



HAL
open science

Uma modelagem bi-objetivo para o problema da Árvore Geradora Mínima com Restrição de Saltos

Iago Carvalho, Amadeu Almeida Coco

► **To cite this version:**

Iago Carvalho, Amadeu Almeida Coco. Uma modelagem bi-objetivo para o problema da Árvore Geradora Mínima com Restrição de Saltos. LIV Simpósio Brasileiro de Pesquisa Operacional, Sobrapo, Nov 2022, Juiz de Fora - MG, Brazil. hal-04197013

HAL Id: hal-04197013

<https://hal.science/hal-04197013>

Submitted on 5 Sep 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/363320251>

Uma modelagem bi-objetivo para o problema da Árvore Geradora Mínima com Restrição de Saltos

Conference Paper · November 2022

CITATIONS

0

READS

20

2 authors:



Iago Augusto Carvalho
Universidade Federal de Alfenas

50 PUBLICATIONS 139 CITATIONS

SEE PROFILE



Amadeu Almeida Coco
Université du Havre

20 PUBLICATIONS 101 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Logistics operations after industrial disasters (LIS) [View project](#)



Methods for optimization problems under uncertain data (METHOD!) [View project](#)

Uma modelagem bi-objetivo para o problema da Árvore Geradora Mínima com Restrição de Saltos

Iago A. Carvalho

Universidade Federal de Alfenas

Av. Jovino Fernandes de Sales, 2600 - Santa Clara, Alfenas - MG, Brasil, 37133-840

iago.carvalho@unifal-mg.edu.br

Amadeu A. Coco

Université Le Havre Normandie

25 Rue Philippe Lebon, 76600, Le Havre, France

amadeu.almeida-coco@univ-lehavre.fr

RESUMO

Este trabalho propõe um modelo bi-objetivo para o problema da Árvore Geradora Mínima com Restrição de Saltos. Neste problema, o primeiro objetivo visa minimizar o peso da árvore geradora, enquanto o segundo objetivo minimiza o número de vértices no caminho entre um nó raiz da árvore e os demais vértices desta mesma árvore. Nós propomos um modelo bi-objetivo baseado nas restrições de eliminação de subciclos de Miller-Tucker Zemlin e o resolvemos utilizando um algoritmo de ϵ -constraints aumentado. Experimentos computacionais, realizados em instâncias clássicas da literatura para a versão mono-objetivo deste problema, demonstram os limites do método proposto.

PALAVRAS CHAVE. Otimização multi-objetivo, Árvore Geradora Mínima com Restrição de Saltos, ϵ -constraints aumentado

Otimização combinatória, Programação matemática

ABSTRACT

This work proposes a bi-objective model for the Hop-Constrained Minimum Spanning Tree problem. In this problem, the first objective is to minimize the weight of the spanning tree, while the second objective constrains the number of vertices in the path between a root vertex of the tree and the remaining vertices of this same tree. We propose a bi-objective model based on the Miller-Tucker-Zemlin subtour elimination constraints and solve it by employing an Augmented ϵ -constraints algorithm. Computational experiments, performed on classical literature instances for the mono-objective variant of this problem, exhibit the limits of the proposed method.

KEYWORDS. Multi-objective optimization, Hop-Constrained Minimum Spanning Tree, Augmented ϵ -constraint

Combinatorial optimization, mathematical programming

1. Introdução

Seja $G = (N, E)$ um grafo conectado, onde N e E são, respectivamente, o conjunto de vértices e arestas. Além disso, toda aresta $e \in E$ é associada com um peso $w_e > 0$. Uma árvore geradora é um subgrafo de G , conectado, que alcança todos os vértices em N e não possui ciclos. O problema da Árvore Geradora Mínima (AGM) consiste em computar a árvore geradora de G cujo somatório dos pesos de suas arestas é o mínimo possível. Este problema aparece de forma natural nas áreas de redes de comunicação, no desenho de circuitos e transportes, dentre outras, sendo utilizado quando deve-se encontrar uma estrutura de custo mínimo que conecta três ou mais pontos de interesse dispersos fisicamente. A AGM pode ser resolvida de forma eficiente utilizando algoritmos polinomiais, tais como o Algoritmo de Kruskal ou o Algoritmo de Prim [Cormen et al., 2009].

