

UM PROCEDIMENTO PARA GERAR FRONTEIRAS DE PARETO ÓTIMAS PARA O PROBLEMA BIOBJETIVO DA ÁRVORE GERADORA DE CUSTO E DIÂMETRO MÍNIMOS

Ernando Gomes de Sousa¹, Andréa Cynthia Santos², Dario José Aloise¹

¹ Universidade do Estado do Rio Grande do Norte, UERN
Rua Almino Afonso, 478, CEP 59.610-210, Mossoró, RN, Brasil.

² ICD-LOSI, Université de Technologie de Troyes
12, rue Marie Curie, CS 42060, 10004, Troyes CEDEX, France.
{ernandogomess@gmail.com, andrea.duhamel@utt.fr, darioaloise@uern.br}

RESUMO

Neste trabalho, apresentamos um procedimento para calcular fronteiras de Pareto ótimas para o problema biobjetivo da Árvore Geradora de Custo e Diâmetro Mínimos (bi-AGCDM). O bi-AGCDM é definido em um grafo $G = (V, E)$ e objetiva encontrar árvores geradoras de custo e diâmetro mínimos. Problemas de otimização no nível estratégico para instalação de infraestruturas de trens bala, assim como problemas de otimização em redes de computadores e transportes nos níveis tático e operacional podem ser modelados como bi-AGCDM. O procedimento proposto usa componentes inspirados nas estratégias exatas multiobjetivos ϵ -constraint e *Parallel Partitioning Method*. Em nosso conhecimento, não existem procedimentos exatos na literatura “dedicados” a encontrar soluções Pareto ótimas para o bi-AGCDM.

PALAVRAS CHAVE. Árvores geradoras, biobjetivo, fronteira de Pareto ótima.

Áreas Principais: Algoritmos exatos, otimização multiobjetivo, Projeto de redes.

ABSTRACT

In this work, we propose a procedure to compute Pareto-optimal fronts for the bi-objective Minimum Diameter-Cost Spanning Tree problem (bi-MDCST). The bi-MDCST is defined in a given graph $G = (V, E)$ and aims at finding spanning trees with minimum total cost and minimum diameter. Problems in the strategic level such as the high-speed trains infrastructure, as well as network design and transportation optimization problems at tactical and operational levels can be modeled as the bi-MDCST. The proposed procedure makes use of components from the multiobjective exact methods ϵ -constraint and *Parallel Partitioning Method*. To the best of our knowledge, there are not works dedicated to provide Pareto-optimal fronts for the bi-MDCST.

KEY WORDS. Spanning trees, Biobjective, Optimal Pareto front.

Main areas: Exact algorithm, Multiobjective optimization, Network design.

1 Introdução

O problema biobjetivo da Árvore Geradora de Custo e Diâmetro Mínimos (bi-AGCDM) é definido em um grafo $G = (V, E)$, onde V é o conjunto de vértices e E é o conjunto de arestas, com custos $c_{ij} \geq 0$ associados às arestas $[i, j] \in E$. Por definição, uma Árvore Geradora \mathcal{T} de G é um subgrafo conexo de G , sem ciclos e com exatamente $|V|-1$ arestas. Uma Árvore Geradora de custo Mínimo (AGM) é aquela cuja soma total dos custos associados às arestas $[i, j] \in \mathcal{T}$ é mínima. O diâmetro de uma árvore é igual a quantidade de arestas em seu maior caminho. Assim, uma árvore geradora de diâmetro mínimo é aquela que possui o menor diâmetro entre todas as possíveis árvores geradoras \mathcal{T} de G . Existem algoritmos que calculam AGM (Kruskal, Prim) e o diâmetro de árvores geradoras \mathcal{T} em tempo computacional polinomial, Cormen *et al.* (1990). Entretanto, o bi-AGCDM é NP-difícil e não se conhece algoritmos que o resolvam em tempo polinomial, independente da instância, Ho *et al.* (1991).

Problemas de otimização no nível estratégico para instalação de infraestruturas de trens bala, assim como problemas de otimização em redes de computadores e transportes nos níveis tático e operacional podem ser modelados como bi-AGCDM. Um problema relacionado com o bi-AGCDM e bastante investigado na literatura é o problema da Árvore Geradora de custo Mínimo com Restrição de Diâmetro (AGMRD). O objetivo do AGMRD é encontrar uma árvore geradora de custo mínimo, de modo que todos os caminhos da árvore tenham no máximo D arestas, onde $D \geq 2$. O AGMRD é NP-difícil quando $3 < D < |V| - 1$, Garey e Johnson (1979). Uma vasta literatura para o AGMRD é disponível: algoritmos exatos são encontrados em T. F. Noronha (2010), Gouveia *et al.* (2011); heurísticas e metaheurísticas foram propostas por Lucena *et al.* (2010), Gruber e Raidl (2005), Requejo e Santos (2009); formulações matemáticas e desigualdades válidas foram desenvolvidas em Achuthan *et al.* (1994), Gouveia e Magnanti (2003), Santos *et al.* (2004). O bi-AGCDM difere do AGMRD porque busca otimizar o custo e o diâmetro simultaneamente. Se um diâmetro específico é fixado no espaço de busca do bi-AGCDM, o problema se reduz ao AGMRD.

