

PROPOSTA DE UM ALGORITMO BRANCH AND BOUND PARA PROJETO DE REDES DE CUSTO MÍNIMO SUJEITA A RESTRIÇÃO DE CONECTIVIDADE

ROCHA, Marcelo Lisboa¹

ASSIS, João Gilberto Oliveira²

¹Doutor em Engenharia Elétrica pela UFRJ; Mestre em Ciência da Computação pela UFF; Mestre em Engenharia Elétrica pela UFRJ.

²Bacharel em Ciência da Computação pelo Centro Universitário UnirG.

RESUMO

Este artigo visa o projeto de redes de custo mínimo com capacidade de sobrevivência, onde esta sobrevivência é expressa em termos dos requisitos de conectividade de cada nó da rede. Aqui é proposta uma abordagem baseada no algoritmo *Branch and Bound* que obtém soluções ótimas com baixo tempo computacional. Resultados computacionais e comparações com outros métodos da literatura são apresentados.

Palavras-Chave: Otimização Combinatória. Branch and Bound/Heurísticas. Sobrevivência de rede.

ABSTRACT

This article aims the network projects of minimum costs with capacity of survival, where this survival is expressed in terms of the requisites of connectivity of each knot of the network. Here is proposed a boarding based in Branch and Bound algorithm that achieves optimal solutions in low computational time. Computational results and comparisons with other methods of literature are presented.

Keywords: Combinatorial Optimization. Branch and Bound/Heuristics. Network Survivability.

1 INTRODUÇÃO

A partir dos anos 70, observou-se o surgimento de novas tecnologias para redes de dados digitais, objetivando responder às necessidades de empresas e instituições com tráfego de dados cada vez maiores, que desenvolvem aplicações cada vez mais complexas e com requisitos cada vez mais rigorosos sobre a rede (ROBLEDO, 2005).

Aplicações tais como: jogos interativos, programas de compartilhamento de arquivos, conferências de áudio e vídeo, entre outras, estão presentes no cotidiano de praticamente todos os usuários de computadores que utilizam a internet (FERREIRA, 2000). Além da necessidade de grande largura de banda passante e de baixo atraso, existe também uma tendência para que as aplicações apresentem um comportamento cada vez mais dinâmico, modificando em pouco tempo o conjunto de origens e destinos de tráfegos na rede. Outra questão é relativa ao aumento do período de duração de uma aplicação, que pode variar de algumas horas a até 24 horas por dia. Assim, se torna evidente que as redes de telecomunicações possuam alta disponibilidade e confiabilidade (NICOLLETTI, 2000).

Para o cenário mundial e brasileiro, as empresas têm exigido maneiras tanto seguras quanto econômicas para o fornecimento dos serviços de rede, e conseqüentemente, aplicações adequadas de ferramentas computacionais e técnicas de otimização no planejamento destas redes (LACHTERMACHER, 2002).

Segundo (BICUDO, 2005), com o surgimento das fibras óticas nos meios de transmissões, a capacidade de transporte aumentou substancialmente, permitindo transportar grandes massas de dados sobre uma única estrutura de rede, aumentando, assim, a complexidade da gerência de todos os seus elementos e proporcionalmente sua vulnerabilidade. Contudo, os custos de se manter uma rede de comunicação ativa variam conforme a complexidade com que esta esteja implantada. Logo, a necessidade de manter estas conexões “vivas”, torna-se necessária. Sobrevivência é a capacidade de uma rede continuar funcionando na ocorrência de falhas eventuais (BARBOSA; ROCHA, 2006).

Para garantir a sobrevivência da rede em certos níveis, a análise de sua topologia é de fundamental importância. Buscar projetar a topologia de uma rede que satisfaça as necessidades de sobrevivência ao menor custo possível é um problema do tipo NP-Completo (CANCELA; ROBLEDO; RUBINO, 2003). Assim sendo, a utilização de técnicas eficientes para o problema em questão é de grande valia, o que vai de encontro à proposta deste trabalho.

Neste trabalho, a característica de sobrevivência considerada é o número mínimo de conexões a que cada nó (elemento) deve estar conectado. Uma abordagem frequentemente empregada na formulação e resolução deste problema consiste no emprego de Modelos de Otimização, caracterizando-o como um Problema de Otimização Combinatória.

Segundo (GOLDBARG; LUNA, 2000), Otimização Combinatória (OC) trouxe consigo vários benefícios à ciência de modo geral, resolvendo vários problemas práticos tais como: Projetos de Sistemas de Distribuição de Energia Elétrica, Posicionamento de Satélites, Projetos de Computadores e de Chips VLSI, Roteamento ou Escalonamento de Veículos, Alocação de Trabalhadores ou Máquinas a Tarefas, Empacotamento de Caixas em *Containers*, Corte de Barras e Placas, Sequenciamento de Genes e DNA, entre outras. Problemas de Otimização têm como objetivo encontrar um ponto ótimo (mínimo ou máximo) de uma função definida sobre certo domínio, função esta denominada “Função Objetivo” que é uma função das variáveis de decisão a ser minimizada ou maximizada.