Problemas de árvores geradoras mínimas restritos são variantes de AGM que possuem, ao menos, uma característica adicional. Estes problemas são usualmente NP-Difíceis, tais como o Problema da Árvore Geradora Mínima com Restrição de Ângulo [Aschner e Katz, 2017], o Problema da Árvore Geradora Mínima com Restrição em *Delay* [Salama et al., 1997], o problema da Árvore Geradora Mínima com Restrição em Grau [de Almeida et al., 2012], o problema da Árvore Geradora Mínima com Restrição de Saltos [Gouveia, 1995] e o problema da Árvore Geradora Mínima com Restrição de Folhas [Deo e Micikevicius, 1999]. Devido a sua estrutura complexa, estes problemas são resolvidos, de forma exata, utilizando métodos sofisticados baseados em programação por restrições [Noronha et al., 2010], métodos de plano de corte [Martinez e Da Cunha, 2014] e geração de colunas [Gouveia et al., 2008], dentre outros.

Este trabalho estuda o problema da Árvore Geradora Mínima com Restrição de Saltos (AGM-RS). Este problema foi proposto e demonstrado ser NP-Completo por Gouveia [Gouveia, 1995]. Existem diversas formulações por Programação Linear Inteira para este problema. Estas são baseadas nas restrições de eliminação de subciclo de Miller-Tucker-Zemlin [Gouveia, 1995; Akgün e Tansel, 2011; Akgün, 2011], restrições multi-fluxo [Gouveia, 1996] e uma formulação *cut-set* derivada do politopo do problema da Árvore de Steiner [Gouveia et al., 2011b]. Além disso, heurísticas [Fernandes et al., 2007; Gouveia et al., 2011a], um algoritmo de pré-processamento [Gouveia, 1995] e limites duais [Gouveia, 1996, 1998; Gouveia e Requejo, 2001] também foram propostos para o AGM-RS.

Apesar do volume de trabalhos na literatura relacionados a este problema, até onde sabemos, sua variante bi-objetivo nunca foi estudada. Desta forma, o objetivo deste trabalho é apresentar uma modelagem bi-objetivo para o AGM-RS onde o primeiro objetivo visa minimizar o peso da árvore geradora, enquanto o segundo objetivo restringe o número de vértices no caminho entre um nó raiz da árvore e seus demais vértices. Esta modelagem permite apresentar um maior conjunto de soluções a um tomador de decisão quando este utilizar o AGM-RS para resolver algum problema prático relevante a sua realidade. Desta forma, sua decisão pode levar em consideração o compromisso entre ambos os objetivos do problema.

Outros trabalhos na literatura já trataram problemas de árvore geradora mínima com dois ou mais objetivos. O mais simples deles é o problema da Árvore Geradora Mínima Bi-Objetivo [Corley, 1985] onde cada aresta da árvore é associada a duas classes de pesos. Além deste, também podemos citar o problema bi-objetivo da Árvore Geradora Mínima com Restrição de Diâmetro [Ho et al., 1991], o problema bi-objetivo da Árvore Geradora Mínima com Restrição de Grau [Knowles e Corne, 2001] e o problema bi-objetivo da Árvore Geradora Mínima Restrita em Erro de Calibração [Carvalho e Ribeiro, 2019].

Este trabalho modela o AGM-RS bi-objetivo utilizando uma formulação por Programação

Linear Inteira baseada nas restrições de eliminação de subciclos de Miller-Tucker-Zemlin. Este modelo, então, é resolvido utilizando um método ϵ -constraints aumentado [Mavrotas, 2009; Carvalho e Ribeiro, 2020].

O restante deste trabalho é organizado como se segue. Inicialmente, a Seção 2 apresenta os principais conceitos relacionados a otimização multi-objetivo. Então, a Seção 3 propõe o modelo bi-objetivo para o AMG-RS baseado em uma formulação com as restrições de eliminação de subciclo de Miller-Tucker-Zemlin para a versão mono-objetivo deste problema [Gouveia, 1995]. Logo após, a Seção 4 propõe um algoritmo de ϵ -constraints aumentado para a resolução do modelo desenvolvido na seção anterior. Este algoritmo é avaliado computacionalmente na Seção 5 utilizando um conjunto de instâncias da literatura para a versão mono-objetivo do AGM-RS. Por fim, a última seção apresenta as conclusões e trabalhos futuros.