Existem diversos trabalhos na literatura dedicados ao estudo de problemas biobjetivos de árvores geradoras de custo mínimo considerando duas funções de custo, tais como o algoritmo genético proposto por Zhou e Gen (1999), o *Greedy Randomized Adaptive Search Procedure* (GRASP) apresentado em Arroyo *et al.* (2008), o algoritmo de enumeração sugerido por Steiner e Radzik (2008) e o algoritmo *Branch-and-Bound* descrito por Sourd e Spanjaard (2008). Nesse caso, em termos de modelagem matemática, o problema a ser resolvido corresponde ao cálculo de uma árvore geradora com duas funções objetivos de custo. A minimização do diâmetro no bi-AGCDM está diretamente ligada a estrutura dos caminhos entre os nós da árvore geradora. Consequentemente, os modelos matemáticos são adaptados para poder contabilizar a quantidade de arestas nos caminhos da árvore. Alguns poucos trabalhos na literatura foram desenvolvidos para resolver o bi-AGCDM. Um estudo teórico sobre algoritmos aproximativos para os problemas de bi-AGCDM e AGM com minimização de grau (quantidade de arestas incidentes a cada nó de G) é apresentado em Marathe *et al.* (1998), sem resultados computacionais. Algoritmos heurísticos baseados em algoritmos genéticos multiobjetivos foram propostos por Saha e Kumar (2011), Lima *et al.* (2012) e uma formulação matemática utilizando multifluxo foi proposta por Santos *et al.* (2012).

Neste trabalho, um procedimento para gerar soluções Pareto ótimas para o bi-AGCDM é proposto inspirado no ϵ -constraint, Y. Haimes (1971) e no *Parallel Partitioning Method*, Lemesre *et al.* (2007). Em nosso conhecimento, não existem trabalhos na literatura “dedicados” a encontrar soluções Pareto ótimas para o bi-AGCDM. Assim sendo, uma das contribuições deste trabalho é fornecer as soluções Pareto ótimas para as instâncias utilizadas nos trabalhos de Lima *et al.* (2012) e Santos *et al.* (2012), as quais são provenientes do trabalho para o AGMRD de Santos *et al.* (2004). Este trabalho está organizado da seguinte forma: definições são fornecidas na seção

2. Em seguida, alguns procedimentos exatos clássicos para resolver problemas multiobjetivos são brevemente descritos na seção 3. Posteriormente, o procedimento proposto neste trabalho é detalhado na seção 4. Resultados preliminares são apresentados na seção 5. Finalmente, conclusões e perspectivas são fornecidas na seção 6.

2 Definições

Um problema de otimização multiobjetivo \mathcal{P} pode ser formulado como $\{ \min f(\vec{x}) \mid \vec{x} \in \mathcal{X} \}$, onde \mathcal{X} é o conjunto de soluções viáveis de \mathcal{P} (espaço de soluções) e $f(\vec{x})$ é um vetor de objetivos $(f_1(\vec{x}), f_2(\vec{x}), \dots, f_n(\vec{x}))$ a ser otimizado. Em problemas multiobjetivos, existem diversas soluções de compromisso entre os objetivos tratados. O conjunto de tais soluções é comumente referenciado como fronteira de Pareto que pode ser definida através do conceito de dominância, descrito abaixo. Seja um problema de minimização, sem perda de generalidade, uma solução x domina y se e somente se satisfaz as condições da Equação (1).

$$\begin{cases} f_k(\vec{x}) \leq f_k(\vec{y}) & \forall k \in 1..n \quad e \\ f_k(\vec{x}) < f_k(\vec{y}) & \exists k \in 1..n \end{cases} \quad (1)$$

Uma solução Pareto ótima $f_k(x^*)$ não é dominada por nenhuma solução pertencente ao espaço de busca \mathcal{X} . A fronteira de Pareto ótima contém o conjunto de soluções Pareto ótimas (não dominadas). Nas próximas seções deste texto, considera-se $n = 2$, caracterizando assim um problema biobjetivo.

3 Procedimentos exatos multiobjetivos

Os procedimentos ϵ -constraint, *Two-Phase Method* (TPM), *Parallel Partitioning Method* (PPM), descritos abaixo, são comumente empregados para calcular a fronteira de Pareto ótima em problemas de otimização multiobjetivos. O procedimento desenvolvido para o bi-AGCDM utiliza componentes do ϵ -constraint e do PPM.

O método ϵ -constraint é baseado no princípio de enumeração ϵ -constraint proposto por Y. Haimes (1971). A idéia geral deste procedimento é integrar um dos objetivos no conjunto de restrições e otimizar o outro. Assim sendo, o problema restrito de otimização \mathcal{P}' é definido como $\{ \min f_1(\vec{x}) \mid f_2(\vec{x}) \leq \epsilon; \vec{x} \in \mathcal{X} \}$. Inicialmente, o procedimento para o método ϵ -constraint calcula um ponto extremo para a função objetivo $f_1(x)$. Em seguida, uma nova solução é calculada utilizando restrições que possuem como limite superior, o valor do ponto extremo calculado. O procedimento é repetido sucessivamente até que mais nenhuma solução possa ser encontrada. Este método utiliza os valores das soluções encontradas como limite superior para restringir o espaço de busca e guiar a busca por soluções da fronteira de Pareto.

O TPM foi proposto por Y. Haimes (1971) e utiliza o conceito de agregação das funções objetivos da seguinte forma: as funções objetivos são devidamente normalizadas e tratadas como uma combinação linear dada por $\alpha f_1(x) + (1-\alpha)f_2(x)$, onde $\alpha \in [0, 1]$. Se $\alpha = 1$, o objetivo $f_1(x)$ é otimizado, enquanto que se $\alpha = 0$, $f_2(x)$ é otimizado. O procedimento para o TPM é realizado em duas fases. Na primeira fase, soluções não dominadas são calculadas utilizando o princípio de agregação e novas soluções são calculadas recursivamente entre as soluções encontradas. Quando o procedimento não encontra mais soluções não dominadas, passa-se para a próxima fase. A segunda fase é dedicada a encontrar os pontos dominados da fronteira de Pareto no espaço entre cada par de soluções.