O presente trabalho tem como objetivo analisar os resultados obtidos com a implementação e testes de uma técnica que utiliza um método exato denominado *Branch and Bound* (Método de Partição e Avaliação Sucessivas) que será aplicado ao problema de sobrevivência em redes de telecomunicações e compará-los com resultados apresentado no trabalho de Barbosa; Rocha (2006). Desta forma, uma das principais questões a respeito do projeto de redes de telecomunicações é computar topologias de rede que forneçam proteção contra falhas de equipamentos e links da rede e, conseqüentemente, aumente sua sobrevivência. Assim, o problema de computação de topologia que será focado neste trabalho, consiste em selecionar links de modo que a soma de seus custos seja minimizada e a exigência do número mínimo de conexões de cada nó seja satisfeita.

Este trabalho está dividido como se segue. Na Seção 2, é apresentada uma análise do problema de sobrevivência de redes. Na Seção 3, é apresentada a modelagem matemática do problema proposto. A Seção 4 discorre a respeito do método *Branch and Bound* adotado para resolver o problema em questão. Já na Seção 5, são mostrados os testes computacionais, resultados e tempo de execução do método utilizado, como também uma comparação com outros trabalhos da literatura. O trabalho é então concluído na Seção 6.

2 ANÁLISE DO PROBLEMA DE SOBREVIVÊNCIA EM REDES

Manter certos serviços ativos em uma estrutura de redes é um problema complicado, pois confiabilidade e integridade dependem da robustez da infra-estrutura de rede, ou seja, da confiabilidade dos equipamentos (links ou nós) como também de como os nós estão interligados (topologia da rede). Por isso, a confiabilidade da rede pode ser caracterizada por diversos parâmetros distintos. No presente trabalho, a confiabilidade da rede foi baseada na satisfabilidade da exigência de conectividade de cada nó da rede, permitindo, assim, que a rede possa resistir a falhas tanto nos *links* de comunicação quanto nos nós de serviço.

A sobrevivência pode ser conceituada como a capacidade de sobrevivência da rede na ocorrência de eventuais falhas, onde esta capacidade pode ser total ou parcial. Sobreviver significa manter seus serviços ativos em certos níveis (MELLO; TAVARES, 1996). Segundo Robledo (2005), as falhas geralmente são causadas por episódios não desejados, como por exemplo: Erros não detectados no software e/ou hardware, interrupções nos cabos de transmissão originados por eventos como raios, furacões, terremotos, tornados, incêndios, entre outras. As falhas podem ser classificadas em:

- Falha de nó: Quando ocorre num centro de fios;
- Falha de arcos: Quando ocorre num link de comunicação.

No entanto, há várias maneiras diferentes de avaliar a integridade da estrutura de rede após uma catástrofe. Por exemplo, pode-se calcular o número de assinantes conectados, a receita preservada ou calcular o volume de tráfego sobrevivente, analisar os níveis de serviços ativos sobreviventes (BARBOSA; ROCHA, 2006).

A confiabilidade das redes de telecomunicações tem despertado interesse maior devido à importância crescente dos serviços prestados, ao nível de exigência da qualidade dos serviços prestados aos usuários e à severidade das falhas em sistemas de alta hierarquia (BOVO, 2004).

Como consequência, as redes tornaram-se muito vulneráveis e a gerência de todos os seus elementos constituintes tornou-se bastante complexa (DETONI, 2001). Por isso, é importante proceder uma pré-classificação seja dos centros de fios seja de serviços que circulam na rede, rotulando-os como especiais ou comuns. Um serviço será classificado como especial se possui importância destacada; por exemplo, o destaque pode ser conferido pela receita gerada (DETONI, 2001).

O projeto de redes tolerante a falhas é extremamente complexo, pois, de acordo com Stoer (1996), computar a melhor solução para este problema o caracteriza como NP-Difícil. Desta forma, não pode ser considerado um problema de otimização simples.

