

Uma Abordagem Híbrida GRASP-ILS para o Problema de Projeto de Redes com Topologia Anel-Estrela[†]

L.M.M.S. ANDRADE^{1*}, L.A.F. CABRAL², T.C.S. DIAS³ e E.R. PINTO³

Recebido em 24 março, 2014 / Aceito em 3 março, 2016

RESUMO. As mudanças decorrentes do crescimento das redes de telecomunicações trazem consigo a elevação dos problemas de organização, dificuldades de transmissão, localização e custo. Dentro deste cenário, o presente trabalho aborda o Problema do Ciclo Mediano sem Restrições de Capacidade (PCMRC), aplicado a uma rede de telecomunicação com topologia anel-estrela. Com o intuito de obter a menor soma resultante do custo do anel principal e do custo da atribuição dos elementos, foi utilizado um algoritmo heurístico baseado nos conceitos dos procedimentos *Greedy Randomized Adaptive Search Procedure* (GRASP) e *Iterated Local Search* (ILS). Experimentos com um conjunto de problemas de teste da literatura mostram que o algoritmo desenvolvido produz boas soluções.

Palavras-chave: projeto de redes, heurística, GRASP, ILS.

1 INTRODUÇÃO

O crescimento constante da comunicação atribuído a dinâmica da inovação tecnológica, traz consigo uma grande mudança no mercado das telecomunicações, permitindo assim grande transporte de dados em uma única estrutura de rede. Com isso, as organizações atuais buscam processos ágeis, de interfaces inteligentes e com redução de custo na fase de planejamento de novos projetos [6].

Nesse cenário, este trabalho tem como objetivo projetar uma topologia de redes anel estrela, tendo sua representação através de grafos que buscam encontrar boas soluções para o custo de projeto [13]. Esta topologia híbrida é construída através das conexões entre si das estações, constituindo um ciclo fechado, no qual algumas destas estações servirão de concentradores para outros nós em ligações ponto a ponto.

*Autor correspondente: Lisieux Marie Marinho dos Santos Andrade.

†Trabalho apresentado no XXXIV Congresso Nacional de Matemática Aplicada e Computacional.

¹Universidade Federal de Pernambuco, Centro de Informática, Cidade Universitária, 50740-560 Recife, PE, Brasil.

E-mail: lmmsa@cin.ufpe.br

²Universidade Federal da Paraíba, Centro de Informática, Campus V, 58055-000 João Pessoa, PB, Brasil.

E-mail: lucidio@ci.ufpb.br

³Centro Universitário de João Pessoa, Departamento de Informática, 58053-000 João Pessoa, PB, Brasil.

E-mails: thaysesouzadias@gmail.com; ducaribas@gmail.com

O problema descrito anteriormente é definido como Problema do Ciclo Mediano sem Restrição de Capacidade (PCMRC), em inglês *Ring Star Problem* (RSP). Formulado para determinar um ciclo simples contendo um subconjunto de vértices de um grafo, analisando separadamente dois tipos de custos: custo de roteamento associado e o custo das atribuições [9].

A formalização do PCMRC pode ser realizada através de um grafo $G = (V, E \cup A)$, sendo $V = \{v_1, v_2, \dots, v_n\}$, o conjunto de vértices; $E = \{(v_i, v_j) : v_i, v_j \in V, i < j\}$, o conjunto das arestas e $A = [v_i, v_j] : v_i, v_j \in V$, o conjunto dos arcos. Com isso, determina-se um subconjunto V' contido em V , incluindo o nó de depósito e no mínimo dois vértices pertencentes a V' . Cada aresta (v_i, v_j) está associada a um custo c_{ij} , denominado custo de roteamento, que corresponde a parte do anel da topologia. Os vértices restantes conectados aos vértices pertencentes ao ciclo geram um custo d_{ij} , denominado custo de atribuição, para cada arco $[v_i, v_j]$. O custo de roteamento é dado por:

$$CR(S) := \sum_{i=2}^m c_{v_{i-1}, v_i} + c_{v_m, v_1}, \quad (1.1)$$

e o custo de atribuição por:

$$CA(S) := \sum_{i=m+1}^n \min_{j=2, \dots, m} d_{v_i, v_j}, \quad (1.2)$$

onde v_i representa o nó depósito.

Considera-se como objetivo, minimizar a combinação convexa do custo de roteamento e atribuição, para uma solução representada por S^* :

$$CR(S^*) + CA(S^*). \quad (1.3)$$

O PCMRC é conhecido por ser NP-difícil, por tratar-se de um caso especial no qual o custo de atribuição é muito grande em comparação com o custo de roteamento revelando ser o clássico Problema do Caixeiro Viajante (PCV) [10]. O PCMRC possui atualmente poucas abordagens, destacando-se porém o trabalho de [14], em que o PCMRC é tratado através de uma heurística híbrida, denominada *Variable Neighborhood Tabu Search* (VNNTS) baseada nas nos procedimentos *Variable Neighbourhood Search* (VNS) e *Tabu Search* (TS). Outro trabalho relacionado é [3], que propôs algoritmo heurístico baseado nos conceitos dos procedimentos *Greedy Randomized Adaptive Search Procedure* (GRASP) e *General Variable Neighborhood Search* (GVNS).

