57º SBPO
SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL
Gramado • RS
DE 5 A 9 DE OUTUBRO DE 2025

# Decomposição de Benders Acelerada para o Problema da Rota Mediana Generalizada em Dois Níveis

**Nickolas Gueller, Rafael Martinelli, Bruno Fânzeres**
Departamento de Engenharia Industrial, PUC-Rio
Rua Marquês de São de Vicente, 225 – Gávea, Rio de Janeiro, RJ – Brasil – 22541-900
nickolas@aluno.puc-rio.br
martinelli@puc-rio.br
bruno.santos@puc-rio.br

**RESUMO**

Este artigo propõe um algoritmo baseado em decomposição de Benders para resolver o Problema da Rota Mediana Generalizada em Dois Níveis (PRMGDN), que modela sistemas de distribuição hierárquicos divididos em clusters. O algoritmo incorpora técnicas modernas de aceleração, como a abordagem em duas fases, uso de uma única árvore de *branch-and-bound*, subproblemas normalizados e uso de desigualdades válidas. Os experimentos computacionais realizados em 32 instâncias de até 417 nós da GTSP-Lib demonstram a eficácia da abordagem proposta: todas as instâncias foram resolvidas até a otimalidade, com tempo médio de apenas 395.69 segundos. Isso representa uma melhoria significativa em relação ao método do trabalho original, que apresentou gaps médios de 3.10%, e à decomposição automática do CPLEX, que apresentou gaps médios de 0.08%. Os resultados confirmam que a abordagem desenvolvida é robusta, eficiente e promissora para problemas de projeto de redes com estrutura hierárquica e custos fixos.

**PALAVRAS CHAVE. Programação Inteira. Decomposicão de Benders. Projeto de Redes.**

**OD – Otimização Discreta, L&T – Logística e Transportes**

**ABSTRACT**

This paper proposes a Benders decomposition-based algorithm to solve the Two-Level Generalized Median Tour Problem (TLGMTP), which models hierarchical distribution systems partitioned into clusters. The algorithm incorporates modern acceleration techniques, such as the two-phase approach, the use of a single branch-and-bound tree, normalized subproblems, and the addition of valid inequalities. Computational experiments conducted on 32 instances with up to 417 nodes from the GTSP-Lib demonstrate the effectiveness of the proposed approach: all instances were solved to optimality, with an average solution time of only 395.69 seconds. This represents a significant improvement over the original work method, which presented average GAPS of 3.10%, and over the CPLEX automatic Benders decomposition, which presented average GAPS of 0.08%. The results confirm that the proposed method is robust, efficient, and promising for network design problems with hierarchical structure and fixed costs.

**KEYWORDS. Integer Programming. Benders Decomposition. Network Design.**

**DO - Discrete Optimization, L&T – Logistics and Transport**

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

## 1. Introdução

Este estudo foca em sistemas de distribuição nos quais os pontos de demanda são naturalmente particionados em regiões (clusters) devido a fatores como barreiras geográficas (rios, rodovias), restrições de transporte (alcance dos veículos) ou divisões geopolíticas. Esses sistemas surgem frequentemente na entrega de bens essenciais, incluindo alimentos, água, ajuda humanitária e vacinas. Para modelar tais cenários, Obreque et al. [2020] propuseram o Problema da Rota Mediana Generalizada (PRMG), um problema de programação linear inteira-mista (PLIM) onde deseja-se planejar uma rota que visite todos os clusters, enquanto nós não visitados são atribuídos a nós visitados dentro do mesmo cluster.

A partir desse modelo, Obreque et al. [2024] propuseram uma extensão denominada Problema da Rota Mediana Generalizada em Dois Níveis (PRMGDN), com o objetivo de representar a hierarquia observada em muitos sistemas de distribuição do mundo real. O PRMGDN introduz uma estrutura de roteamento em três camadas. No nível superior, um veículo primário parte de um depósito central e percorre uma rota que visita um subconjunto de clusters, denominados clusters primários. Os demais clusters são classificados como secundários e atendidos por veículos dedicados (caminhos secundários), que partem de nós previamente visitados. Por fim, dentro de cada cluster, seja ele primário ou secundário, pequenos veículos locais (como vans ou motocicletas) realizam a atribuição dos nós restantes. Essa arquitetura hierárquica visa garantir a cobertura total da rede ao passo que minimiza os custos logísticos.

A Figura 1 ilustra uma solução representativa do PRMGDN aplicada a uma rede composta por 14 nós organizados em 6 clusters. Os clusters visitados pelo veículo primário (formando a rota principal) estão destacados em vermelho, enquanto os clusters secundários, alcançados por caminhos direcionados, estão em azul. As setas vermelhas indicam a rota do veículo primário, as setas azuis representam os caminhos unidirecionais percorridos pelos veículos secundários, e as linhas tracejadas retratam as atribuições dos nós não visitados para nós já cobertos pela rota primária ou pelos caminhos secundários dentro do mesmo cluster.