Neste trabalho, a característica de sobrevivência considerada é o número mínimo de conexões a que cada nó (elemento) deve estar conectado. A cada nó da rede é associado um valor que indica o número mínimo de conexões que o mesmo deve ter (quanto maior este número, mais importante é este nó) (BARBOSA; ROCHA, 2006). Isto se relaciona à sobrevivência da rede, pois, quanto maior for o número mínimo de conexões de um determinado nó, maior é o número de caminhos (conexões) disponíveis em caso de falha. Assim, o problema de computação de topologia, que será focado, consiste em selecionar links, de modo que a soma de seus custos seja minimizada e os requisitos de conectividade dos nós sejam satisfeitos. Aqui, uma rede é representada por uma coleção de nós (elementos) (hubs, switches, roteadores, satélites, rádio-bases, etc.) e a conexão entre eles se dá por arestas (links) (fibra ótica, fios elétricos, etc.), sendo que a robustez da topologia da rede está de acordo com sua confiabilidade (BOVO, 2004).

3 MODELAGEM DO PROBLEMA PROPOSTO

Dado um grafo conexo $G=(V, E)$ não direcionado, onde 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 ainda que, cada vértice $v_i \in V$ possui um número mínimo de ligações r_i e que cada aresta interligando os vértices i e j possui um custo associado c_{ij} (valor ou distância). Este problema tem, como objetivo, buscar o subgrafo (topologia de rede) $G'=(V', E')$, onde $V'=V$ e $E' \subseteq E$ de custo mínimo e que garanta sua sobrevivência, obedecendo aos requisitos de conectividade de cada nó (r_i) que é obtido de acordo com os níveis de importância dos serviços que esse possui. Aqui, o custo de G' (topologia) é determinado pela soma dos custos das arestas E' . Um exemplo de grafo G representado uma instância do problema proposto, pode ser visto na Figura 1. A mesma possui 10 nós e 17 arestas, onde: o número interior a cada vértice (círculo) é a sua indicação; o número adjacente a cada vértice é o seu valor de r_i (requisito de conectividade); e o número adjacente a cada aresta é o custo da mesma. Já na Figura 2, é apresentado o subgrafo G' (topologia da rede) de custo mínimo (valor 27) referente a instância da Figura 1.

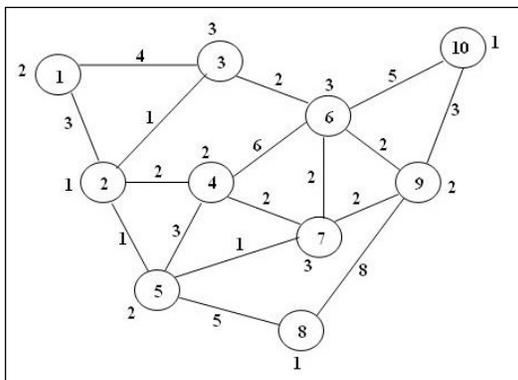


Figura 1. Grafo G de referência.

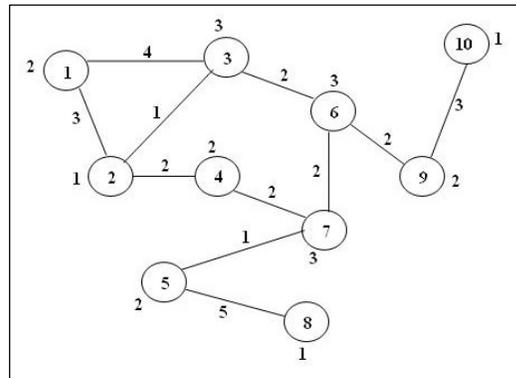


Figura 2. Subgrafo G' (topologia) de G

A seguir, será apresentado o modelo matemático do problema proposto.

3.1 Modelo Matemático

O problema de projeto de redes de custo mínimo sujeita a restrições de conectividade, pode ser formulado como:

$$\text{Minimizar: } \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}, \text{ onde } i \neq j \quad (1)$$

$$\text{Sujeito a: } \sum_{j=1}^n x_{ij} \geq r_i, \forall i = 1, \dots, n \quad (2)$$

$$x_{ij} \in \{0, 1\}, \forall i, j \in E \quad (3)$$

onde (1) é a função objetivo que representa o custo total da topologia, que é a soma das arestas que estão sendo consideradas. As arestas que estão ($x_{ij} = 1$) ou não ($x_{ij} = 0$) sendo consideradas estão representadas em (3). Já o conjunto de restrições (2) garante que os requisitos mínimos de ligações de cada vértice $v_i \in V$ sejam respeitados.

4 MÉTODO EXATO BRANCH AND BOUND

O algoritmo *Branch and Bound* (B&B) é um dos procedimentos mais utilizado na resolução de problemas de PLI (Programação Linear Inteira) ou PLIM (Programação Linear Inteira Mista) (LACHTERMACHER, 2002). Existem diversas variantes deste método para o tratamento de diversos tipos de problemas específicos. A idéia geral é o de dividir o conjunto de soluções viáveis em subconjuntos de interseções entre si, calculando os limites superiores e inferiores para cada subconjunto de acordo com as regras preestabelecidas, ou seja, o método B&B baseia-se na idéia de desenvolver uma enumeração inteligente dos pontos candidatos à solução ótima inteira do problema (LONGO, 2004).