Tratando instâncias com distâncias euclidianas, a heurística proposta neste trabalho é baseada nos conceitos da heurística GRASP e a heurística *Iterated Local Search* (ILS). Dentro deste contexto, o objetivo deste trabalho é comparar os resultados obtidos por [3], observando novos comportamentos das instâncias à heurística utilizada.

O presente artigo está estruturado conforme as seguintes seções. Na seção 2 são apresentados alguns trabalhos correlacionados com a pesquisa. Na seção 3 são apresentados os parâmetros para construção de uma solução viável para o PCMRC. Na seção 4 é descrita a heurística GRASP

com adaptação para o PCMRC. Na seção 5 é apresentado o procedimento ILS, também adaptado para o PCMRC. A seção 6 descreve o procedimento para a combinação da heurísticas GRASP e ILS para resolução do Problema do Ciclo Mediano sem Restrição de Capacidade. A seção 7 apresenta os resultados computacionais e na oitava e última seção são tecidas as considerações finais.

2 TRABALHOS CORRELATOS

O PCMRC, proposto por [9], é derivado dos problemas de alocação e localização estudados nos anos 60. Para sua solução, os autores utilizaram um algoritmo *branch and cut*. A partir deste trabalho, surgiram outros estudos, como o formulado por [11], o qual investigou o problema pela primeira vez de maneira bi-objetiva. Para isto, foram utilizadas três heurísticas distintas em busca de um conjunto de soluções eficientes a partir de instâncias entre 50 e 300 nós. Tal abordagem obteve bons resultados em seus experimentos.

Um algoritmo *branch-and-cut*, foi desenvolvido por [8] sendo bastante semelhante ao proposto por [9], diferenciando apenas na aplicação de certos cortes em vértices desiguais. Tal proposta foi composta por 5 etapas: a primeira de forma iterativa, similar a proposta por [9], onde realiza saltos para o cálculo do custo de cada vértice, construindo e associando subconjuntos a uma determinada lista; a segunda etapa consiste em selecionar um destes subproblemas utilizando a política do primeiro melhor, avaliando para isto o valor da função objetivo; na terceira e na quarta etapa são realizados alguns procedimentos de critério de aceitação; e na quinta e última etapa é aplicada uma estratégia de ramificação para o retorno a segunda etapa, até ser encontrada uma solução cujo o valor da função objetivo seja mínimo.

Uma versão capacitada, derivada do PCMRC, é apresentado por [7], denominado de *Capacitated m-Ring-Star Problem* (CmRSP). O problema consiste na construção de vários anéis, com limites de capacidade, para atribuições de nós na formação da topologia estrela em cada anel construído. Mantendo o objetivo de minimizar o custo do anel e das atribuições, foram elaborados dois algoritmos de programação inteira que convergem para um algoritmo *branch-and-cut*. Os procedimentos foram testados em um grande conjunto de instâncias do mundo real e comparando os resultados obtidos por [2]. O autor comprovou que seu método obteve um desempenho equivalente, tendo algumas melhorias.

Em [14] é proposta uma heurística híbrida, combinando os conceitos da heurística *Variable Neighbourhood Search* (VNS) e a *Tabu Search* (TS) aplicado ao PCMRC, cujo autor a denomina como *Variable Neighborhood Tabu Search* (VNTS). Na fase de construção foi utilizado o procedimento de TS, que realiza uma série de buscas locais utilizando uma lista tabu para o armazenamento dos nós já visitados. Após o término da construção, deu-se início o procedimento de busca local com o VNS, constituído a partir de um conjunto de movimentos aleatórios. Os experimentos foram executados em instâncias da TSLIB, com 50 à 200 vértices.

Em [3] são apresentadas três heurísticas híbridas aplicadas ao PCMRC. A primeira constituída da junção entre a heurística baseadas nos procedimentos *Simulated Annealing* (SA) e a *Tabu*

Search (TS), denominada de SABT, na qual a TS segue a explorar o espaço de vizinhança de forma iterativa após o processamento do algoritmo AS. A segunda, denominada GBT, foi construída através de junção das heurísticas baseadas nos conceitos da GRASP e TS, onde o procedimento GRASP gera uma solução inicial que será refinada pelo método TS. Por fim, a terceira forma híbrida, denominada GGVNS, é originada da junção do procedimento baseados na GRASP e na heurística GVNS, onde este último realiza o refinamento da solução obtida inicialmente pelo GRASP. Utilizando duas classes de instâncias, com distâncias Euclidianas, a primeira denominada C1 com 10, 20, 30, 50, 75 e 100 vértices e a segunda denominada C2 com 51, 52, 76, 99 e 100 vértices, foram observados bons desempenhos nas heurísticas GGVNS e GBT. Além disso, conclui-se também, que associação da heurística GRASP com uma outra heurística fez com que os resultados para o PCMRC melhorassem.

3 SOLUÇÃO INICIAL PARA O PCMRC

Uma solução viável para o PCMRC, segundo [10], é um ciclo fechado constituído por um nó depósito v_1 e no mínimo dois outros vértices. O critério de escolha para adição de cada vértice v_j obedece a Equação 3.1, em que verifica-se a melhor média de conectividade para cada v_j .

$$conectividade_{media} = \frac{\sum_{i=2, i=j}^m d_{v_j, v_i}}{m - 2}. \tag{3.1}$$

Como representado também na Figura 1, em que o conjunto de vértices $V = \{1, 2, 3\}$, formam as arestas $E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$ que compõem o ciclo básico para o PCMRC.

