Meta-heurísticas para o Problema da Árvore Geradora com Rotulação Mínima e Restrições Orçamentárias

BIANCA LIMA DE CARVALHO OLIVEIRA (IFPB, Campus João Pessoa), KERVEN MACIEL MONTEIRO DE ALBUQUERQUE (IFPB, Campus João Pessoa), EMANUEL EDUARDO OLIVEIRA DIONÍSIO (IFPB, Campus João Pessoa), THIAGO GOUVEIA (IFPB, Campus João Pessoa)

E-mails: carvalho.bianca@academico.ifpb.edu.br, albuquerque.kerven@academico.ifpb.edu.br, emanuel.eduardo@academico.ifpb.edu.br, thiago.gouveia@ifpb.edu.br

Área de conhecimento:(Tabela CNPq): 1.03.03.04-9 Sistemas de Informação.
Palavras-Chave: grafos com arestas rotuladas; árvores geradoras; GRASP; BRKGA.

1 Introdução

Neste trabalho, propomos uma variação do Problema da Árvore Geradora com Rotulação Mínima (PAGRM), introduzido por Chang e Leu (1997), que tem como objetivo encontrar uma árvore geradora minimizando o número de rótulos utilizados. A variação proposta, denominada Problema da Árvore Geradora com Rotulação Mínima e Restrições Orçamentárias (PAGRMRO), impõe restrições ao número de arestas que podem ser utilizadas por rótulo.

O PAGRM possui várias aplicações práticas, como o projeto de redes de comunicação, redes de transporte multimodal e compressão de dados. Por exemplo, em redes de comunicação, busca-se conectar diversos dispositivos utilizando o menor número possível de tipos de meio de comunicação, reduzindo tanto os custos quanto a complexidade da rede (LIN et al., 2020). A introdução de restrições no PAGRMRO torna o problema mais aplicável a cenários reais, garantindo que a solução esteja adequada ao orçamento disponível.

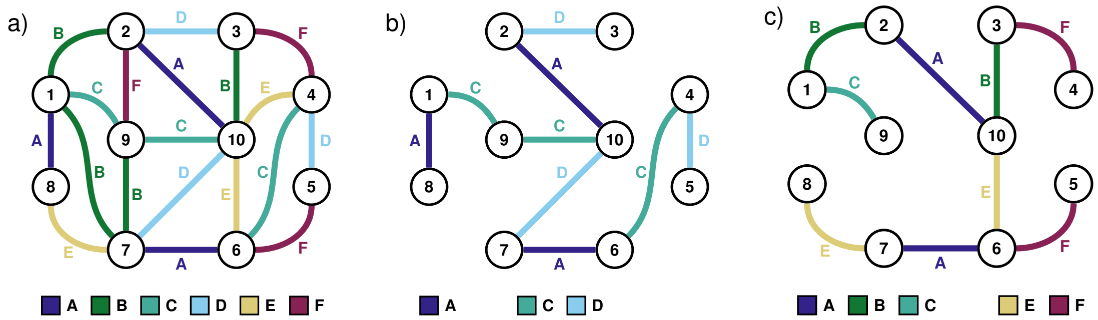

Figura 1: Grafo com arestas rotuladas com 10 vértices e 6 rótulos (a); solução do PAGRMRO com restrição de até 3 arestas por rótulo (b); e solução do PAGRMRO com restrição de até 2 arestas por rótulo (c)

A Figura 1(a) ilustra um grafo com arestas rotuladas, composto por 10 vértices, 19 arestas e 6 rótulos distintos. Cada rótulo é identificado por uma letra maiúscula e pela cor correspondente da aresta. Para esse grafo, a Figura 1(b) apresenta uma solução do PAGRMRO considerando uma restrição de até 3 arestas por rótulo. Nessa solução, o conjunto de rótulos utilizados é $L' = \{A,C,D\}$, resultando em um custo $|L'| = 3$. Já a Figura 1(c) mostra outra solução para o mesmo grafo, agora com uma restrição mais rígida de no máximo 2 arestas por rótulo. Nesse cenário, a solução obtida utiliza os rótulos $L' = \{A,B,C,E,F\}$, com custo $|L'| = 5$.