O PPM possui três fases distintas e foi desenvolvido por Lemesre *et al.* (2007) para melhorar o TPM. A primeira fase do PPM consiste em definir pontos extremos através da otimização independente dos objetivos $f_1(x)$ e $f_2(x)$. Na segunda fase, partições de mesmo tamanho são geradas dentro dos limites encontrados na primeira fase. Em seguida, uma solução é gerada dentro de cada partição, utilizando o princípio ϵ -constraint. O problema \mathcal{P}' é considerado neste ponto

como um conjunto de problemas com um único objetivo, um para cada partição, a serem resolvidos recursivamente. Diversas estratégias para resolver na otimalidade estes sub-problemas podem ser consideradas, tais como, enumeração, algoritmos exatos do tipo *Branch-and-Bound*, utilização de softwares comerciais, etc. A terceira fase objetiva realizar uma busca nos retângulos formados entre pares de soluções em partições vizinhas.

A Figura 1 ilustra as modificações no espaço de busca para cada uma das fases do PPM. Os pontos extremos na fase 1 são identificados por quadrados pretos e a região cinza é a limitação do espaço de busca. As partições na fase 2 da Figura 1 foram realizadas utilizando o objetivo $f_2(x)$ e os pontos em forma de “x” correspondem as soluções encontradas dentro de cada partição. Finalmente, o retângulo cinza da Figura 1 apresentado na fase 3, exemplifica uma das zonas alvo para busca de novas soluções.

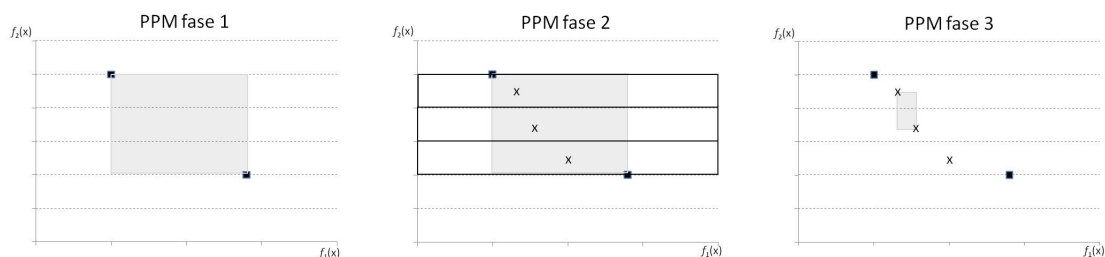


Figura 1: Fases do PPM.

A primeira e segunda fase do PPM são inspiradas no ϵ -constraint. As diferenças entre o PPM e o ϵ -constraint encontram-se no particionamento do espaço de busca e no fato de achar dois pontos extremos ao invés de um. A terceira fase do PPM usa idéias do TPM para povoar regiões entre as soluções. A busca por novas soluções é realizada através da combinação linear dos objetivos, tendo como alvo um retângulo formado entre cada par de soluções vizinhas. A principal diferença entre TPM e PPM neste aspecto é que as partições no PPM têm tamanhos idênticos.

4 Um procedimento para resolver o bi-AGCDM

Nesta seção, as etapas do procedimento desenvolvido para resolver o bi-AGCDM são detalhadas. Considera-se o grafo G definido na seção 1. Além disto, o custo e o diâmetro são referenciados respectivamente como o primeiro e o segundo objetivos.

O procedimento proposto utiliza a primeira fase do PPM. Assim sendo, um dos pontos extremos corresponde à AGM de G calculada através do algoritmo de Prim, Cormen *et al.* (1990). Se o grafo G considerado possui arestas com custos idênticos, diferentes AGM de G podem existir. A Figura 2 exemplifica tal situação. Um grafo é dado na Figura 2-(a) e duas AGM deste grafo são apresentadas respectivamente nas Figuras 2-(b) e (c), com custos iguais a 37 e diâmetros respectivos iguais a $D = 5$ e $D = 7$. É importante ressaltar que a solução da Figura 2-(c) é dominada pela solução da Figura 2-(b) com relação à minimização do diâmetro.

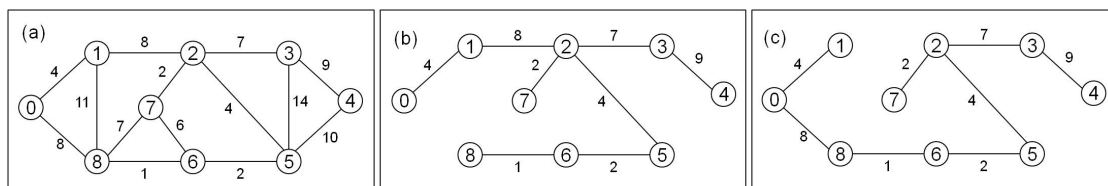


Figura 2: Exemplo de um grafo onde a AGM não é única.