De acordo com Golbbarg e Luna (2000), o termo *branch*, refere-se ao fato de que o método efetua partições no espaço das soluções e o termo *bound* ressalta que a prova de otimalidade da solução utiliza-se de limites calculados ao longo da enumeração. Para utilizar o método, é necessário modelar o problema como um problema de Programação Linear Inteira e em seguida relaxar a formulação para Programação Linear e enumerar o espaço de soluções, através de uma árvore. Aqui, cada nó da árvore representa um programa linear. Cada programa linear (nó) é ramificado (*branch*) enquanto houver valores fracionários em sua solução. Se um nó leva a atribuições inviáveis, ou a solução não promissora, podar (*bound*) o nó.

O algoritmo B&B varre uma árvore na qual cada nó representa um subproblema do problema inicial na intenção de achar o caminho da raiz até uma folha com o menor custo, ou seja, para esse algoritmo, encontrar a solução não é o fim, ele precisa saber qual solução tem o menor esforço e/ou melhor solução. Sabendo isso, ele pode se dar ao direito de dispensar alguns caminhos que o levarão a uma solução, desde que se saiba que essa solução certamente será mais custosa do que uma outra solução já encontrada previamente (LONGO, 2004).

A Figura 3 apresenta os procedimentos básicos do algoritmo B&B. Já a Figura 4 denota um exemplo de modelagem para Programação Linear Inteiro.

```

1  LB ← -∞ % Lower bound
2  x* ← Heurística sobre Po (∅ se não encontrar solução viável)
3  UB ← val(x*) (onde val(∅) = +∞)
4  Ativos ← {Po}
5  enquanto Ativos ≠ ∅ faça
6     escolha P ∈ Ativos
7     Ativos ← Ativos \ {P}
8     se P é viável, % se inviável, ramo é podado
9     x ← solução ótima de P
10    se val(x) > LB então atualiza LB (se necessário)
11    se val(x) < UB % se val(x) ≥ UB, ramo é podado
12    se x é inteiro então
13       UB ← val(x)
14       x* ← x
15    senão
16       crie dois subproblemas P' e P'' a partir de P
17       Ativos ← Ativos ∪ {P', P''}
18  devolva x

```

$$\begin{array}{l}
 \text{minimize} \\
 \text{sujeito a}
 \end{array}
 \left\{ \begin{array}{l}
 3x_1 + 4x_2 \\
 -3x_1 + 2x_2 \leq 2 \\
 x_1 + 3x_2 \leq 11 \\
 x_1 + x_2 \leq 6 \\
 x_1 \geq 0, \quad x_2 \geq 0 \\
 x_i \in \mathbb{Z}^*
 \end{array} \right.$$

Figura 4. Modelagem do Programa Linear Inteiro.

Figura 3. Algoritmo B&B do problema Relaxado (P^0).

A Figura 5 apresenta a árvore de solução gerada a partir do problema da Figura 4.

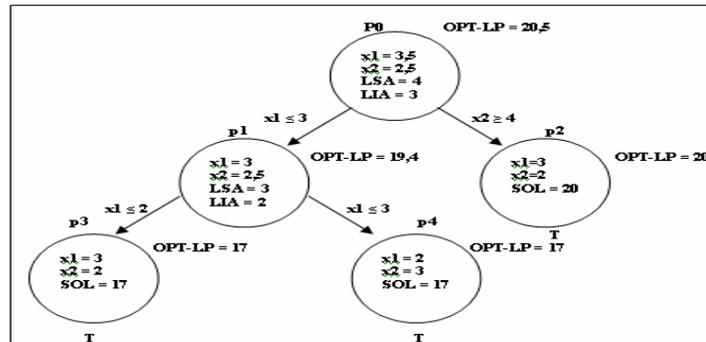


Figura 5. Árvore completa de Branch (sem Bound).

No método B&B considerado neste trabalho para o problema em questão, a árvore de enumeração foi criada de acordo com a ordem crescente de r_i , considerando primeiro os $r_i = 1$, depois os $r_i = 2$ e assim sucessivamente. Isto com o objetivo de alocar primeiramente as arestas (links) relativas aos nós com menor folga. Aqui, também foi utilizada para o *branch* a estratégia de Busca em Profundidade, devido esta estratégia de desdobramento da árvore de busca ter apresentado bons resultados. A técnica de *Bound* da árvore foi desenvolvida de acordo o limite ou corte dos subprogramas, atribuindo em *MS* a melhor solução encontrada e utilizando-a no corte dos nós não promissores.