2. Conceitos de otimização multi-objetivo

Um problema de otimização multi-objetivo pode ser formulado, de forma geral, pelo conjunto de funções objetivos descritas em (1) e pelas restrições em (2). Sem perda de generalidade, assume-se que todas as funções objetivos descritas são de minimização. Neste problema, a Equação (1) representa um conjunto de m funções objetivos e as Desigualdades (2) representam as restrições do problema, as quais geram um polítopo \mathcal{P} . Além disso, Φ representa o conjunto de soluções viáveis em \mathcal{P} e $\phi \in \Phi$ denota uma solução viável.

$$\min_{\phi \in \Phi} [f^1(\phi), f^2(\phi), \dots, f^m(\phi)]^T \quad (1)$$

s.t.

$$q_j(\phi) \leq 0, \quad j = 1, \dots, J \quad (2)$$

Definição 1 (Conceito de dominância): Uma solução $\phi_1 \in \Phi$ domina outra solução $\phi_2 \in \Phi$ se e somente se $f^i(\phi_1) \leq f^i(\phi_2)$ para todas as funções objetivo $\{f^1, f^2, \dots, f^m\}$ e $f^i(\phi_1) < f^i(\phi_2)$ para, ao menos, uma das funções objetivo [Deb, 2014].

Definição 2 (Não-dominância): Uma solução $\phi' \in \Phi$ é dita ser não-dominada se e somente se não existe uma solução $\phi \in \Phi$ tal que ϕ domina ϕ' [Deb, 2014].

Seja $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ uma função que mapeia as soluções em Φ no espaço m -dimensional de imagens de soluções Ψ tal que $\Psi = \{\psi \in \mathbb{R}^m : \psi = g(\phi), \phi \in \Phi\}$. A i -ésima posição do vetor ψ corresponde ao valor da função objetivo f^i para a solução ϕ , i. e., $\psi_i = f^i(\phi)$.

Definição 3 (Otimidade de pareto): Um ponto $\psi^* \in \Psi$ é dito ser Pareto-ótimo se e somente se não existem pontos $\psi \in \Psi$ tal que $\psi_i \leq \psi_i^*$, para $i \in \{1, \dots, m\}$, e que $\psi_i < \psi_i^*$ para ao menos um $i \in \{1, \dots, m\}$ [Deb, 2014].

Definição 4 (Fronteira de pareto): Uma fronteira de pareto é o conjunto de todos os pontos pareto-ótimos [Deb, 2014].

Uma solução ótima para um problema multi-objetivo é denotada como um vetor \mathcal{F} contendo todos os pontos pareto-ótimos, i. e., a fronteira de pareto do problema.

3. Um modelo bi-objectivo para o AGM-RS

Seja $G = (N, E)$ um grafo conectado como definido na Seção 1. Além disso, seja $r \in N$ um nó raiz da árvore e $p > 1$ o número máximo de saltos (arestas) permitido pelo problema. O AGM-RS consiste em encontrar uma árvore geradora de G de peso mínimo e que o caminho entre r e qualquer outro vértice do grafo tenha, no máximo, p saltos.

Uma formulação por Programação Linear Inteira para o AGM-RS é definido em um digrafo simétrico $G' = (N, A)$. O conjunto de arcos A deste digrafo é obtido bi-direcionando as arestas em E do grafo original. Além disso, os pesos dos arcos também são simétricos, *i. e.*, $w_{ij} = w_{ji}$ para todo $(i, j) \in A$. Desta forma, uma solução para o AGM-RS é uma arborescência geradora de G' .

Nossa formulação utiliza variáveis de decisão binárias $\{x_{ij} \in \mathbb{B} : (i, j) \in A\}$ que definem a topologia da arborescência, tal que $x_{ij} = 1$ se o arco x_{ij} pertence a arborescência e $x_{ij} = 0$ caso contrário. Além disso, variáveis não-negativas $\{u_i \in \mathbb{N} : i \in N\}$ são usadas como rótulos para os vértices.

Com base nas definições acima, a função objetivo da formulação do AGM-RS é definida como

$$y_1^* = \min \left\{ \sum_{(i,j) \in A} w_{ij} x_{ij} : (x, u) \in \mathcal{P}_1 \cap \left(\mathbb{B}^{|A|} \times \mathbb{N}^{|N|} \right) \right\}, \quad (3)$$

onde o conjunto \mathcal{P}_1 define a região poliedral construída pelas restrições em (4)–(8).

$$\sum_{(i,j) \in A} x_{ij} = 1, \quad \forall j \in N \setminus \{r\} \quad (4)$$

$$u_i - u_j + |N| x_{ij} \leq |N| - 1, \quad \forall (i, j) \in A, j \in N \setminus \{r\} \quad (5)$$