O outro ponto extremo do espaço de solução corresponde a uma árvore geradora de diâmetro mínimo. Inicialmente, calcula-se o diâmetro do grafo que é o maior dos menores

caminhos, em quantidade de arestas, entre cada par de nós $[i, j] \in G$. Se o grafo fornecido tem diâmetros $D = 2$ ou $D = 3$, pode-se calcular a AGM de diâmetro mínimo em tempo polinomial. Quando $D = 2$, o procedimento constrói, a cada iteração, uma árvore geradora considerando cada vértice $i \in G$ como central e os outros vértices $j \in V \setminus \{i\}$ são conectados através da aresta $[i, j]$. Em seguida, basta comparar as $O(|V|)$ árvores de G . Este procedimento tem complexidade computacional de pior caso $O(|V|^2)$. Se $D = 3$, a cada iteração uma aresta $[i, j] \in E$ é tomada como aresta central da árvore. Posteriormente, os vértices $k \in V \setminus \{i, j\}$ são conectados em uma das extremidades de $[i, j]$. A conexão $[k, i]$ ou $[k, j]$ de menor custo é realizada. Assim, $O(|E|)$ árvores são comparadas e o cálculo de uma AGM com $D = 3$, consome no pior caso $O(|E| \cdot |V|)$. Caso o diâmetro $D \geq 4$, o problema passa a ser NP-difícil e os modelos matemáticos descritos na seção 4.1 são empregados para encontrar este ponto extremo através de um software de Programação Linear Mista (PLM).

Diferentes árvores de diâmetro mínimo podem existir em um grafo G . Um exemplo de tal situação é apresentado na Figura 3. Um grafo G é dado na Figura 3-(a), cujo diâmetro é igual a 4. Duas árvores de diâmetro mínimo são fornecidas respectivamente nas Figuras 3-(b) e (c), sendo que a primeira tem custo igual a 47 e a segunda tem custo igual a 54. Desse modo, a solução da Figura 3-(b) domina a solução da Figura 3 - (c) por causa do valor da função de custo.

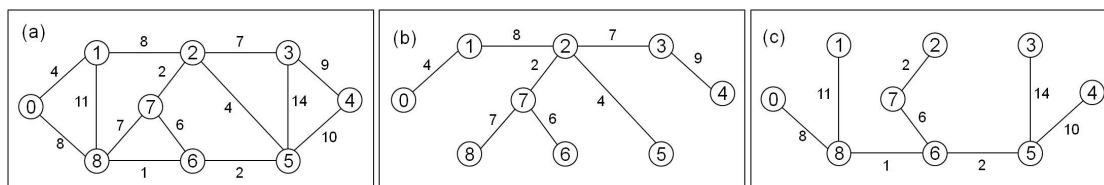


Figura 3: Exemplo de um grafo que possui várias árvores de diâmetro mínimo.

O segundo passo do procedimento proposto para o bi-AGCDM consiste em particionar o espaço de busca em retângulos idênticos como na segunda fase do PPM. Para o bi-AGCDM, esta divisão é realizada utilizando a função objetivo relativa ao diâmetro, pois corresponde a valores inteiros e cada retângulo tem altura igual a 1. Considerando os pontos extremos encontrados na primeira fase, divide-se o espaço de busca de modo que cada retângulo contenha um valor alvo (inteiro positivo) para cada diâmetro.

Após o particionamento, uma otimização é realizada para encontrar uma solução em cada partição. A maneira de realizar esta otimização baseia-se no método ϵ -constraint, pois um dos objetivos é considerado como restrição. Encontrar uma AGM com $3 < D < |V| - 1$ corresponde a tratar um problema NP-Difícil. Isto ocorre porque uma vez que o diâmetro é considerado como restrição, o problema passa a ser um AGMRD. Nestes casos, provar otimalidade está condicionado e limitado pela instância fornecida. Mesmo assim, é interessante analisar a fronteira de Pareto para algumas instâncias importantes da literatura resolvidas na perspectiva da otimização multiobjetivo. Neste trabalho, o procedimento para resolver na otimalidade instâncias com restrições de tais diâmetros consiste em resolver recursivamente um modelo matemático para o AGMRD através de um software PLM.

As formulações matemáticas utilizadas correspondem à versão melhorada dos modelos de Achuthan *et al.* (1994), proposta em Santos *et al.* (2004) e descritas na Seção 4.1. Embora estes modelos tenham uma relaxação linear mais fraca que os modelos baseados em multifluxo, Gouveia e Magnanti (2003), softwares PLM trabalham eficientemente sob tais modelos. Uma possível razão é a quantidade de restrições e variáveis que nos modelos multifluxo são da ordem de $O(|V^4|)$, enquanto que nos modelos propostos por Achuthan *et al.* (1994) são significativamente menores, ver Seção 4.1.

Os modelos apresentados na Seção 4.1 buscam AGM com diâmetros $D \leq d$, onde $d \geq 4$

é um inteiro positivo. Uma vantagem do emprego desses modelos é se a solução retornada possui diâmetro $D < d$, significa que a solução com $D = d$ é dominada. Assim, o procedimento de resolução não precisa necessariamente ser executado para todos os diâmetros, o que implica uma economia de tempo de cálculo.

O procedimento proposto repete os passos mencionados acima e pára antes de atingir o ponto extremo que corresponde a uma árvore geradora de diâmetro mínimo. Note que a terceira fase do PPM não se aplica ao bi-AGCDM porque o diâmetro é definido no domínio dos inteiros positivos. Consequentemente, não existem soluções entre árvore de diâmetros vizinhos.

4.1 Formulações utilizadas para calcular os problemas restritos

Dois modelos matemáticos são considerados, um para o caso em que o diâmetro considerado é par e outro para o caso em que o diâmetro a ser tratado é ímpar. Esses modelos utilizam a propriedade definida por Handler (1978) que determina se o diâmetro de uma árvore geradora \mathcal{T} de G é par, existe um vértice central $i \in V$ tal que nenhum outro vértice de \mathcal{T} tenha uma altura maior do que $L = D/2$ arestas de distância i . Quando o diâmetro de uma árvore geradora \mathcal{T} de G é ímpar, existe uma aresta central $[i, j] \in \mathcal{T}$ tal que todos os outros vértices $k \in V \setminus \{i, j\}$ estão a no máximo $L = (D - 1)/2$ arestas de uma das extremidades da aresta central $[i, j]$.

