor desempenho para $P = 0.1$. O ganho fornecido pelas heurísticas, de forma geral, podem melhorar até duas vezes um valor aleatório. Os resultados não dependem do tamanho da rede, no intervalo medido.

VI. CONCLUSÕES

Neste artigo abordamos o problema de desenho topológico de uma rede de serviços *overlay*, que oferece um valor agregado aos clientes, através de compra de enlaces com garantias de banda de um determinado número de ISPs. Dentro das suposições feitas no trabalho, formulamos o problema geral de desenho desta topologia como um problema de otimização NP-hard, mas que em alguns casos onde a estrutura de custos permite, este problema pode ser resolvido rapidamente. Apresentamos algumas heurísticas que permitem a construção da topologia mesmo quando a solução exata não pode ser obtida de forma prática. Os resultados numéricos apresentados demonstram que nas situações onde é possível se comparar as soluções obtidas com a heurística *simulated annealing* com

a solução exata, a heurística apresenta um bom desempenho. Os dados também mostram que heurísticas ambiciosas simples podem fornecer um bom resultado bom. Os resultados apresentados dependem particularmente da estrutura linear de custos dos enlaces. Uma estrutura de custo diferente, que dependa da quantidade de banda reservada poderia fornecer resultados diferentes e uma abordagem nova para o problema.

REFERÊNCIAS

- [1] Akamai, Inc. <http://www.akamai.com>.
- [2] D.G. Andersen, H. Balakrishnan, M.F. Kaashoek and R. Morris. Resilient Overlay Network. In *Proc. 18th ACM SOSP 2001*, Banff, October 2001.
- [3] R. Braynard, D. Kostic, A. Rodriguez, J. Chase and A. Vahdat. Opus: An Overlay Peer Utility Service. *Proc. IEEE OpenArch'02*, June 2002.
- [4] R.E. Burkard and F. Rendl. A Thermodynamically Motivated Simulation Procedure for Combinatorial Optimization Problems. *European Journal of Operational Research*, 17:169-14, 1983.
- [5] E. Cela. *The Quadratic Assignment Problem*. Kluwer Academic Publishers, 1998.
- [6] Y. Chu, S.G. Rao and H. Zhang. A Case for End System Multicast. In *Proc. ACM Sigmetrics*, June, 2000.
- [7] Z. Duan, Z.L. Zhang and Y.T. Hou. Service Overlay Networks: SLAs, QoS and Bandwidth Provisioning. In *Proc. 10th IEEE International Conference on Network Protocols*, Paris, France, November 2002.
- [8] Gnutella. <http://gnutella.wego.com>.
- [9] X. Gu, K. Nahrstedt, R.H. Chang and C. Ward. QoS-Assured Service Composition in Managed Service Overlay Networks. In *Proc. IEEE 23rd International Conference on Distributed Computing Systems*, Providence, May 2003.
- [10] Internap Network Services Corporation. <http://www.internap.com>.
- [11] International Telecommunication Union. IP-based networks: Pricing of telecommunication services. Final report, January 2003.
- [12] A. Keromytis, V. Misra and D. Rubenstein. Secure Overlay Networks. *Proc. SIGCOMM'02*, pp.61-72, 2002.
- [13] S. Kirkpatrick, C. Gelatt and M. Vecchi. Optimization by Simulated Annealing. *Science*, 220:291-307, 1983.
- [14] Z. Li and P. Mohapatra. QRON: QoS-aware Routing in Overlay Networks. *IEEE Journal of Selected Areas of Communications*, 22(1):29-40, January, 2004.
- [15] M. Padberg and M.P. Rijal. *Location, Scheduling, Design and Integer Programming*. Kluwer Academic Publishers, 1996.
- [16] T. Robertazzi. *Planning Telecommunication Networks*. John Wiley & Sons, 1999.
- [17] S. Sofianopoulou. Simulated annealing applied to the process allocation problem. *European Journal of Operational Research*, 60:327-334, 1992.
- [18] L. Subramanian, I. Stoica, H. Balakrishnan and R.H. Katz. OverQoS: Offering Internet QoS using Overlays. In *Proc. HotNet-1 Workshop*, October 2002.
- [19] E. W. Zegura. GT-ITM: Georgia Tech Internetwork topology models (software). <http://www.cc.gatech.edu/project>, 1996.