Foram propostos dois métodos heurísticos, baseados nas meta-heurísticas GRASP e BRKGA, além de uma formulação exata resolvida via algoritmo de branch-and-cut. Os resultados mostraram que o GRASP encontrou soluções ótimas nas instâncias resolvidas pelo método exato dentro do tempo limite, superando o BRKGA na maioria dos casos.

2 Materiais e Métodos

Do ponto de vista matemático, seja um grafo com arestas rotuladas $G = (V,E,L)$, onde V é o conjunto de vértices, E o conjunto de arestas e L o conjunto de rótulos. Cada aresta $e \in E$ possui um rótulo associado $\ell(e) \in L$. Além disso, considere uma função de orçamento $w : L \to \mathbb{Z}^+$, que define o número máximo de arestas permitidas para cada rótulo. O objetivo do PAGRMRO é encontrar uma árvore geradora $T = (V,E',L')$ que minimize a quantidade $|L'|$ de rótulos distintos em T, garantindo que $|\mathscr{E}_T(l)| \leq w(l)$ para todo $l \in L$, onde $\mathscr{E}_T(l) = \{e \in E' \mid \ell(e) = l\}$ representa o conjunto de arestas em T com rótulo l. Um exemplo de instância é ilustrado na Figura 1.

O PAGRMRO é um problema NP-difícil, por se tratar de uma generalização do PAGRM, que também é NP-difícil (CHANG; LEU, 1997). Para abordá-lo, propomos dois métodos heurísticos baseados nas meta-heurísticas GRASP (do inglês, *Greedy Randomized Adaptive Search Procedure*) e BRKGA (do inglês, *Biased Random-Key Genetic Algorithm*), além de uma formulação baseada em cortes, com um algoritmo de *branch-and-cut*, para validar os métodos heurísticos.

2.1 GRASP

Neste trabalho, propomos um método heurístico baseado na meta-heurística GRASP para resolver o PAGRMRO. O GRASP consiste em uma construção gulosa aleatorizada e uma busca local, e novas soluções são geradas até um critério de parada ser atingido (SAAD et al., 2018). Neste caso, o critério é o tempo limite.

Na fase de construção, a solução é formada incrementalmente a partir de uma floresta inicial. Os rótulos são ordenados conforme a contribuição de suas arestas para a conectividade da floresta, considerando também as restrições orçamentárias. Um subconjunto dos rótulos mais promissores é selecionado com base no parâmetro $\alpha \in [0,1]$, que representa o nível de aleatoriedade do GRASP. As arestas correspondentes são adicionadas aleatoriamente, desde que não formem ciclos e respeitem os limites orçamentários. Caso não seja possível formar uma árvore viável, a construção é abortada.

A busca local aprimora a solução por meio de dois mecanismos: remoção de um rótulo, seguida pela reconstrução da solução, ou substituição de dois rótulos por um novo. Cada tentativa gera uma nova solução do zero, utilizando as arestas dos rótulos remanescentes. O número máximo de tentativas de melhoria é controlado pelo parâmetro $n_T \in \mathbb{Z} \geq 1$, que define quantas vezes cada tipo de operação é repetida para explorar melhorias. A primeira melhoria encontrada é aceita, e o processo se repete até atingir esse limite.

2.2 BRKGA

A segunda heurística proposta baseia-se na meta-heurística BRKGA. Nesse método, cada solução é representada por um cromossomo, modelado como um vetor de n números reais no intervalo $[0,1]$, chamados genes. A evolução da população ocorre por meio de recombinação entre cromossomos e da introdução de mutações aleatórias, sendo que um algoritmo decodificador associa cada cromossomo a uma solução viável do problema. O processo segue uma abordagem elitista, mantendo os melhores indivíduos da população e favorecendo, na recombinação, a herança dos genes provenientes da elite (GONÇALVES; ALMEIDA, 2002; ERICSSON; RESENDE; PARDALOS, 2002).

Inicialmente, a população é gerada aleatoriamente e evolui ao longo das gerações até atingir o tempo limite. Em cada geração, a elite é mantida, novos indivíduos são criados por recombinação enviesada e mutantes são introduzidos. Cada cromossomo é decodificado em uma solução viável por meio da ordenação dos rótulos conforme as prioridades definidas pelos genes. A melhor solução obtida pela população é atualizada e preservada até o fim do processo.