Em ambas formulações, um nó artificial r é introduzido no grafo G com custos $c_{ri} = 0 \forall i \in V$. Um novo grafo G' é obtido de G através da introdução de um nó artificial r e da orientação das arestas, da seguinte forma: para cada aresta $[i, j] \in E$, com $i < j$ do grafo G , dois arcos são criados em G' , (i, j) e (j, i) , com custos $c_{ij} = c_{ji}$. Assim, $G' = (V', A')$, onde o conjunto de vértices é definido por $V' = V \cup \{r\}$ e o conjunto de arestas é dado por $A' = A \cup \{(r, 1), \dots, (r, |V|)\}$. A orientação dos arcos é realizada para controlar a construção da AGM a partir do nó artificial, mas desconsideradas após a obtenção da solução final.

O modelo para o caso par é dado nas equações de (2) a (7) e utiliza variáveis binárias x_{ij} que determinam se um arco (i, j) é selecionado para participar da AGM ($x_{ij} = 1$), ou não ($x_{ij} = 0$). Além disto, variáveis u_i são associadas aos vértices $i \in V'$. Estas variáveis determinam uma ordem de visita na solução para cada vértice (ordem topológica), sabendo-se que para o nó artificial $u_r = 0$. Assim, o vértice central l da AGM tem $u_l = 1$. A função objetivo é dada em (2). A equação (3) assegura que o vértice artificial r conecta-se a exatamente um vértice em V , i.e, o vértice central. A restrição (4) define que um único arco deve ser incidente a cada vértice de V . As restrições (5) correspondem a uma adaptação das clássicas restrições MTZ de Miller *et al.* (1960), com o procedimento de *lifting* proposto por Desrochers e Laporte (1991). Estas restrições definem uma ordem topológica para cada nó na solução, assim como eliminam subciclos. As variáveis são definidas de (6) a (7). Este modelo possui $O(|V'| + |A'|)$ variáveis e $O(|V| + |A'|)$ restrições.

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (2)$$

$$\sum_{j \in V} x_{rj} = 1 \quad (3)$$

$$\sum_{(i,j) \in A'} x_{ij} = 1 \quad \forall j \in V \quad (4)$$

$$u_i - u_j + (L + 1)x_{ij} + (L - 1)x_{ji} \leq L \quad \forall (i, j) \in A \quad (5)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A' \quad (6)$$

$$0 \leq u_i \leq L + 1 \quad \forall i \in V' \quad (7)$$

O modelo para o caso ímpar é apresentado nas equações de (8) a (16) e utiliza variáveis x_{ij} e u_i como no caso par. Além disto, possui variáveis binárias $z_{ij}, \forall [i, j] \in E$ com $i < j$, que são

responsáveis pela seleção da aresta central. A aresta central é selecionada quando $z_{ij} = 1$, em caso contrário $z_{ij} = 0$. Nesta formulação, o vértice artificial r se conecta a exatamente dois vértices $p, q \in V$, o que determina implicitamente a seleção de uma aresta central $[p, q] \in E$. Após a obtenção da solução final, além das orientações dos arcos serem desconsideradas, as conexões $[r, p]$ e $[r, q]$ também são e a aresta central é $[p, q]$. A função objetivo é fornecida em (8) e inclui o termo $\sum_{[i,j] \in E} c_{ij} z_{ij}$ para contabilização do custo da aresta central na solução. A equação (9) estabelece que o vértice artificial r conecta-se a exatamente dois vértices de V . As restrições (10) determinam que um único arco é incidente a cada vértice de V . A equação (11) assegura que somente uma aresta central existe na solução final. As desigualdades não lineares em (12) relacionam as variáveis de decisão z_{ij} e x_{ij} e asseguram que o vértice artificial se conecta a exatamente dois nós de V . Estas restrições podem ser facilmente linearizadas como proposto em Santos *et al.* (2004). As restrições MTZ com *lifting*, definidas em (13), são similares ao caso par. O domínio das variáveis é determinado de (14) a (16). Este modelo possui $O(|V'| + |A'|)$ variáveis e $O(|V| + |A'|)$ restrições.

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{[i,j] \in E} c_{ij} z_{ij} \quad (8)$$

$$\sum_{j \in V} x_{rj} = 2 \quad (9)$$

$$\sum_{(i,j) \in A'} x_{ij} = 1 \quad \forall j \in V \quad (10)$$

$$\sum_{[i,j] \in E} z_{ij} = 1 \quad (11)$$

$$z_{ij} \geq x_{ri} \cdot x_{rj} \quad \forall [i, j] \in E \quad (12)$$

$$u_i - u_j + (L + 1)x_{ij} + (L - 1)x_{ji} \leq L \quad \forall (i, j) \in A \quad (13)$$

$$0 \leq u_i \leq L + 1 \quad \forall i \in V' \quad (14)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in A' \quad (15)$$

$$z_{ij} \in \{0, 1\} \quad \forall [i, j] \in E \quad (16)$$

5 Resultados preliminares

O procedimento proposto foi desenvolvido no compilador Dev-C++ 4.9.9.2 através da linguagem de programação C ANSI. Os testes foram realizados em uma máquina com o sistema operacional Windows 7, 64 bits, com 4Gb de memória RAM e processador Intel core i5 com 2,53GHz. Na segunda fase do procedimento proposto, o software Cplex versão 12.2 é empregado para resolução exata dos problemas com $D \geq 4$.