$$u_r = 0, \quad (6)$$

$$u_i \leq p, \quad \forall i \in N \setminus \{r\} \quad (7)$$

$$u_i \geq 1, \quad \forall i \in N \setminus \{r\} \quad (8)$$

Neste modelo, a função objetivo definida pela Equação (3) minimiza o peso total da arborescência. As desigualdades em (4)–(8) garantem que a solução seja acíclica, sendo que as restrições em (4) definem que o grau de entrada de todos os vértices (com exceção do vértice raiz r) deve ser exatamente um. As desigualdades em (5)–(8) são as clássicas restrições de eliminação de subciclo de Miller-Tucker-Zemlin e definem rótulos para os vértices do grafo. Esta definição de rótulos é realizada de tal forma que todo arco incluído na arborescência tem origem de um vértice de menor rótulo e destino em um vértice de maior rótulo. Portanto, elas garantem que o caminho entre r e os outros vértices do grafo siga através de vértices com valores de rótulos estritamente monotônicos, assim inibindo a formação de subciclos. Além disso, estas desigualdades, em conjunto com as restrições em (4), garantem a conectividade da solução. A equação (6) define o rótulo zero para o nó raiz enquanto as desigualdades em (7) e (8) definem, respectivamente, os limites superior e inferior para os valores de rótulos dos outros vértices do grafo.

É possível desenvolver uma formulação bi-objetivo para o AGM-RS ao construir uma segunda função objetivo que minimiza o máximo número de saltos. Esta formulação é não-linear e é representada como

$$y_2^* = \min \max_{i \in N} \left\{ u_i : (x, u) \in \mathcal{P}_2 \cap \left(\mathbb{B}^{|A|} \times \mathbb{N}^{|N|} \right) \right\}, \quad (9)$$

onde \mathcal{P}_2 representa a região poliedral dada pelas restrições em (4)–(6) e (8).

Pode-se linearizar a função objetivo acima ao tratarmos p como uma variável $\{p \in \mathbb{N}\}$. Desta forma, podemos reescreve-la como

$$y_2^* = \min \left\{ p : (x, u, p) \in \mathcal{P}_1 \cap \left(\mathbb{B}^{|A|} \times \mathbb{N}^{|N|+1} \right) \right\}. \quad (10)$$

O problema bi-objetivo da Árvore Geradora Mínima com Restrição de Saltos (BAGM-RS) consiste em otimizar ambas as funções objetivos (3) e (10). Pode-se observar que estes objetivos são conflitantes: quanto menor o valor de p , menor é o espaço de busca para definir caminhos entre r e os outros vértices do digrafo. Por consequência, menor é a liberdade para escolhermos as arestas que minimizam o custo da arborescência de G' .

4. O método de ϵ -constraints aumentado para o BAGM-RS

O método do ϵ -constraints (ϵ -CM) é baseado na estratégia de enumeração de ϵ -constraint [Haimes et al., 1971]. Esta abordagem otimiza um dos objetivos do problema enquanto restringe o valor dos objetivos adicionais, assim transformando as demais funções objetivo em restrições. O ϵ -CM é capaz de computar a fronteira de pareto mesmo para problemas de otimização não-convexos [Marler e Arora, 2004; Chankong e Haimes, 2008]. Este algoritmo foi aplicado com sucesso na resolução de diversos problemas de otimização multi-objetivo, como problemas de transportes [Coco et al., 2020], problemas de mochila [Rong e Figueira, 2014], problemas de agendamento [Binder et al., 2017], e problemas bi-objetivo em árvores geradoras mínimas [Carvalho e Ribeiro, 2020]. O trabalho de Marler e Arora [2004] apresenta uma antiga revisão sobre as aplicações deste algoritmo na literatura.

Dado o problema genérico de otimização bi-objetivo definido na Seção 2, o ϵ -CM iterativamente resolve o problema de otimização (11).

$$\begin{aligned} \min \quad & f^1(\phi) \\ \text{s.t.} \quad & f^2(\phi) \leq \epsilon_1 \\ & f^3(\phi) \leq \epsilon_2 \\ & \vdots \\ & f^m(\phi) \leq \epsilon_{m-1} \\ & \phi \in \Phi \end{aligned} \quad (11)$$

Teorema 1 *Uma solução ótima para (11) é também uma solução não-dominada para o problema (1)–(2) para um vetor específico $\epsilon = \{\epsilon_1, \epsilon_2, \dots, \epsilon_{m-1}\}$ [Wendell e Lee, 1977; Corley, 1980].*

Colorário 2 *Um valor ótimo para (11) é um ponto pareto-ótimo para o problema (1)–(2) para um vetor específico $\epsilon = \{\epsilon_1, \epsilon_2, \dots, \epsilon_{m-1}\}$ [Wendell e Lee, 1977; Corley, 1980].*

Colorário 3 *O ϵ -CM é capaz de gerar a fronteira de pareto completa de um problema de otimização multi-objetivo (Chankong e Haimes [2008], página 276).*

É possível computar a fronteira de pareto de um problema de otimização multi-objetivo ao alterarmos os valores do lado direito da equação do problema (11), *i. e.*, ao variar os valores do vetor ϵ .