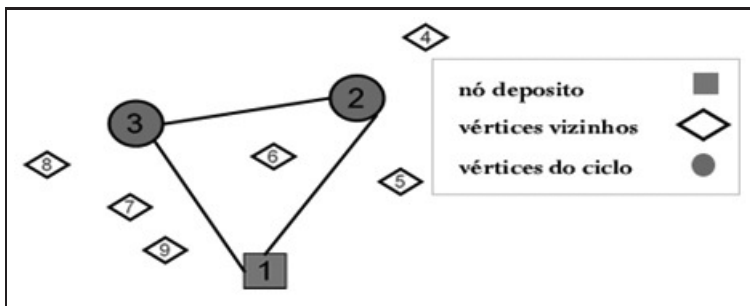


Figura 1: Representação de um ciclo válido para o PCMRC.

Caso o conjunto de vértice V utilizado possua mais de três vértices, estes são adicionados ao ciclo, de forma que a média de conectividade do ciclo, calculada através da equação 3.2, seja mínima, contribuindo assim para o valor do custo de roteamento (Equação 1.1).

$$incremento(v_i) + \frac{\sum_{j \in (V \setminus S)} d_{v_i, v_j}}{|V \setminus S|}. \tag{3.2}$$

4 HEURÍSTICA GRASP PARA O PCMRC

A heurística GRASP, proposta por [5] é um método *multi-start* para problemas combinatórios, sua iteração consiste de duas fases: a fase de construção e a fase de busca local. A fase de construção é realizada de forma randômica, sendo um processo iterativo que analisa cada elemento gerado através de uma função adaptativa gulosa, que valida a seleção de cada elemento que poderá compor a solução corrente. Este método possibilita a geração de várias soluções GRASP diferentes a cada iteração [15].

A fase de busca local é aplicada com a finalidade de aprimorar a solução gerada na construção inicial. Trabalhando de maneira iterativa em busca de uma melhor solução em sua vizinhança. O método é finalizado ao não encontrar melhor solução no espaço de vizinhança.

O procedimento GRASP utilizado neste trabalho conta com uma lista de componentes candidatos (vértices) para solução, denominada Lista de Candidatos Restrita (LCR) [4]. A escolha de cada vértice pertencente a solução básica para o PCMRC, conforme a seção anterior, é estabelecida através da Equação 3.1, e a escolha dos demais vértices é determinada através da Equação 3.2, as escolhas são realizadas de forma aleatória, porém não no conjunto de vértices e sim na LCR.

A classificação dos vértices candidatos ocorre através do cálculo do seu custo definido como:

$$g(v_i) \leq g_{\min} + \alpha(g_{\max} - g_{\min}) \quad (4.1)$$

Onde α corresponde ao fator de gulosidade, que varia no intervalo de $[0, 1]$, e g_{\min} e g_{\max} , são determinados através da seguinte equação:

$$g_{\min} = \min \{g(v_i) \mid v_i \in V \setminus S\} \text{ e } g_{\max} = \max \{g(v_i) \mid v_i \in V \setminus S\} \quad (4.2)$$

Diferente da metodologia padrão GRASP, em que a cada iteração uma única solução é gerada, foram utilizadas várias iterações com o objetivo de produzir n soluções iniciais de maneira diversificada, estabelecidas pela variável *MaxIterFiltro*, para a escolha da melhor solução inicial.

Após a construção da solução inicial, representada no pseudo-código da Figura 2, é executado o procedimento de busca local, realizado pela heurística ILS.

5 HEURÍSTICA ILS PARA O PCMRC

O método ILS proposto por [12], é uma dos procedimentos mais simples na pesquisa operacional [1]. Sua abordagem consiste perturbar as soluções iniciais classificadas como boas, a fim de melhorar as soluções ótimas locais.

A função de perturbação utilizada neste trabalho é estruturada em três níveis, utilizando a exploração do espaço de vizinhança na forma de pequenos movimentos na solução S com o objetivo de construção de novas soluções. Para isto, foram utilizados três movimentos: adição, remoção e troca.

O movimento de adição (*add*) consiste em inserir um vértice v_j , antes pertencente a uma ligação ponto-a-ponto, à outro vértice v_x , no ciclo constituído por n vértices. Para isto é analisado o custo

procedimento ConstruçãoSolução(MaxIterFiltro, $g(\cdot)$, α)

1. $f(s^*) \leftarrow \infty$;
2. **para** IterFiltro $\leftarrow 1$ ate MaxIterFiltro **faça**
3. $s \leftarrow \text{BasicoCiclo}(v_1)$;
4. **enquanto** a condição de parada não for satisfeita **faça**
5. $g_{min} = \min\{g(v_i) \mid v_i \in V \setminus S\}$;
6. $g_{max} = \max\{g(v_i) \mid v_i \in V \setminus S\}$;
7. $L = \{v_i \in V \setminus S \mid g(v_i) \leq g_{min} + \alpha(g_{max} - g_{min})\}$;
8. Selecione aleatoriamente um vértice $v_i \in L$;
9. $s = s \cup \{v_i\}$;
10. $V = V \setminus \{v_i\}$;
11. **fim-enquanto**
12. **se** ($f(S) < f(s^*)$) **então**
13. $s^* \leftarrow S$;
14. $f(s^*) \leftarrow f(s)$;
15. **fim-se**
16. **fim-para**
17. retorne(s^*);

fim ConstruçãoSolução.