Resultados preliminares são mostrados na Tabela 1 para as instâncias provenientes de Santos *et al.* (2004). Cada linha corresponde a uma instância. A primeira coluna identifica o nome das instâncias formados da seguinte forma: $k_Vy_Az_Dw$, onde k indica se o grafo é completo (“C”) ou esparso (“S”), Vy fornece a quantidade y de nós do grafo, Az especifica a quantidade z de arestas do grafo e Dw corresponde ao diâmetro w utilizado em Santos *et al.* (2004). Algumas instâncias tratadas em Santos *et al.* (2004) correspondem a grafos idênticos, com diâmetros diferentes, tais como as de 15 nós. Aqui, considera-se somente as instâncias de grafos distintos. A coluna “Q” indica a quantidade de pontos ótimos encontrados na fronteira de Pareto. Da coluna 3 até a quatorze, os diâmetros são especificados e cada célula contém os custos encontrados para o respectivo diâmetro e instância. Finalmente, a coluna *tempo(s)* mostra os tempos totais em segundos para achar todos os pontos ótimos da fronteira de Pareto.

Os resultados apresentados na Tabela 1 indicam que, exceto a instância $s_v40_a100_d4$, o espaço de soluções contém no máximo 13 soluções (13 diâmetros alvos). Em termos de tempo

Tabela 1: Resultados de fronteiras de Pareto ótimas para o bi-AGCDM.

| Instâncias | Q | Diâmetros | | | | | | | | | | | | | | | | | | | tempo(s) |
|---------------|----|-----------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|----------|
| | | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | |
| C_V10_A45_D4 | 8 | 392 | 279 | 252 | 234 | 218 | 210 | 208 | 206 | - | - | - | - | - | - | - | - | - | - | - | 0,70 |
| C_V10_A45_D5 | 5 | 382 | 330 | 247 | 230 | 218 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | 0,31 |
| C_V10_A45_D6 | 5 | 360 | 275 | 235 | 226 | 221 | - | - | - | - | - | - | - | - | - | - | - | - | - | - | 0,26 |
| C_V10_A45_D7 | 8 | 447 | 292 | 250 | 226 | 211 | 203 | 199 | 198 | - | - | - | - | - | - | - | - | - | - | - | 0,72 |
| C_V10_A45_D8 | 6 | 389 | 297 | 243 | 236 | 233 | 232 | - | - | - | - | - | - | - | - | - | - | - | - | - | 0,30 |
| C_V10_A45_D9 | 7 | 483 | 347 | 284 | 270 | 263 | 256 | 254 | - | - | - | - | - | - | - | - | - | - | - | - | 0,55 |
| C_V15_A105_D4 | 11 | 681 | 482 | 346 | 331 | 314 | 303 | 295 | 290 | 286 | - | 277 | - | - | - | - | - | - | - | - | 25,16 |
| C_V15_A105_D8 | 7 | 491 | 390 | 291 | 274 | 256 | 247 | 240 | - | - | - | - | - | - | - | - | - | - | - | - | 23,81 |
| C_V20_A190_D4 | 11 | 629 | 481 | 349 | 330 | 302 | 287 | 282 | 276 | 274 | - | 271 | - | - | - | - | - | - | - | - | 520,53 |
| C_V20_A190_D5 | 11 | 792 | 641 | 441 | 414 | 382 | 367 | 352 | 350 | 347 | 343 | - | - | - | - | - | - | - | - | - | 199,01 |
| C_V20_A190_D6 | 14 | 681 | 498 | 343 | 316 | 298 | 287 | 278 | 276 | 272 | 269 | 265 | 263 | 261 | - | - | - | - | - | - | 387,10 |
| C_V20_A190_D7 | 8 | 716 | 546 | 414 | 376 | 345 | 333 | 321 | 316 | - | - | - | - | - | - | - | - | - | - | - | 248,32 |
| C_V20_A190_D8 | 13 | 789 | 592 | 421 | 382 | 353 | 344 | 331 | 323 | 317 | 311 | 299 | 295 | - | - | - | - | - | - | - | 2241,32 |
| C_V20_A190_D9 | 12 | 822 | 599 | 418 | 390 | 356 | 345 | 333 | 327 | 324 | - | 321 | 316 | 313 | - | - | - | - | - | - | 1014,45 |
| S_V20_A50_D4 | 10 | 693 | 601 | 442 | 410 | 369 | 357 | 340 | 334 | - | 332 | - | - | - | - | - | - | - | - | - | 11,47 |
| S_V20_A50_D5 | 9 | 807 | 694 | 417 | 381 | 347 | 330 | 322 | 318 | 315 | - | - | - | - | - | - | - | - | - | - | 14,15 |
| S_V20_A50_D6 | 10 | 755 | 617 | 395 | 366 | 329 | 323 | 321 | 309 | 303 | 300 | - | - | - | - | - | - | - | - | - | 36,35 |
| S_V20_A50_D7 | 9 | 707 | 598 | 404 | 386 | 370 | 362 | 358 | 357 | 356 | - | - | - | - | - | - | - | - | - | - | 22,86 |
| S_V20_A50_D8 | 10 | 822 | 679 | 434 | 411 | 379 | 373 | 366 | 362 | 359 | 357 | - | - | - | - | - | - | - | - | - | 49,60 |
| S_V40_A100_D4 | 19 | 1574 | 1403 | 755 | 706 | 615 | 589 | 567 | 552 | 546 | 538 | 528 | 525 | 523 | 508 | 505 | 503 | 501 | 500 | - | 46690,46 |

de processamento, encontrar todas as soluções da fronteira de Pareto ótima para estas instâncias testes e os modelos matemáticos utilizados consumiu no máximo 50 segundos para as instâncias de grafos completos com 10 e 15 nós e de grafos esparsos com 20 nós, menos de 40 minutos para as instâncias de 20 nós em grafos completos e aproximadamente 13 horas para a instância com 40 nós. Observou-se também que as instâncias de diâmetros pequenos, por exemplo, $D = 4$ e $D = 5$, consomem a maior parte do tempo de processamento. Em geral, o problema torna-se mais fácil a medida que se aproxima do diâmetro da AGM. Além disto, instâncias com arestas de custos idênticos são mais difíceis de serem resolvidas. Possivelmente, a resolução de tais problemas implica uma combinatória mais significativa na árvore de *Branch-and-Bound*, como é o caso da instância *s.v40_a100_d4*.