5 RESULTADOS COMPUTACIONAIS

Nesta Seção, serão apresentados os testes e resultados computacionais realizados sobre metodologia proposta. A seguir, serão apresentadas as instâncias utilizadas para testes e o desempenho do algoritmo utilizado, tanto sobre a qualidade das soluções obtidas quanto sobre o tempo de execução.

5.1 Geração das Instâncias

As instâncias utilizadas neste trabalho consistem de grafos criados com o gerador de instâncias, que foi desenvolvido pelos autores na linguagem C para o problema de sobrevivência de redes. Foram atribuídos valores de r_i para cada vértice i do grafo, sorteado entre $\{1, 4\}$. O valor 4 foi escolhido como limite para r_i por proporcionar uma grande sobrevivência, dado que em Grötschel; Monma; Stoer (1995), foi considerado $r_i = 3$ para uma topologia de rede de fibra ótica conectando pontos críticos de um porta-aviões. Isto foi feito, dado a inexistência de uma biblioteca pública na internet que contenha instâncias para o problema abordado neste trabalho. Vale a pena ressaltar que os autores do trabalho entraram em contato com autores de trabalhos na área, solicitando as instâncias utilizadas em seus respectivos trabalhos e até o momento da conclusão deste trabalho não recebeu retorno dos mesmos.

Para realização dos testes, foram criadas 26 instâncias. Cada uma das 26 instâncias é um grafo conexo, onde o número de nós varia de 10 a 5000; e, por fim, o número de arestas varia de 17 a 8567. Cada aresta é atribuída um peso (custo, valor ou distância). Na Tabela 1 são apresentadas todas as instâncias e seus parâmetros.

5.2 Testes Computacionais

Os testes foram realizados em um computador *Pentium IV*, 3.0 GHz (1MB de Cache L2) com 512 MB de RAM e Sistema Operacional Microsoft Windows XP Service Pack 2, sobre cada uma das 26 instâncias especificadas na Tabela 1.

Os métodos guloso aleatorizado e GRASP foram executados com os seguintes parâmetros:

- Critério de parada (ambos os métodos): 5000 iterações;
- Valor de Alfa (somente GRASP): 0.2.

Para maiores detalhes a respeito do funcionamento e parâmetros dos métodos gulosos, gulosos aleatorizados e GRASP, consultar os textos de Ferreira (2000) e Barbosa; Rocha (2006).

Nas Tabelas 2, 3 e 4 são apresentadas as soluções (custos das topologias) e os tempos de execução (em segundos de CPU) encontradas pelo método B&B, GRASP, guloso aleatorizado e guloso respectivamente sobre as 26 instâncias da Tabela 1.

Tabela 1. Instâncias utilizadas para teste

Instâncias	Nº. de Vértices	Nº. de Arestas
i1	10	17
i2	10	20
i3	11	22
i4	20	31
i5	20	30
i6	21	40
i7	30	51
i8	33	62
i9	39	79
i10	50	83
i11	55	110
i12	54	98
i13	75	130
i14	79	128
i15	94	168
i16	100	166
i17	150	253
i18	180	303
i19	200	342
i20	305	515
i21	444	746
i22	500	889
i23	850	1442
i24	963	1654
i25	1700	2928
i26	5000	8567

Tabela 2. Desempenho do B&B

Resultado BRANCH AND BOUND		
Instância	Valor da Solução	Tempo de CPU
i1	27	0,000
i2	74	0,000
i3	89	0,000
i4	131	0,000
i5	197	0,000
i6	113	0,000
i7	203	0,000
i8	357	0,000
i9	158	0,000
i10	166	0,000
i11	1314	0,000
i12	1149	0,000
i13	1909	0,000
i14	2002	0,000
i15	2379	0,000
i16	4503	0,000
i17	3374	0,000
i18	20855	0,000
i19	8713	0,000
i20	72913	0,000
i21	69274	0,000
i22	13149	0,000
i23	13629	0,100
i24	108656	0,200
i25	494602	0,400
i26	2103163	3,800
		Tempo de CPU em Segundos

Tabela 3. Desempenho do GRASP

Resultado GRASP		
Instância	Valor da Solução	5000 iterações Tempo de CPU
i1	28	0,016
i2	79	0,031
i3	92	0,032
i4	136	0,046
i5	197	0,047
i6	118	0,047
i7	217	0,078
i8	364	0,109
i9	1312	0,157
i10	691	0,188
i11	1433	0,281
i12	1233	0,235
i13	2134	0,375
i14	2229	0,375
i15	2648	0,563
i16	5062	0,562
i17	3756	1,079
i18	22664	1,437
i19	9895	1,735
i20	84228	3,469
i21	79630	6,641
i22	15229	9,078
i23	157516	22,156
i24	124895	28,688
i25	569486	85,625
i26	2450495	760,969
		Tempo de CPU em Segundos