Figura 2: Pseudo-código para a fase de construção GRASP.

de redução roteamento adquirido por sua adição em certa posição, tendo-se como referência para análise seu antecessor no ciclo, o vértice v_k , conforme as equações a seguir:

$$CR(S_{kj}) := CR(S) + d_{v_k, v_j} + d_{v_j, v_{k+1}} - d_{v_k, v_{k+1}}, \quad (5.1)$$

para $k = 1, \dots, m - 1$, e

$$CR(S_{kj}) := CR(S) + d_{v_m, v_j} + d_{v_j, v_1} - d_{v_m, v_1}, \quad (5.2)$$

para $k = m$.

O critério de aceitação para a inclusão do vértice v_j ao ciclo, é a verificação do valor de $CR(S_{kj})$ (com j vértices candidatos para a inclusão), sendo escolhido o de menor valor. O custo de atribuição é realizado de forma a considerar o melhor valor, obtido da através da Equação 1.2. Como pode ser observado na representação da adição do vértice 7, Figura 3, ao ciclo composto pelo conjunto de vértices $V = \{1, 2, 3\}$, onde o vértices 1 é o nó depósito.

O segundo movimento utilizado no procedimento de perturbação é o de remoção de um vértice do ciclo (*drop*), neste é verificado o custo de roteamento para a remoção do vértice v_k , conectado ao vértice $v_k - 1$ e $v_k + 1$, conforme as equações a seguir:

$$CR(S_k) := CR(S) + d_{v_{m-1}, v_1} - d_{v_{m-1}, v_m} - d_{v_m, v_1}, \quad (5.3)$$

para $k = m$.

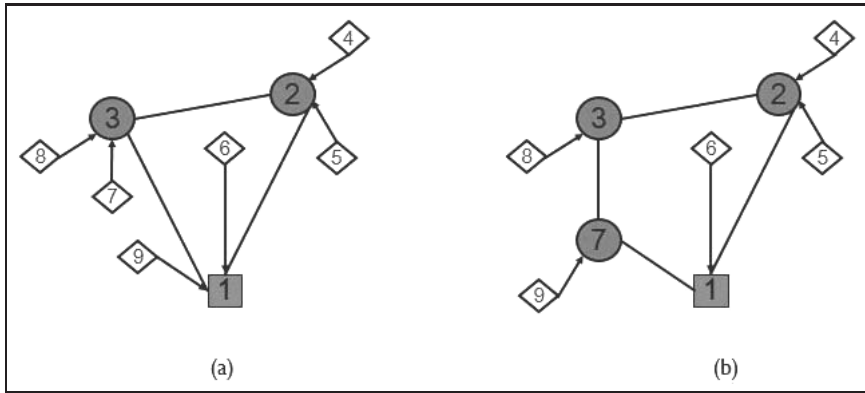


Figura 3: Representação do movimento de adição (*add*).

A escolha do menor valor para $LC(S_k)$ é realizada para a reconexão dos vértices $v_k - 1$ e $v_k + 1$ ao seu vizinho mais próximo, como também a atribuição do vértice v_k a um outro vértice, cujo o custo seja mínimo, em uma ligação ponto-a-ponto. Para assim ser recalculado o valor do novo custo de atribuição.

Fazendo uso do mesma mesma estrutura na representação do movimento *add*, pode ser observado o comportamento do procedimento de remoção (Fig. 4) ao extrair o nó 3, antes pertence ao ciclo na Figura 3.

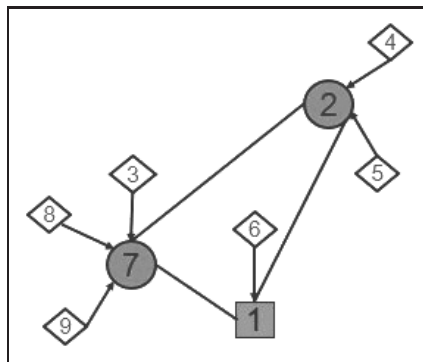


Figura 4: Representação do movimento de remoção (*drop*).

O último movimento utilizado é ação de troca entre dois vértices (*swap*), combinando um movimento de adição ao ciclo com um de remoção do mesmo, utilizando os procedimentos de *drop* e *add*, descritos anteriormente, analisando a redução simultânea do custo de atribuição e do custo de roteamento, como descrito na equação a seguir:

$$CA(S_{ij}) := \min \{d_{v_i, v_j}, \min_{l=1, \dots, m_l \neq i} d_{v_i, v_l}\} + \sum_{k=m+1}^n \min \{d_{v_k, v_j}, \min_{l=1, \dots, m_l \neq i} d_{v_k, v_l}\} \quad (5.4)$$

Seu comportamento pode ser observado na representação da Figura 5, que segue a mesma estrutura da Figura 4, e realiza a troca dos elementos 2 e 5, proporcionando a redução simultânea do custo de atribuição e do custo de roteamento.