Os gráficos da Figura 4 ilustram as fronteiras de Pareto ótimas para quatro instâncias, cujo resultados são representativos para o conjunto de instâncias testadas. Algumas fronteiras Pareto ótimas geradas aparecem com pontos dominados, como é o caso da instância *C_V20_A190_D4* com $D = 11$ e $D = 13$. Outras instâncias possuem a solução da AGM dominada, como é o caso da instância *C_V20_A190_D5*, cuja AGM de diâmetro $D = 13$ é dominada pela solução com $D = 12$. Finalmente, observa-se ainda que na maioria dos casos, a função de custo das soluções com $D = 2$ e $D = 3$ tem valores bastante elevados em relação aos outros diâmetros, gerando um esparsamento significativo na fronteira de Pareto ótima nessas regiões do espaço de busca.

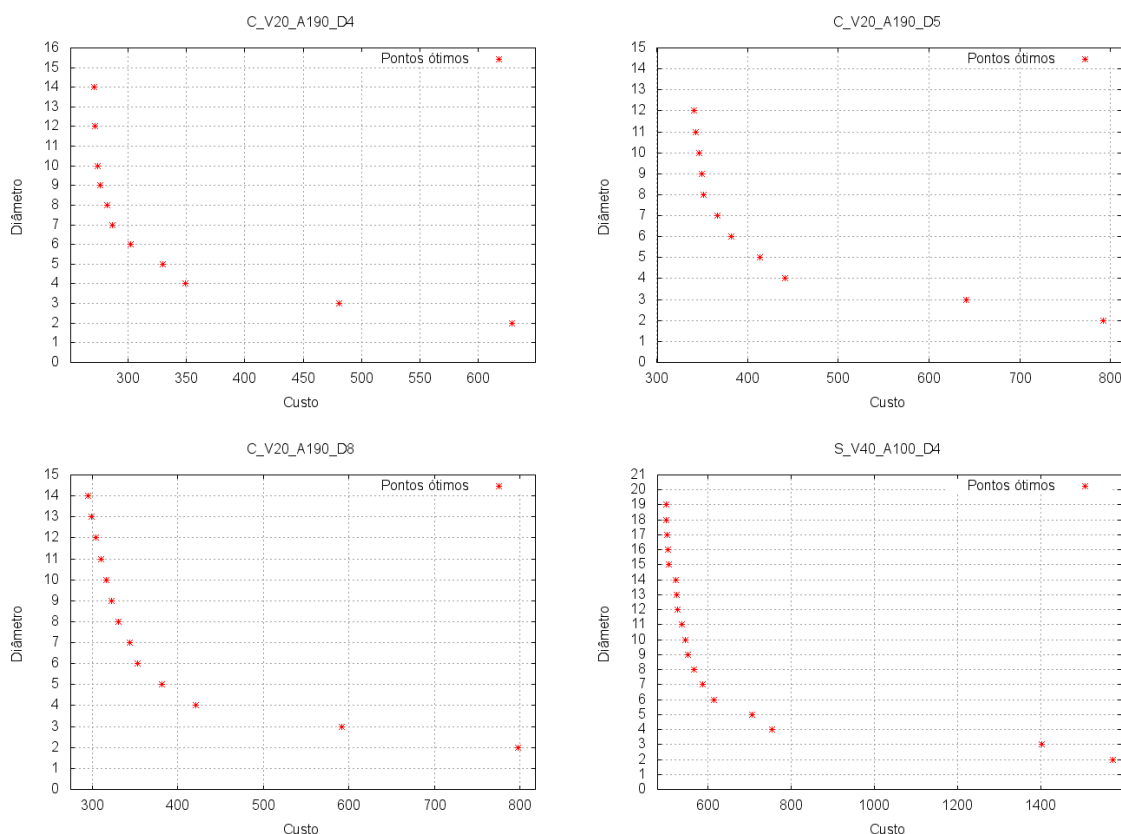


Figura 4: Fronteira de Pareto ótima para quatro instâncias.

6 Conclusões e perspectivas

Neste trabalho, um procedimento para calcular fronteira de Pareto ótimas para o bi-AGCDM é apresentado. Este procedimento utiliza componentes do ϵ -constraint e do PPM, além de empregar formulações e alguns algoritmos existentes na literatura.

Embora a geração das fronteiras de Pareto ótimas esteja limitada pelo fato dos subproblemas tratados serem NP-difíceis para alguns diâmetros, as fronteiras de Pareto geradas para algumas instâncias da literatura constituem uma contribuição interessante. Em particular, esses resultados podem servir de base para análises da qualidade dos resultados heurísticos para o bi-AGCDM.

Em relação ao AGMRD, nós contribuimos por generalizar e abordar este problema na perspectiva da otimização multiobjetivo. Especialmente, utiliza-se o conceito de dominância e fronteira de Pareto. Assim, é possível realizar análises sobre o espaço de busca. Por outro lado, este tipo de análise também pode ser útil a decisores que se interessam por aplicações do AGMRD, pois pode-se observar que alguns diâmetros não são interessantes por constituírem soluções dominadas.