3 Resultados e Discussão

Os experimentos, realizados em C++ com o solver CPLEX, avaliaram dois métodos heurísticos para o PAGRMRO em 1080 instâncias com diferentes características, seguindo a proposta de Cerulli et al. (2005). Nos testes com instâncias menores, o método baseado em GRASP se destacou ao obter os melhores resultados em todos os cenários, como mostra a Tabela 1. A Figura 2 ilustra a convergência do GRASP em uma instância representativa, evidenciando sua proximidade constante do ótimo, impulsionada por uma busca local eficaz — responsável por melhorias em mais de 80% dos casos. Em contraste, o BRKGA apresentou maior diversidade de soluções, porém com menor capacidade de convergência.

Para as instâncias maiores, o método exato não foi aplicado, já que não conseguiu convergir dentro do tempo limite de uma hora nas instâncias menores. As heurísticas foram avaliadas com tempo limite de 60 segundos, e o GRASP continuou a apresentar melhor desempenho na maioria dos cenários, mesmo diante de maior complexidade. A análise estatística, por meio dos testes de Shapiro-Wilk e Wilcoxon, confirmou a superioridade do GRASP com p-valores extremamente baixos, reforçando sua robustez e eficácia em comparação ao BRKGA nas instâncias testadas.

4 Considerações Finais

Este trabalho trata do PAGRMRO, um problema de conectividade em grafos com arestas rotuladas que busca construir uma árvore geradora utilizando o menor número possível de rótulos, sob restrições orçamentárias no número de arestas por rótulo. Foram desenvolvidos dois métodos heurísticos baseados nas meta-heurísticas GRASP e BRKGA, além de uma formulação exata com algoritmo de *branch-and-cut*, utilizada para validação dos métodos propostos.

Tabela 1: Resultados dos experimentos para grafos com 20, 30, 40 e 50 vértices

$\lvert V \rvert$	d	50%						60%						70%					
		Exato		GRASP		BRKGA		Exato		GRASP		BRKGA		Exato		GRASP		BRKGA	
		obj	t (s)	obj	t_T (s)	obj	t_T (s)	obj	t (s)	obj	t_T (s)	obj	t_T (s)	obj	t (s)	obj	t_T (s)	obj	t_T (s)
20	0,2	**12,0**	37,446	**12,0**	0,000	**12,0**	0,005	**8,9**	0,289	**8,9**	0,000	**8,9**	0,001	**7,1**	0,015	**7,1**	0,000	**7,1**	0,004
20	0,5	**5,1**	TLE	**5,1**	0,000	**5,1**	0,005	**4,0**	0,207	**4,0**	0,000	**4,0**	0,065	**3,8**	0,110	**3,8**	0,000	**3,8**	0,001
20	0,8	**3,7**	454,732	**3,7**	0,000	**3,7**	0,001	**3,0**	0,346	**3,0**	0,000	**3,0**	0,043	**2,8**	0,205	**2,8**	0,000	**2,8**	0,004
30	0,2	**11,9**	TLE	**11,9**	0,005	**11,9**	0,051	**9,3**	28,290	**9,3**	0,006	**9,3**	0,049	**8,0**	0,107	**8,0**	0,008	**8,0**	0,279
30	0,5	**5,3**	TLE	**5,3**	0,001	**5,3**	0,073	**4,5**	TLE	**4,5**	0,000	**4,5**	0,214	**4,0**	4,741	**4,0**	0,000	**4,0**	0,073
30	0,8	**3,8**	TLE	**3,8**	0,001	**3,8**	0,013	**3,0**	1,185	**3,0**	0,000	**3,0**	0,372	**3,0**	1,257	**3,0**	0,000	**3,0**	0,030
40	0,2	*12,1	TLE	**12,0**	0,022	**12,0**	0,176	*9,8	TLE	**9,7**	0,106	9,9	0,552	**8,5**	73,084	**8,5**	0,019	8,7	0,028
40	0,5	**5,5**	TLE	**5,5**	0,001	**5,5**	0,014	**4,5**	TLE	**4,5**	0,002	**4,5**	1,581	**4,0**	8,638	**4,0**	0,007	4,1	0,769
40	0,8	**4,0**	TLE	**4,0**	0,000	**4,0**	0,003	**3,0**	81,194	**3,0**	0,000	3,3	0,875	**3,0**	2,375	**3,0**	0,000	**3,0**	1,258
50	0,2	*13,1	TLE	**12,5**	1,032	12,6	0,996	*10,3	TLE	**10,2**	0,674	10,4	0,801	**9,2**	TLE	**9,2**	0,312	9,7	0,316
50	0,5	*6,0	TLE	**5,8**	0,017	5,9	0,425	**4,9**	TLE	**4,9**	0,003	5,0	0,333	**4,3**	TLE	**4,3**	0,044	4,6	0,806
50	0,8	**4,0**	TLE	**4,0**	0,000	**4,0**	0,006	**3,1**	TLE	**3,1**	0,003	3,9	0,831	**3,0**	7,203	**3,0**	0,000	3,6	2,296