Tabela 4. Desempenho do Guloso Aleatorizado

Resultado GULOSO ALEATORIZADO		
Instância	Valor da Solução	5000 iterações Tempo de CPU
i1	28	0,016
i2	79	0,031
i3	93	0,031
i4	140	0,032
i5	197	0,046
i6	122	0,063
i7	231	0,078
i8	428	0,094
i9	1359	0,109
i10	797	0,125
i11	1698	0,172
i12	1407	0,156
i13	2374	0,218
i14	2371	0,203
i15	3036	0,297
i16	5450	0,281
i17	4352	0,438
i18	26432	0,547
i19	11194	0,625
i20	97503	1,000
i21	91648	1,515
i22	17743	1,844
i23	183210	3,219
i24	148625	3,719
i25	677384	7,188
i26	2895000	26,140
		Tempo de CPU em Segundos

Tabela 5. Desempenho do Guloso

Resultado GULOSO		
Instância	Valor da Solução	Tempo de CPU
i1	32	0,000
i2	87	0,000
i3	100	0,000
i4	161	0,000
i5	215	0,000
i6	219	0,000
i7	216	0,000
i8	390	0,000
i9	1512	0,000
i10	746	0,000
i11	1536	0,000
i12	1348	0,000
i13	2270	0,000
i14	2433	0,000
i15	2710	0,000
i16	5494	0,000
i17	3959	0,000
i18	23774	0,000
i19	10198	0,000
i20	87334	0,000
i21	81721	0,000
i22	15449	0,000
i23	159698	0,000
i24	125779	0,000
i25	572188	0,000
i26	2454511	0,000
		Tempo de CPU em Segundos

5.3 Análises e Comparações dos Resultados

Nesta seção, serão feitas análises e comparações dos resultados obtidos nos testes computacionais obtidos neste trabalho com os do trabalho de Barbosa; Rocha (2006) da literatura. De modo a ser possível realizar comparações e análises diretamente entre as técnicas, foram utilizados os programas referentes ao Guloso, Guloso Aleatorizado e GRASP, como também as instâncias de Barbosa; Rocha (2006).

5.3.1 Análises

Nesta Seção, serão analisados os resultados obtidos para os três métodos heurísticos do trabalho de Barbosa; Rocha (2006) e o método Branch-and-Bound aqui proposto, para uma das 26 instâncias utilizadas para testes (i3). Na Figura 6 é apresentado o grafo da instância i3 analisada.

A Figura 7 apresenta a melhor topologia de rede obtida pelo método guloso, com valor da solução igual a 100 e possuindo as seguintes arestas: 1-2; 1-4; 1-8; 1-9 1-11; 2-4; 2-7; 2-10; 3-6; 4-5; 4-9; 4-11; 5-6; 5-11; 6-10.

Com a execução da instância de referência i3 com o método guloso aleatorizado, obteve uma topologia ou estrutura de rede com o valor de solução igual a 93 e com as respectivas arestas: 1-2; 1-8; 1-9; 2-4; 2-7; 2-10; 3-6; 4-5; 4-9; 4-11; 5-6; 5-11, conforme representado na Figura 8. Fazendo uma analogia com o Método Guloso, observou-se uma melhora da solução obtida pelo Guloso Aleatorizado, contudo, a um maior custo computacional.

O Algoritmo GRASP apresentou uma topologia com valor de solução 92 para a instância de referência, como pode ser visto na Figura 9, com as respectivas arestas: 1-2; 1-4; 1-11; 2-4; 2-7; 2-10; 3-6; 4-5; 4-9; 4-11; 5-6; 5-11; 8-9. Analisando o grafo da solução do GRASP com o grafo da solução do Método Guloso Aleatorizado, observou-se uma melhora em sua topologia, tanto no valor da solução quanto em tempo computacional.

A Figura 10 apresenta a topologia de rede obtida pelo Método B&B, com o valor da solução 89 e possuindo as arestas: 1-2; 1-4; 1-11; 2-4; 2-7; 2-10; 3-6; 4-5; 4-9; 5-6; 5-11.

A heurística GRASP foi a que obteve resultados mais próximos dos apresentados pelo método exato (B&B), devido à mesma combinar as características de heurísticas de construção e melhoramento (busca local). Para a instância analisada, o GRASP realizou a ligação 4-11, motivo do não alcance do resultado exato (melhor possível).

Tal situação comprova empiricamente, que para as instâncias consideradas, o método B&B utilizado foi superior às heurísticas competidoras.