Mavrotas [2009] propôs uma estratégia para computar somente os pontos pareto-ótimo utilizando o ϵ - CM , desta forma, acelerando a convergência do método ao evitar iterações redundantes. Este novo método, denotado de ϵ -constraints aumentado (*aum* ϵ - CM), baseia-se em dois aspectos: (i) transformar a restrição associada a cada função objetivo em uma igualdade incorporando variáveis de folga; e (ii) usar estas variáveis de folga como um segundo termo na função objetivo do problema com um peso muito pequeno. Desta forma, é possível reescrever (11) como o problema (12), onde eps é uma constante pequena no intervalo $[10^{-6}, 10^{-3}]$.

$$\begin{aligned}
 \min \quad & f^1(\phi) - eps \cdot (s_1 + s_2 + \dots + s_{m-1}) \\
 \text{s.t.} \quad & \\
 & f^2(\phi) + s_1 = \epsilon_1 \\
 & f^3(\phi) + s_2 = \epsilon_2 \\
 & \vdots \\
 & f^m(\phi) + s_{m-1} = \epsilon_{m-1} \\
 & \phi \in \Phi \\
 & s_i \in \mathbb{R}_{\geq 0} \quad \forall i \in \{1, 2, \dots, m-1\}
 \end{aligned} \tag{12}$$

4.1. Utilização do *aum* ϵ - CM para a resolução do BAGM-RS

O *aum* ϵ - CM proposto para o BAGM-RS restringe o peso da árvore enquanto otimiza diretamente o número de saltos p . Pode-se reescrever a função objetivo descrita pela equação (10), que minimiza o máximo número de saltos, como

$$y_2^* = \min \left\{ p - eps \cdot s : (x, u, p, s, \epsilon) \in \mathcal{P}_3 \cap \left(\mathbb{B}^{|A|} \times \mathbb{N}^{|N|+1} \times \mathbb{R}^2 \right) \right\}, \tag{13}$$

onde \mathcal{P}_3 representa a região poliedral dado por $\mathcal{P}_2 \cap (14)$.

$$\sum_{(i,j) \in A} w_{ij} x_{ij} + s = \epsilon \tag{14}$$

O algoritmo *aum* ϵ - CM para o BAGM-RS consiste em resolver, de forma iterativa, o problema de otimização 13 variando o valor de ϵ . Sejam Z_1 e Z_2 as funções objetivo denotadas pelas equações 3 e 13 respectivamente. Adicionalmente, seja Φ o conjunto de soluções viáveis contidas em \mathcal{P}_3 e seja $\phi \in \Phi$ uma solução viável. É possível representar $Z_1(\phi)$ e $Z_2(\phi)$ como, respectivamente, os valores das funções objetivo Z_1 e Z_2 para uma solução ϕ . O algoritmo 1 apresenta o pseudo-código para o método *aum* ϵ - CM proposto. Ele recebe duas entradas: (i) as funções objetivo do problema; e (ii) o polítopo correspondente. O algoritmo retorna a fronteira de pareto \mathcal{F} de uma dada instância. Primeiro, ele inicializa a fronteira de pareto como um conjunto vazio (line 1). Então, ele restringe $s = 0$ e resolve o problema resultante (linhas 2–3) para obter um valor inicial para ϵ (linha 4). Em sequência, remove-se a restrição do valor de s (linha 5). O laço das linhas 6–9 é executado até que uma solução inviável seja encontrada. Cada iteração deste laço resolve o problema utilizando o valor corrente de ϵ (linha 7), persiste o ponto de pareto resultante (linha 8) e atualiza o valor de ϵ (linha 9). Finalmente, a última linha retorna a fronteira de pareto \mathcal{F} .

5. Experimentos computacionais

Os experimentos computacionais foram realizados em um único núcleo de um processador Intel Xeon CPU E5-2630 v4 com clock de 2.2 GHz e 64 GB de RAM, rodando o sistema operacional Linux 16.04. O algoritmo *aum* ϵ - CM foi implementado utilizando C++14 e compilado com o GNU g++ 8.2. O modelo de programação linear inteira utilizado pelo *aum* ϵ - CM