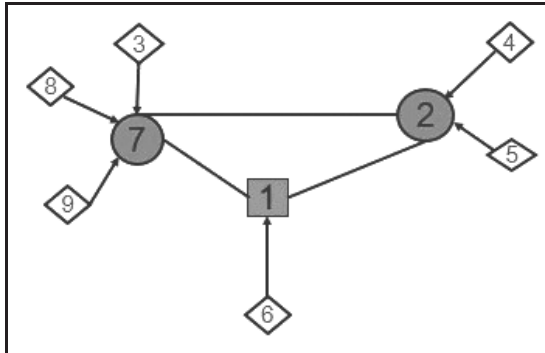


Figura 5: Representação do movimento de troca (*swap*).

Na tentativa de melhorar os resultados obtidos, foi utilizado uma estratégia de perturbação por nível. Em que a solução S ao entrar no fluxo de perturbação, passaria por um conjunto determinado de movimentos, pertencente ao primeiro nível de perturbação, e a medida que fosse atingido um número máximo de tentativas de melhora ($MaxTentILS$), caso o valor da função objetivo de S não fosse melhorado, esta solução seria submetida ao próximo nível de perturbação, como será descrito na próxima seção.

6 HIBRIDIZAÇÃO GRASP-ILS

A proposta híbrida baseada nas heurísticas GRASP e ILS consiste na aplicação da heurística ILS, apresentada na seção 5, na segunda fase posterior ao método de busca local da heurística GRASP, apresentada na seção 4. Como pode ser observado na Figura 6.

A fase de construção da metodologia GRASP é iterativa, de forma que o próximo elemento a ser selecionado depende do anterior escolhido. Já a fase de construção da metodologia ILS é o resultado da primeira fase da metodologia GRASP, que logo é submetido à fase de perturbação ILS.

O procedimento de perturbação utilizado na metodologia ILS consistiu em três níveis. No primeiro nível foram realizados dois movimentos de troca (*swap*) e um movimento de exclusão de um elemento do anel (*drop*). No segundo nível foram realizados cinco movimentos de *swap*, uma adição de um elemento no anel (*add*) e um movimento de *drop*; Por fim, no terceiro nível foram realizados sete movimentos de *swap*, três movimentos de *add* e dois movimentos de *drop*. Vale salientar que as escolhas dos elementos que participaram de tais movimentos foram realizadas de forma aleatória e a passagem de um nível de perturbação para outro ocorram ao atingir o número máximo de tentativas por nível, previamente estabelecido.


```

procedimento GRASP-ILS(MaxIterGRASP, MaxIterFiltro,
                        NumMaxTentativaILS, MaxIterILS )
1.   $so \leftarrow$  GRASP(MaxIterGRASP, MaxIterFiltro, s);
2.   $s \leftarrow$  BuscaLocal(so);
3.  enquanto  $i < \text{MaxIterILS}$  faça
4.     $s' \leftarrow$  Pertuba(s);
5.     $s'' \leftarrow$  BuscaLocal(s');
6.    se ( $f(s'') < f(s')$ ) faça
7.       $s \leftarrow s''$ ;
8.       $f(s) \leftarrow f(s'')$ ;
9.    fim-se
10. fim-enquanto
11. retorna(s);
fim GRASP-ILS.

```

Figura 6: Pseudo código GRASP-ILS.

A finalização de cada iteração GRASP-ILS é determinada após a verificação de melhoria do resultado obtidos da metodologia ILS. A execução da hibridização chega ao fim, após ser atingido o máximo de iterações GRASP (MaxIterGRASP).

7 RESULTADOS COMPUTACIONAIS

Esta seção apresenta os resultados obtidos através da implementação da heurística híbrida GRASP-ILS para o PCMRC. Os experimentos computacionais foram executados em um processador Intel Quad Core, tendo 8GB de memória RAM, sob um sistema operacional Windows 7. A linguagem utilizada para a implementação foi a C++, utilizando a IDE DEV C/C++ em sua versão 4.1.

As instâncias utilizadas pertencem a uma das duas categorias estudadas por [3], denominada C1, que envolvem instâncias de 10, 20, 30, 40, 50 e 75 vértices em formato EUC2D (distâncias Euclidianas). Cada instância possui 10 casos, que são abordados por três valores distintos de β , totalizando assim 180 casos. Nos cálculos do custo do roteamento e das atribuições, definidos respectivamente nas equações apresentadas a seguir. Foi empregada a mesma técnica que [3] e [14], cujos valores de $\beta \in \{5, 7, 9\}$ e l_{ij} correspondem à distância Euclidiana entre os vértices v_i e v_j .

$$c_{ij} = \beta \cdot l_{ij} \quad (7.1)$$

$$d_{ij} = (10 - \beta) \cdot l_{ij}. \quad (7.2)$$

Cada um dos 180 casos das instâncias utilizadas foram executados 5 vezes, escolhendo-se assim o melhor valor obtido. Cada iteração foi executada separadamente, como um processo único, para obter melhor desempenho.