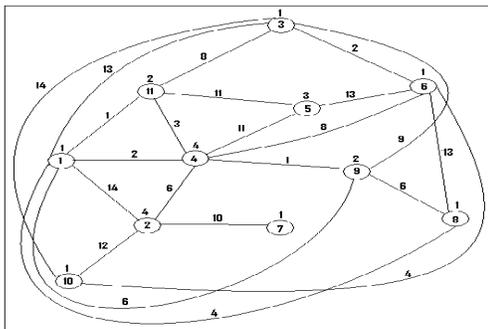


Figura 6. Grafo conexo da Instância i3 analisada.

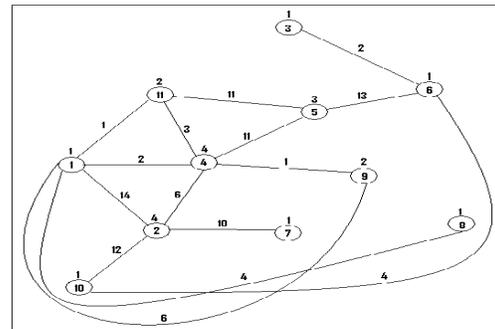


Figura 7. Solução do Método Guloso.

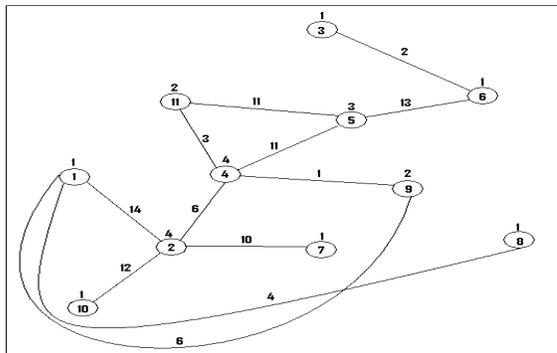


Figura 8. Solução do Método Guloso Aleatorizado.

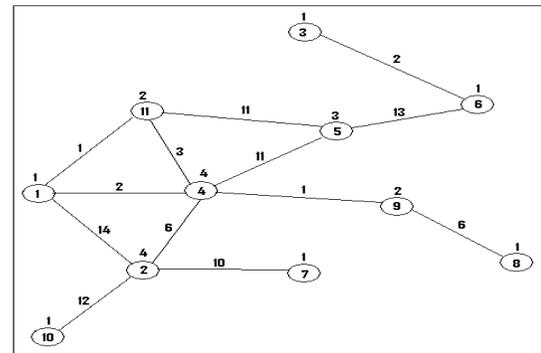


Figura 9. Solução do GRASP.

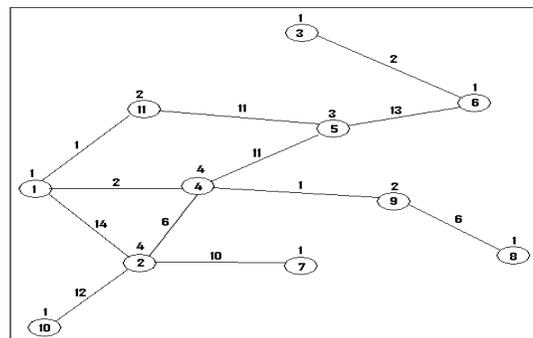


Figura 10. Solução do B&B.

Dentre os quatros métodos analisados, tanto para a instância i3 aqui estudada quanto para as demais 25, a técnica desenvolvida que utilizou o algoritmo *Branch and Bound*, aqui proposto, obteve melhor desempenho tanto nos valores das soluções obtidas quanto nos tempos computacionais. Vale ressaltar que, quanto aos tempos computacionais, o Método Guloso registrou tempo desprezível para a precisão adotada, devido este método construir iterativamente uma única combinação de solução possível, sem testar outras possibilidades, através de um critério de otimização definido por uma função gulosa. Os métodos heurísticos permitem a obtenção de bons resultados, porém a otimalidade não é garantida. Já métodos exatos, tais como o B&B, garantem a obtenção da solução ótima. No caso específico deste trabalho, o bom desempenho obtido pelo B&B, deve-se a estratégia proposta de seleção de nós para realização da enumeração.

6 CONCLUSÕES E RECOMENDAÇÕES

Neste trabalho, foi proposto um algoritmo *Branch and Bound* para o projeto de redes de custo mínimo sujeitas a restrição de conectividade, que é um tópico relevante na área de sobrevivência de redes. Este problema está classificado como NP-Difícil, o que, em teoria, limita o uso de técnicas exatas para encontrar boas soluções em instâncias grandes. No entanto, os resultados da execução da técnica proposta com instâncias grandes com até 5000 vértices e 8567 arestas foram satisfatórios.