Algoritmo 1: O algoritmo *aum* ϵ -*CM* para o BAGM-RS

```

entrada:  $Z_1, Z_2$  // the problem's objective functions
entrada:  $\mathcal{P}_3$  // the problem's polytope
saída :  $\mathcal{F}$  // the Pareto-front
1  $\mathcal{F} \leftarrow \emptyset$ 
2  $s \in \{0\}$ 
3  $\phi \leftarrow \arg \min Z_1$ 
4  $\epsilon \leftarrow Z_2(\phi) - 1$ 
5  $s \in \mathbb{R}_{\geq 0}$ 
6 while viável ( $Z_1$ ) do
7    $\phi \leftarrow \arg \min Z_1$ 
8    $\mathcal{F} \leftarrow \mathcal{F} \cup (Z_1(\phi), Z_2(\phi))$ 
9    $\epsilon \leftarrow Z_2(\phi) - 1$ 
10 return  $\mathcal{F}$ 

```

foi resolvido utilizando o IBM/ILOG CPLEX¹ versão 12.10 utilizando os parâmetros padrão. Além disso, nós limitamos o tempo de execução do nosso algoritmo para 36 000 segundos (10 horas).

Este estudo avaliou dois conjuntos de instâncias de teste, respectivamente nomeadas de TC e TE. Ambos os conjuntos de instâncias foram utilizadas previamente em experimentos computacionais para o AGM-RS [Gouveia et al., 2008]. Em ambos os conjuntos de instâncias, os vértices foram posicionados em um plano cartesiano de tamanho 40×40 usando uma distribuição discreta uniforme $\mathcal{U}(1, 40)$ para gerar suas coordenadas. Foi construído um digrafo completo, sendo que o peso dos arcos foram computados utilizando a distância Euclidiana entre os vértices. As instâncias diferem entre si unicamente pela localização do vértice raiz: no conjunto TC, este está posicionado no centro do plano, *i. e.*, nas coordenadas (20, 20). Já nas instâncias do conjunto TE o vértice raiz está localizado no canto do plano, *i. e.*, nas coordenadas (0, 0).

Os experimentos computacionais avaliaram instâncias com $|N| \in \{10, 11, \dots, 25\}$. Para cada conjunto (TC e TE) e para cada valor de $|N|$, um total de 10 diferentes instâncias foram geradas utilizando diferentes sementes para o gerador de números pseudo-aleatórios. Desta forma, avaliamos um total de 320 instâncias, sendo 160 do conjunto TC e 160 do conjunto TE.

6. Resultados experimentais

A Tabela 1 exhibe os resultados dos experimentos computacionais realizados. A primeira coluna mostra o conjunto de instâncias avaliado, enquanto a segunda coluna denota o número de nós das instâncias em cada subconjunto, *i. e.*, o valor de $|N|$. A terceira, quarta e quinta colunas mostram os resultados do *aum* ϵ -*CM*: a terceira coluna exhibe o número de instâncias com soluções ótimas dentro do tempo limite de 36000 segundos, enquanto a quarta coluna mostra o valor médio e o desvio padrão dos tamanhos das fronteiras de pareto \mathcal{F} encontradas e a quinta coluna exhibe o tempo médio e o desvio padrão do tempo de execução do *aum* ϵ -*CM* para cada subconjunto de instâncias. Vale a pena lembrar que cada linha sumariza os resultados para 10 instâncias geradas utilizando diferentes sementes para o gerador de números pseudo-aleatórios.

Sobre o conjunto TC, pode-se observar que todas as instâncias com $|N| \leq 19$ foram resolvidas na otimalidade. Entretanto, somente uma instância com $|N| \leq 24$ foi capaz de ser