Foram realizados testes computacionais, para a determinação dos valores dos parâmetros de execução, analisando o desempenho dos algoritmos e verificando seu tempo computacional. Para a heurística GRASP, na hibridização GRASP-ILS, o número máximo de iterações (MaxIter-GRASP) foi definido como 100 e o número máximo de iterações do filtro na fase de construção (MaxIterFiltro) 25. No que se refere ao parâmetro *alpha* (α), inicialmente foram aplicados os valores 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, e 0.95. Após a realização das execuções, observou-se que os melhores resultados foram obtidos para valores fixo de *alpha*, sendo adotado os valores apresentados na Tabela 1. Em decorrência disso, os demais valores de alfa foram desconsiderados.

Tabela 1: Valores do parâmetro *alpha*, de acordo com o número de vértices das instâncias.

Instância	α
10 vértices	0.95
20 vértices	0.8
30 vértices	0.4
40 vértices	0.7
50 vértices	0.3
75 vértices	0.3

Na heurística ILS, o número máximo de iterações (MaxIterILS) foi determinado por 70% de acordo com o número de vértices de cada instância.

Para análise dos resultados, em ordem, serão apresentados os nomes das instâncias; os valores de β correspondentes; os resultados de cada heurística, com seus valores mínimos da função objetivo (f_{min}); e o tempo mínimo de execução (t_{min}) em milisegundos, coletado através do *clock* do processador utilizado na execução dos testes. Na última coluna, é apresentado o valor correspondente ao GAP, dado equivalente à diferença entre os valores mínimos obtidos da função objetivo, em que o cálculo realizado foi baseado no mesmo procedimento adotado por [3], conforme pode ser observado nas equações 7.3, que equivale a coluna GAP.

$$\frac{f_{\min_{GGVNS}} - f_{\min_{GRASP-ILS}}}{f_{\min_{GRASP-ILS}}} \quad (7.3)$$

Analisando os resultados computacionais das instâncias de 10, 20 e 30 vértices, apresentados nas Tabelas 2 e 3, pode-se verificar que em 81 dos 90 casos de testes, as duas heurísticas apresentaram mesmos resultados computacionais (GAP= 0), tendo a hibridização GRASP-ILS demonstrado 6 novos valores, correspondentes aos GAPs positivo, para o PCMRC em comparação ao GGVNS.

Verificando os resultados das instâncias com maior número de vértices, 40, 50 e 75, presentes nas Tabelas 3 e 4, a heurística GRASP-ILS obteve bons resultados, contudo foi com instâncias de 75 vértices que o procedimento obteve o maior número de sucessos, 14 no total.

Observando os resultados coletados das execuções dos experimentos, de maneira geral, a GRASP-ILS obteve 31 casos de sucesso, equivalente aos valores de GAP positivo, comparado a GGVNS.

Tabela 2: GAP entre GRASP-ILS e GGVNS para instâncias com 10 e 20 vértices.

Instância	β	GGVNS	GRASP-ILS	GAP	Instância	β	GGVNS	GRASP-ILS	GAP
		f_min	f_min				f_min	f_min	
f10_1.rcp	5	12295	12295	0	f20_1.rcp	5	16195	16195	0
	7	12212	12212	0		7	16295	16295	0
	9	8364	8364	0		9	9100	9100	0
f10_2.rcp	5	13600	13600	0	f20_2.rcp	5	17075	17075	0
	7	14239	14239	0		7	19946	19946	0
	9	8083	8083	0		9	12117	12085	0,002648
f10_3.rcp	5	12315	12315	0	f20_3.rcp	5	16275	16275	0
	7	12307	12307	0		7	19444	19444	0
	9	8459	8459	0		9	15967	15967	0
f10_4.rcp	5	14485	14485	0	f20_4.rcp	5	19315	19315	0
	7	15247	16194	-0,05848		7	20517	20517	0
	9	11029	11029	0		9	11540	12138	-0,04927
f10_5.rcp	5	14705	14705	0	f20_5.rcp	5	16140	16140	0
	7	16754	16754	0		7	16298	16298	0
	9	11338	11338	0		9	8835	8835	0
f10_6.rcp	5	15535	15495	0,002581	f20_6.rcp	5	18045	18045	0
	7	15521	15521	0		7	19818	19818	0
	9	9420	9420	0		9	12721	12721	0
f10_7.rcp	5	14455	14455	0	f20_7.rcp	5	17305	17305	0
	7	15282	15282	0		7	20520	20520	0
	9	10926	11954	-0,086		9	11581	11581	0
f10_8.rcp	5	13525	13525	0	f20_8.rcp	5	16045	16045	0
	7	13929	13929	0		7	18615	18615	0
	9	10323	10323	0		9	10860	10860	0
f10_9.rcp	5	16475	16475	0	f20_9.rcp	5	16085	16085	0
	7	15178	14684	0,033642		7	17202	17202	0
	9	11846	11846	0		9	11352	11352	0
f10_10.rcp	5	13050	13050	0	f20_10.rcp	5	18420	18420	0
	7	13574	13574	0		7	19688	19688	0
	9	8898	8898	0		9	12535	12535	0

Desta forma, tais resultados revelam que a GGVNS não obteve soluções ótimas globais, como comprova este trabalho, apresentando a comunidade acadêmica novos valores para a resolução do PCMRC. Além disso revela uma boa coleta de resultados, já que em seu trabalho [3], obteve 43 melhores resultados em 210 casos de teste, trabalhando com instâncias de até 100 vértices, já o presente trabalho utilizou 180 casos de teste, com até 75 vértices, obtendo sucesso em 31 casos, revelando assim uma maior parcela de resultados, em proporção, comparado a [3].