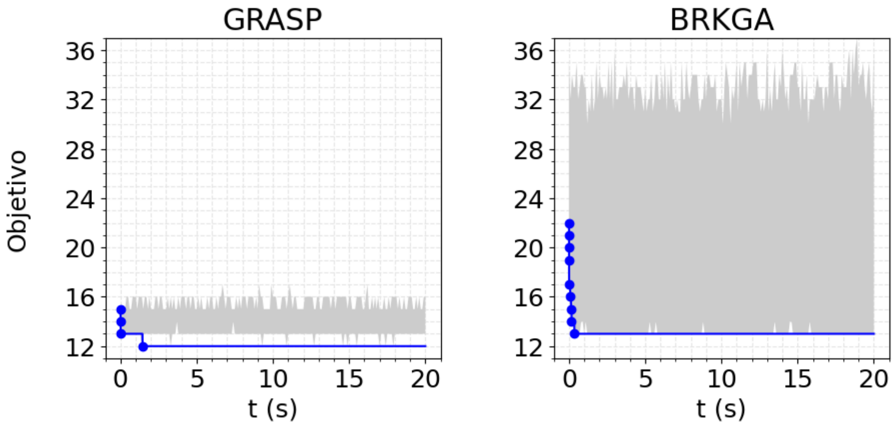

Figura 2: Gráfico de convergência dos métodos heurísticos baseados em GRASP e BRKGA para uma instância com 50 vértices, 50 rótulos, densidade de 20%, orçamento de 50% e solução ótima com 12 rótulos

Os resultados computacionais demonstraram que o método baseado em GRASP encontrou a solução ótima em todas as instâncias nas quais o algoritmo exato convergiu dentro do tempo limite. Nas demais instâncias, o GRASP também obteve, na maioria dos casos, soluções superiores às do BRKGA, com diferença estatisticamente significativa.

Como proposta para trabalhos futuros, sugere-se um estudo mais aprofundado das abordagens exatas, com o objetivo de permitir comparações mais abrangentes com os métodos heurísticos. Além disso, recomenda-se a investigação do uso de novas meta-heurísticas em combinação com os métodos desenvolvidos, visando aumentar a variabilidade do método baseado em GRASP e melhorar a convergência do método baseado no BRKGA.

Referências

CERULLI, R. et al. Metaheuristics comparison for the minimum labelling spanning tree problem. In: SPRINGER. *The Next Wave in Computing, Optimization, and Decision Technologies*. [S.l.], 2005. p. 93–106.

CHANG, R.-S.; LEU, S.-J. The minimum labeling spanning trees. *Information Processing Letters*, v. 63(5), p. 277–282, 1997.

ERICSSON, M.; RESENDE, M. G. C.; PARDALOS, P. M. A genetic algorithm for the weight setting problem in ospf routing. *Journal of combinatorial optimization*, Springer, v. 6, p. 299–333, 2002.

GONÇALVES, J. F.; ALMEIDA, J. R. A hybrid genetic algorithm for assembly line balancing. *Journal of heuristics*, Springer, v. 8, p. 629–642, 2002.

LIN, M. et al. A novel binary firefly algorithm for the minimum labeling spanning tree problem. *Computer Modeling in Engineering & Sciences*, v. 125(1), p. 197–214, 2020.

SAAD, A. et al. A grasp-genetic metaheuristic applied on multi-processor task scheduling systems. In: IEEE. *2018 13th International Conference on Computer Engineering and Systems (ICCES)*. [S.l.], 2018. p. 109–115.