¹<https://www.ibm.com/products/ilog-cplex-optimization-studio>

Tabela 1: Resultados do *aum* ϵ -*CM* para o BAGM-RS

Conjunto	$ N $	Ótimo	$ \mathcal{F} $	Tempo (s)
TC	10	10	5.5 ± 0.7	0.2 ± 0.1
	11	10	5.8 ± 1.0	0.4 ± 0.3
	12	10	6.9 ± 0.7	1.3 ± 1.6
	13	10	7.0 ± 1.3	3.1 ± 4.0
	14	10	8.5 ± 2.0	8.5 ± 6.7
	15	10	8.7 ± 1.2	17.3 ± 10.9
	16	10	9.2 ± 1.0	32.7 ± 44.2
	17	10	10.0 ± 1.0	86.5 ± 76.1
	18	10	10.2 ± 1.0	522.5 ± 594.8
	19	10	11.0 ± 1.0	436.7 ± 703.4
	20	9	11.3 ± 1.3	6643.8 ± 12423.7
	21	8	11.2 ± 1.7	11526.5 ± 14035.5
	22	5	7.7 ± 4.1	24284.3 ± 14775.5
	23	4	7.6 ± 4.1	30198.6 ± 9897.5
	24	0	3.7 ± 0.8	36000.0
25	1	4.9 ± 3.2	32892.2 ± 9827.8	
TE	10	10	9.0 ± 0.7	17.3 ± 46.8
	11	10	10.0	44.3 ± 121.3
	12	10	11.0 ± 0.7	25.1 ± 41.1
	13	10	12.1 ± 0.3	105.1 ± 203.8
	14	9	12.7 ± 2.6	3682.4 ± 11355.5
	15	10	14.1 ± 0.3	646.9 ± 588.7
	16	9	14.7 ± 1.0	5674.1 ± 10778.6
	17	7	14.8 ± 1.5	14505.7 ± 16637.6
	18	5	13.2 ± 0.8	25914.2 ± 14423.6
	19	3	11.0 ± 2.3	30571.2 ± 11202.4
	20	2	10.9 ± 2.0	30147.4 ± 12339.2
	21	1	10.4 ± 2.7	33961.6 ± 6445.9
	22	0	7.7 ± 4.1	36000.0
	23	0	7.6 ± 4.1	36000.0
	24	0	3.7 ± 0.8	36000.0
25	0	4.9 ± 3.2	36000.0	

resolvida pelo método proposto. No total, o *aum* ϵ -*CM* resolveu 127 das 160 instâncias propostas no conjunto TC.

Os experimentos computacionais também demonstraram que o conjunto de instâncias TE é mais difícil de ser resolvido do que o conjunto TC. Pode-se observar na Tabela 1 que já existe uma instância com $|N| = 14$ sem solução ótima, sendo que foram encontradas as soluções ótimas de somente 86 das 160 instâncias avaliadas. Além disso, o tempo médio de execução para as instâncias do conjunto TE foi estritamente superior aos tempos de execução das instâncias do conjunto TC quando comparados os mesmos valores de $|N|$. Além disso, no geral, o tamanho $|\mathcal{F}|$ das fronteiras de Pareto encontradas nas instâncias do conjunto TE também foram maiores, na média, do que

aqueles para o conjunto TC.

7. Conclusões

Este trabalho estudou o problema da Árvore Geradora Mínima com Restrição de Saltos (AGM-RS) como um problema bi-objetivo. O primeiro objetivo minimiza o custo total da árvore enquanto o segundo minimiza o número de saltos permitidos de um vértice raiz da árvore r até seus demais vértices. O problema foi modelado utilizando as clássicas restrições de eliminação de subciclo de Miller-Tucker-Zemlin e o modelo resultante foi utilizado dentro de um algoritmo de ϵ -constraints aumentado (*aum ϵ -CM*). Experimentos computacionais avaliaram a performance do algoritmo proposto em dois conjuntos de instâncias clássicos da literatura.

Os resultados iniciais aqui expostos servem para motivar o estudo da variante bi-objetivo do AGM-RS. Foi demonstrado que o algoritmo proposto é capaz de resolver instâncias pequenas, com 20 ou menos vértices, em um intervalo de 10 horas. Desta forma, espera-se que trabalhos futuros neste tema melhorem a formulação proposta, utilizando métodos mais elaborados baseado em restrições multi-fluxo [Gouveia, 1996] e na formulação *cut-set* derivado do politopo do problema da Árvore de Steiner [Gouveia et al., 2011b].

Além disso, trabalhos futuros sobre este problema também podem avaliar uma diferente implementação do algoritmo *aum ϵ -CM*, de acordo com o trabalho de Carvalho e Ribeiro [2020]. Neste, foi demonstrado que é possível obter melhores resultados ao se otimizar o custo da árvore enquanto mantêm-se o objetivo adicional do problema.

Referências

- Akgün, I. (2011). New formulations for the hop-constrained minimum spanning tree problem via sherali and driscoll's tightened miller-tucker-zemlin constraints. *Computers & Operations Research*, 38(1):277–286.
- Akgün, I. e Tansel, B. C. (2011). New formulations of the hop-constrained minimum spanning tree problem via miller-tucker-zemlin constraints. *European Journal of Operational Research*, 212(2):263–276.
- Aschner, R. e Katz, M. J. (2017). Bounded-angle spanning tree: modeling networks with angular constraints. *Algorithmica*, 77(2):349–373.
- Binder, S., Maknoon, Y., e Bierlaire, M. (2017). The multi-objective railway timetable rescheduling problem. *Transportation Research Part C: Emerging Technologies*, 78:78–94.
- Carvalho, I. A. e Ribeiro, M. A. (2019). A node-depth phylogenetic-based artificial immune system for multi-objective network design problems. *Swarm and Evolutionary Computation*, 50:100491.
- Carvalho, I. A. e Ribeiro, M. A. (2020). An exact approach for the minimum-cost bounded-error calibration tree problem. *Annals of Operations Research*, 287(1):109–126.
- Chankong, V. e Haimes, Y. Y. (2008). *Methods for Generating Noninferior Solutions*, chapter 6, p. 221–290. Courier Dover Publications. ISBN 0486462897.
- Coco, A. A., Duhamel, C., e Santos, A. C. (2020). Modeling and solving the multi-period disruptions scheduling problem on urban networks. *Annals of Operations Research*, 285(1-2):427–443.
- Corley, H. (1980). A new scalar equivalence for pareto optimization. *IEEE Transactions on Automatic Control*, 25(4):829–830.