8 CONSIDERAÇÕES FINAIS

Este trabalho propôs uma hibridização baseada nos conceitos das heurísticas *Greedy Randomized Adaptive Search Procedure* (GRASP) e *Iterated Local Search* (ILS), aplicada ao Problema do Ciclo Mediano sem Retrição de Capacidade (PCMRC), que tem por objetivo construir uma rede com topologia anel-estrela.

Tabela 3: GAP entre GRASP-ILS e GGVNS para instâncias com 30 e 40 vértices.

Instância	β	GGVNS	GRASP-ILS	GAP	Instância	β	GGVNS	GRASP-ILS	GAP
		f_{\min}	f_{\min}				f_{\min}	f_{\min}	
f30_1.rcp	5	19605	19605	0	f40_1.rcp	5	24360	24360	0
	7	22578	22578	0		7	27431	27431	0
	9	14052	14052	0		9	17792	17792	0
f30_2.rcp	5	19635	19635	0	f40_2.rcp	5	22940	22940	0
	7	22458	22458	0		7	25255	25255	0
	9	16292	16292	0		9	17590	17590	0
f30_3.rcp	5	21335	21335	0	f40_3.rcp	5	23545	23545	0
	7	25109	24793	0,012746		7	25789	25792	-0,000116
	9	13435	13435	0		9	17530	17530	0
f30_4.rcp	5	18040	18040	0	f40_4.rcp	5	24585	24585	0
	7	20460	20460	0		7	27517	27517	0
	9	12574	12574	0		9	17145	17145	0
f30_5.rcp	5	21520	21520	0	f40_5.rcp	5	24345	24410	-0,002663
	7	24825	24825	0		7	28268	28268	0
	9	16854	16701	0,009161		9	21159	21159	0
f30_6.rcp	5	21905	21905	0	f40_6.rcp	5	24380	24380	0
	7	24807	24807	0		7	27491	27480	0,0004003
	9	16708	16708	0		9	19143	19143	0
f30_7.rcp	5	19655	19655	0	f40_7.rcp	5	21905	21905	0
	7	22471	22383	0,003932		7	25974	25973	0,0000385
	9	15368	15368	0		9	21584	21584	0
f30_8.rcp	5	22220	22220	0	f40_8.rcp	5	23665	23665	0
	7	22440	22440	0		7	26842	26830	0,0004473
	9	13401	13401	0		9	18461	18461	0
f30_9.rcp	5	19845	19845	0	f40_9.rcp	5	24980	24980	0
	7	22718	22718	0		7	29393	29393	0
	9	15026	15026	0		9	24120	24120	0
f30_10.rcp	5	19675	19675	0	f40_10.rcp	5	24395	24395	0
	7	22791	22791	0		7	26377	26323	0,0020514
	9	15279	15279	0		9	17400	17400	0

O procedimento baseado no conceito da heurística GRASP, realizou um processo de busca local iterativa e um procedimento de construção parametrizado através do uso de uma Lista de Candidatos Restrita (LCR). O procedimento baseado nos conceitos da heurística ILS, teve como objetivo explorar o espaço de vizinhança, realizando alguns movimentos de troca, adição e remoção de elementos, a fim de reduzir o custos do anel e dos elementos associados a este.

Para as experimentações, foram utilizados 180 casos de instâncias manipuladas por [3], cujo os resultados foram comparados aos seus experimentos com a heurística GGVNS (método baseado na GRASP com a heurística *General Variable Neighborhood Search*).

Os resultados colhidos revelam que o procedimento GRASP-ILS apresenta bom desempenho, com 31 sucessos em 180 casos de teste, proporcionando assim novos resultados para a resolução do PCMRC.

Tabela 4: GAP entre GRASP-ILS e GGVNS para instâncias com 50 e 75 vértices.

Instância	β	GGVNS	GRASP-ILS	GAP	Instância	β	GGVNS	GRASP-ILS	GAP
		f_{\min}	f_{\min}				f_{\min}	f_{\min}	
f50_1.rcp	5	28100	28145	-0,0016	f75_1.rcp	5	33550	33545	0,000149
	7	30049	30020	0,000966		7	38069	37974	0,002502
	9	22942	22942	0		9	27479	27479	0
f50_2.rcp	5	28510	28510	0	f75_2.rcp	5	30745	30800	-0,00179
	7	30629	30629	0		7	35032	35119	-0,00248
	9	22081	22081	0		9	26972	27043	-0,00263
f50_3.rcp	5	28715	28535	0,006308	f75_3.rcp	5	33285	33345	-0,0018
	7	32555	32129	0,013259		7	37153	37100	0,001429
	9	22300	22300	0		9	27443	27482	-0,00142
f50_4.rcp	5	24735	24800	-0,00262	f75_4.rcp	5	31285	31335	-0,0016
	7	28788	28788	0		7	34750	34583	0,004829
	9	20778	20778	0		9	25946	25855	0,00352
f50_5.rcp	5	26950	26950	0	f75_5.rcp	5	29840	29840	0
	7	29268	29268	0		7	34024	33726	0,008836
	9	23585	23585	0		9	28772	28772	0
f50_6.rcp	5	26225	26225	0	f75_6.rcp	5	32450	32450	0
	7	29857	29841	0,000536		7	37508	37122	0,010398
	9	20709	20709	0		9	26041	25962	0,003043
f50_7.rcp	5	26095	26095	0	f75_7.rcp	5	31855	31845	0,000314
	7	29901	29901	0		7	35524	35590	-0,00185
	9	22127	22127	0		9	28178	28178	0
f50_8.rcp	5	27200	27195	0,000184	f75_8.rcp	5	30950	30960	-0,00032
	7	32643	32603	0,001227		7	35399	35356	0,001216
	9	26564	26564	0		9	26565	26543	0,000829
f50_9.rcp	5	26900	26905	-0,00019	f75_9.rcp	5	30825	30825	0
	7	29261	29438	-0,00601		7	34048	34028	0,000588
	9	22282	22282	0		9	24957	25028	-0,00284
f50_10.rcp	5	27890	27890	0	f75_10.rcp	5	31030	31125	-0,00305
	7	28832	28651	0,006317		7	34981	34877	0,002982
	9	21715	21825	-0,00504		9	28434	28380	0,001903