O procedimento desenvolvido pode ser melhorado em alguns aspectos. Por exemplo, pode-se desenvolver ou integrar outros métodos para a resolução dos subproblemas em cada partição, tais como *Branch-and-Cut*, geração de colunas e algoritmos de enumeração. Uma outra perspectiva de trabalhos futuros é a utilização de heurísticas e matheurísticas como suporte para acelerar as buscas por soluções Pareto ótimas.

Referências

- Achuthan, N. R., L.Caccetta, Caccetta, P. A. e Geelen, J. F.** (1994), Computational methods for the diameter restricted minimum weight spanning tree problem. *Australasian Journal of Combinatorics*, v. 10, p. 51–71.
- Arroyo, J. E. C., Vieira, P. S. e Vianna, D. S.** (2008), A GRASP algorithm for the multi-criteria minimum spanning tree problem. *Annals of Operations Research*, v. 159, p. 125–133.
- Cormen, T., Leiserson, C. e Rivest, R.** *Introduction to algorithms*. McGraw-Hill, New York, 1990.
- Desrochers, M. e Laporte, G.** (1991), Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. *Operations Research Letters*, v. 10, p. 27–36.
- Garey, M. e Johnson, D.** *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman, New York, 1979.
- Gouveia, L. e Magnanti, T. L.** (2003), Network flow models for designing diameter-constrained minimum-spanning and steiner trees. *Networks*, v. 41, p. 159–173.
- Gouveia, L., Simonetti, L. e Uchoa, E.** (2011), Modeling hop-constrained and diameter-constrained minimum spanning tree problems as steiner tree problems over layered graphs. *Mathematical Programming*, v. 128, p. 123–148.
- Gruber, M. e Raidl, G. R.** Variable neighborhood search for the bounded diameter minimum spanning tree problem. Hansen, P., Mladenovic, N., Pérez, J. A. M., Batista, B. M. e MorenoVega, J. M. (Eds.), *Proceedings of the 18th Mini Euro Conference on Variable Neighborhood Search*, p. 1–11, Tenerife, 2005.
- Handler, G.** (1978), Minimax location of a facility in an undirected graph. *Transportation Science*, v. 7, p. 287–293.
- Ho, J.-M., Lee, D. T., Chang, C.-H. e Wong, K.** (1991), Minimum diameter spanning trees and related problems. *SIAM Journal on Computing*, v. 20, n. 5, p. 987–997.
- Lemesre, J., Dhaenens, C. e Talbi, E.** (2007), Parallel partitioning method (PPM): A new exact method to solve bi-objective problems. *Computers & Operations Research*, v. 34, n. 8, p. 2450–2462.

- Lima, D. R., Santos, A. C. e Aloise, D. J.** Um algoritmo evolucionário NSGA-II para resolver o problema bi-objetivo da árvore geradora de custo e diâmetro mínimos. *Anais do XVI Congresso Latino-Iberoamericano de Investigación Operativa, XLIV Simpósio Brasileiro de Pesquisa Operacional (CLAIO-SBPO)*, p. 689–699, 2012.
- Lucena, A., Ribeiro, C. e Santos, A. C.** (2010), A hybrid heuristic for the diameter constrained minimum spanning tree problem. *Journal of Global Optimization*, v. 46, p. 363–381.
- Marathe, M. V., Ravi, R., Sundaram, R., Ravi, S., Rosenkrantz, D. J. e HuntIII, H. B.** (1998), Bicriteria network design problems. *Journal of Algorithms*, v. 28, n. 1, p. 142–171.
- Miller, C., Tucker, A. e Zemlin, R.** (1960), Integer programming formulations and traveling salesman problems. *Journal of the ACM*, v. 7, p. 326 – 329.
- Requejo, C. e Santos, E.** (2009), Greedy heuristics for the diameter-constrained minimum spanning tree problem. *Journal of Mathematical Sciences*, v. 161, p. 930–943.
- Saha, S. e Kumar, R.** (2011), Bounded-diameter MST instances with hybridization of multi-objective EA. *International Journal of Computer Applications*, v. 18, p. 17–25.
- Santos, A. C., Lima, D. R. e Aloise, D. J.** Modeling the bi-objective diameter minimum spanning tree problem. *Proceedings of the Global Optimization Workshop 2012*, p. 125–128, Natal, Brazil, 2012.
- Santos, A. C., Lucena, A. e Ribeiro, C. C.** (2004), Solving diameter constrained minimum spanning tree problem in dense graphs. *Lecture Notes in Computer Science*, v. 3059, p. 458–467.
- Sourd, F. e Spanjaard, O.** (2008), A multiobjective branch-and-bound framework: Application to the biobjective spanning tree problem. *INFORMS Journal on Computing*, v. 20, n. 3, p. 472–484.
- Steiner, S. e Radzik, T.** (2008), Computing all efficient solutions of the biobjective minimum spanning tree problem. *Computers & Operations Research*, v. 35, n. 1, p. 198–211.
- T. F. Noronha, C. C. Ribeiro, A. C. S.** (2010), Solving diameter-constrained minimum spanning tree problems by constraint programming. *International Transactions in Operational Research*, v. 17, n. 5, p. 653–665.
- Y. Haimes, L. Ladson, D. W.** (1971), On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems, Man and Cybernetics*, v. SMC-1, n. 3, p. 296–297.
- Zhou, G. e Gen, M.** (1999), Genetic algorithm approach on multi-criteria minimum spanning tree problem. *European Journal of Operational Research*, v. 114, p. 141–152.