- Corley, H. (1985). Efficient spanning trees. *Journal of Optimization Theory and Applications*, 45(3):481–485.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., e Stein, C. (2009). Minimum spanning trees. In *Introduction to Algorithms*, p. 624–632. MIT Press Cambridge.
- de Almeida, A. M., Martins, P., e de Souza, M. C. (2012). Min-degree constrained minimum spanning tree problem: complexity, properties, and formulations. *International Transactions in Operational Research*, 19(3):323–352.
- Deb, K. (2014). Multi-objective optimization. In *Search Methodologies*, p. 403–449. Springer.
- Deo, N. e Micikevicius, P. (1999). A heuristic for a leaf constrained minimum spanning tree problem. *Congressus Numerantium*, p. 61–72.
- Fernandes, M., Gouveia, L., e Voß, S. (2007). Determining hop-constrained spanning trees with repetitive heuristics. *Journal of Telecommunications and Information Technology*, p. 16–22.
- Gouveia, L. (1995). Using the miller-tucker-zemlin constraints to formulate a minimal spanning tree problem with hop constraints. *Computers & Operations Research*, 22(9):959–970.
- Gouveia, L. (1996). Multicommodity flow models for spanning trees with hop constraints. *European Journal of Operational Research*, 95(1):178–190.
- Gouveia, L. (1998). Using variable redefinition for computing lower bounds for minimum spanning and steiner trees with hop constraints. *INFORMS Journal on Computing*, 10(2):180–188.
- Gouveia, L., Paias, A., e Sharma, D. (2008). Modeling and solving the rooted distance-constrained minimum spanning tree problem. *Computers & Operations Research*, 35(2):600–613.
- Gouveia, L., Paias, A., e Sharma, D. (2011a). Restricted dynamic programming based neighborhoods for the hop-constrained minimum spanning tree problem. *Journal of Heuristics*, 17(1):23–37.
- Gouveia, L., Simonetti, L., e Uchoa, E. (2011b). Modeling hop-constrained and diameter-constrained minimum spanning tree problems as steiner tree problems over layered graphs. *Mathematical Programming*, 128(1-2):123–148.
- Gouveia, L. e Requejo, C. (2001). A new lagrangean relaxation approach for the hop-constrained minimum spanning tree problem. *European Journal of Operational Research*, 132(3):539–552.
- Haimes, Y., Ladson, L., e Wismer, D. (1971). On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, 1:296–297.
- Ho, J.-M., Lee, D., Chang, C.-H., e Wong, C. (1991). Minimum diameter spanning trees and related problems. *SIAM Journal on Computing*, 20(5):987–997.
- Knowles, J. D. e Corne, D. W. (2001). Benchmark problem generators and results for the multiobjective degree-constrained minimum spanning tree problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, p. 424–431.

- Marler, R. e Arora, J. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395.
- Martinez, L. C. e Da Cunha, A. S. (2014). The min-degree constrained minimum spanning tree problem: Formulations and branch-and-cut algorithm. *Discrete Applied Mathematics*, 164:210–224.
- Mavrotas, G. (2009). Effective implementation of the ϵ -constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2):455–465.
- Noronha, T. F., Ribeiro, C. C., e Santos, A. C. (2010). Solving diameter-constrained minimum spanning tree problems by constraint programming. *International Transactions in Operational Research*, 17(5):653–665.
- Rong, A. e Figueira, J. R. (2014). Dynamic programming algorithms for the bi-objective integer knapsack problem. *European Journal of Operational Research*, 236(1):85–99.
- Salama, H. F., Reeves, D. S., e Viniotis, Y. (1997). The delay-constrained minimum spanning tree problem. In *Proceedings Second IEEE Symposium on Computer and Communications*, p. 699–703. IEEE.
- Wendell, R. E. e Lee, D. N. (1977). Efficiency in multiple objective optimization problems. *Mathematical Programming*, 12(1):406–414.