Como extensões deste trabalho, propõe-se a mudança no procedimento de busca local adotado, a utilização de um número maior de instâncias e a execução de todos os procedimentos em mesmo ambiente, para também ser avaliado o desempenho dos procedimentos quanto ao tempo de execução.

ABSTRACT. The changes that came with the development of telecommunication networks have increased some problems such as organization, transmission difficulties, location, and cost issues. In this scenario, the present work approaches the Ring Star Problem (RSP) applied to a ring star topology telecommunication network. With the objective of achieving the lowest net sums of the main ring cost and of the element assignment cost, it is used a heuristic algorithm based on the concepts of Greedy Randomized Adaptive Search Procedure (GRASP) and Iterated Local Search (ILS). Some experiments have been performed with a set of test

problems in the literature and results have shown that the presented algorithm provides good solutions.

Keywords: network project, heuristic, GRASP, ILS.

REFERÊNCIAS

- [1] R. Andrea, M. Blesa, C. Blum & S. Michael. Hybrid metaheuristics – an emerging approach to optimization, (2008).
- [2] R. Baldacci, M. Dell’Amico & J.S. González. The capacitated m-ring-star problem. *Operations Research*, **55**(6) (2007), 1147–1162.
- [3] T.C.S. Dias, G.F. Souza Filho, E.M. Macambira, L.A.F. Cabral & M.H. Fampa. An Efficient Heuristic for the Ring Star Problem. *Lecture Notes in Computer Science*, **4007** (2006), 24–35.
- [4] A. Enrique. Hybrid metaheuristics: a new class of algorithms. Canada: John Wiley & Sons. Wiley series on parallel and distributed computing (2005).
- [5] T. A. Feo, M. G. C Resende. Greedy Randomized Adaptive Search Procedures. *Journal of Global Optimization*, **6** (1995), 109–133.
- [6] M.T. Furtado, A.C. Rego & C.A. Loural. Prospecção Tecnológica e Principais Tendências em Telecomunicações. *Cadernos CPqD Tecnologia*, **1** (2005).
- [7] E.A. Hoshino & C.C. Souza. A Branch-and-Cut-and-Price Approach for the Capacitated m-Ring-Star Problem. *V Latin-American Algorithms, Graphs and Optimization Symposium – LAGOS*, **35** (2009), 103–108.
- [8] S. Kedad-Sidhoum & V.H. Nguyen. An Exact Algorithm for solving the ring star problem. *Optimization A Journal of Mathematical Programming and Operations Research*, **59** (2008), 125–140.
- [9] M. Labbé, G. Laporte, I.R.Martín & J.J.S. González. Median Cycle Problem, Technical Report. Department of Operations Research and Multicriteria Decision Aid at Université Libre de Bruxelles, (2011).
- [10] M. Labbé, G. Laporte, I.R.Martín & J. J.S. González. The Ring Star Problem: Polyhedral Analysis and Exact Algorithm. *Networks*, **43** (2001), 177–189.
- [11] A. Liefooghe, L. Jourdan, M. Basseur, E. Talbi & E.K. Burke. Metaheuristics for the Bi-objective Ring Star Problem. Eighth European Conference on Evolutionary Computation in Combinatorial Optimisation (2008).
- [12] H.R. Lourenço, O.C. Martin & T. Stützle. Iterated Local Search. *Handbook of Metaheuristics*, (2002), 321–353.
- [13] E.M. Macambira, C.C. Souza, N.M. Filho, L.S. Ochi & L.O. Bastos. Algoritmos Eficientes para o Projeto de uma Rede de Telecomunicações com Topologia em Anel. 5º Encontro Regional de Matemática Aplicada e Computacional – V ERMAC-R3 (2005), 19–21.
- [14] J.A.M. Pérez, J.M. Moreno-Vega & I.R. Martin. Variable Neighbourhood Tabu Search and its Application to the Median Cycle Problem. *European Journal of Operational Research*, **151** (2003), 365–378.
- [15] M.G.C. Resende & C.C. Ribeiro. Greedy Randomized Adaptive Search Procedures. *Handbook of metaheuristics*, **57** (2003), 219–249.