Neste trabalho, foram realizados testes computacionais e comparações de desempenho do método B&B com o trabalho desenvolvido em Barbosa; Rocha (2006), onde, para estes, foram utilizadas duas Heurísticas (Guloso e Guloso Aleatorizado) e uma Metaheurística (GRASP), com o método B&B sendo sempre superior aos competidores, tanto no tocante à qualidade da solução quanto no tempo de execução.

O objetivo deste trabalho foi encontrar bons resultados computacionais tanto no valor das soluções quanto no tempo de execução. O objetivo foi plenamente alcançado, a técnica utilizada apresentou bons resultados. Assim, estes resultados validam a utilização desta abordagem para a resolução do problema em sobrevivência em redes.

As recomendações e trabalhos futuros para este trabalho são:

- Desenvolver a técnica, utilizando outros métodos exatos, tais como *branch-and-cut*, *branch-and-price* entre outros, de modo a verificar a possibilidade de obtenção de tempos computacionais menores, com instâncias extremamente grandes.
- Solucionar o problema, utilizando outras metaheurísticas e comparar os resultados obtidos com os deste trabalho.
- Desenvolver uma técnica para geração das topologias na forma gráfica para facilitar a visualização.
- Aperfeiçoar o *Branch* com a combinação das duas estratégias de desdobramento da árvore de busca, a fim de aperfeiçoar a técnica.

REFERÊNCIAS

BARBOSA, R. S.; ROCHA, M. L. *Aplicações de Heurísticas a Sobrevivência de Redes*. Simpósio de Pesquisa Operacional e Logística da Marinha, 2006.

BICUDO, M. D. D. *Sobrevivência em Redes Ópticas Transparentes*. Dissertação de Mestrado, Universidade Federal do Rio de Janeiro, Engenharia de Elétrica, Rio de Janeiro, 2005.

BOVO, A. B. *Um Método de Tradução de Fontes de Informação em um Formato Padrão que Viabilize a Extração de Conhecimento por meio de Link Analysis e Teoria dos Grafos*. Dissertação de Mestrado, Universidade Federal de Santa Catarina, Engenharia de Produção, Rio Grande do Sul, 2004.

CANCELA, H.; ROBLEDO, F.; RUBINO, G. *Network Design with Node Connectivity Constraints*. Proceedings of the 2003 IFIP/ACM Latin America conference on Towards a Latin American agenda for network research. La Paz, Bolivia, 2003.

DETONI, A. A. *Sistema de Apoio à Decisão para Planejamento de Redes de Telecomunicações Baseado em Camadas*. Dissertação de Mestrado, Engenharia Elétrica, Universidade Federal do Espírito Santo, Vitória, pp. 1-13, 2001.

FERREIRA, S. M. *Algoritmos de Otimização para Redes de Telecomunicações*. Dissertação de Mestrado, Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, 2000.

GOLDBARG, M. C., LUNA, H. P. C. *Otimização Combinatória e Programação Linear*. Editora CAMPUS, Rio de Janeiro, 2000.

GRÖTSCHEL, M., MONMA, C. L. and STOER, M. *Polyhedral and Computational Investigations for Designing Communications Network with High Survivability Requirements*. Operations Research, 43(6):1012-1024, 1995.

LACHTERMACHER, G. *Pesquisa Operacional na Tomada de Decisões*. Editora CAMPUS, Rio de Janeiro, 2002.

LONGO, H. J. *Técnicas para Programação Inteira e Aplicações em Problemas de Roteamento de Veículos*. Tese Doutorado, Pontifca Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2004.

MELLO, O. D., TAVARES, H. M. F. *Sobrevivenciabilidade em Redes de Telecomunicações*. XXVIII Smpósio Brasileiro de Pesquisa Operacional, 1, 1137-1142. 1996.

NICOLLETTI, P. S. *Melhores Práticas para Gerência de Redes de Computadores*. Editora Campus Ltda, Rio de Janeiro, 2003.

ROBLEDO, F. *Optimization and Survivability of Telecommunication Networks*. Research Report LIMOS/2005, Facultad de Ingeniería, Uruguay, 2005.

STOER, M. *Design of Survivable Networks*. Volume 1531 of Lecture Notes in Mathematics. Springer Verlag, 1996.

Data de envio: 21.04.2009

Data de aceite: 13.05.2009

CENTRO UNIVERSITÁRIO UnirG 

Av. Guanabara, 1842, Centro. Telefone: (63) 3612-7619.

Cep: 77403-080. Gurupi-TO

<www.unirg.edu.br>.

This document was created with Win2PDF available at <http://www.win2pdf.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.
This page will not be added after purchasing Win2PDF.