Por exemplo, o Cluster 4 é totalmente visitado pelo veículo primário, com todos os seus nós incluídos na rota principal. Já o Cluster 5, seu vizinho, é alcançado por caminhos secundários originados do Nó 6. Enquanto isso, o Cluster 1 funciona exclusivamente como o depósito e dá acesso ao Cluster 2 por meio de um veículo secundário até o Nó 9, que por sua vez serve como um hub para os Nós 10 e 11 atribuídos. A seleção dos clusters primários e secundários é determinada endogenamente pela solução do problema, em vez de ser pré-definida.

Estruturalmente, tanto o PRMG quanto o PRMGDN estão intimamente relacionados a problemas clássicos de pesquisa operacional, como o Problema da Rota Mediana e o Problema do Caixeiro Viajante Clusterizado, tornando-os problemas relevantes e versáteis para resolver desafios logísticos complexos.

O PRMGDN foi originalmente motivado pela necessidade de otimizar sistemas de distribuição de suprimentos de saúde na região do Biobío, no Chile conforme descrito por Obreque et al. [2024]. Nesse estudo, os autores propuseram três formulações PLIM distintas, bem como um algoritmo do tipo *Branch-and-Cut*. Experimentos computacionais com instâncias da biblioteca GTSP-lib evidenciaram a relevância do modelo, mas também revelaram limitações importantes: diversas instâncias não puderam ser resolvidas até a otimalidade dentro de um tempo computacional razoável.

Motivado por essas dificuldades, o presente trabalho investiga o uso da decomposição de Benders como estratégia para resolver o PRMGDN de forma mais eficiente. O problema se enquadra na classe dos problemas de projeto de redes com custos fixos, que envolvem decisões sobre a ativação de elementos da rede (como arcos ou nós), cada um associado a um custo fixo.

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

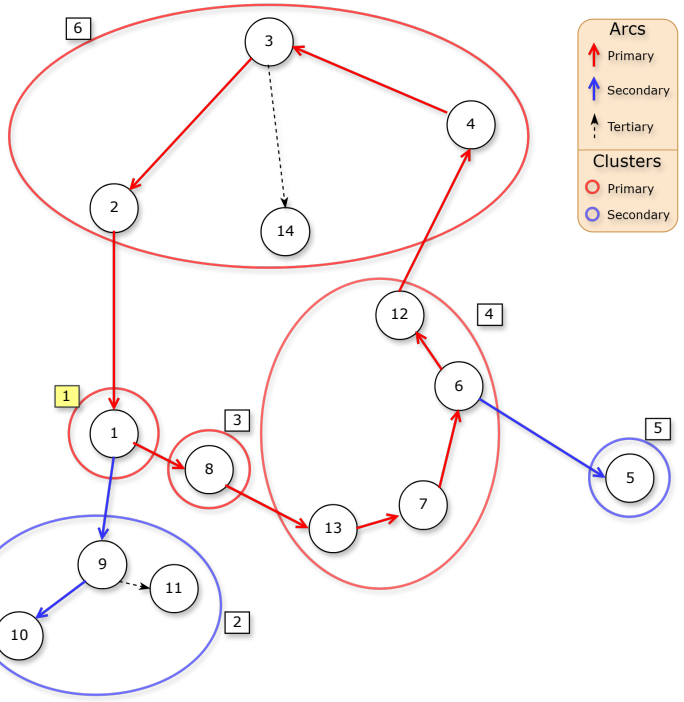



Figura 1: TLGMTP solution for a network composed by 14 nodes and 6 clusters.

Nesse domínio, a decomposição de Benders tem se mostrado eficaz, conforme destacado por Costa [2005], o que torna essa abordagem uma alternativa promissora para melhorar a escalabilidade e a qualidade das soluções obtidas.

Este artigo está organizado da seguinte forma: a Seção 2 apresenta a formulação matemática do PRMGDN. A Seção 3 descreve a aplicação da decomposição de Benders ao problema, enquanto a Seção 4 discute os resultados computacionais obtidos. Por fim, a Seção 5 apresenta as considerações finais e sugestões para trabalhos futuros.

## 2. Formulação Matemática

Esta seção apresenta a formulação PLIM adotada para resolver o PRMGDN. No trabalho original de Obreque et al. [2024], três formulações foram propostas. Aqui, selecionamos a formulação baseada em fluxo, que é particularmente adequada à decomposição de Benders devido à presença variáveis de fluxo contínuas, que naturalmente se separam das decisões de roteamento binárias.

Seja $G = (N, A)$ um grafo direcionado completo, onde $N$ é o conjunto de nós (incluindo o depósito, Nó 1) e $A = \{(i, j) : i, j \in N \wedge i \neq j\}$ é o conjunto de arestas. Seja $K$ o conjunto de clusters, e seja $N_p \subseteq N$ o conjunto de nós pertencentes ao cluster $p \in K$. Assume-se que os clusters são disjuntos, ou seja, $N_p \cap N_q = \emptyset$ para todos $p \neq q$ em $K$, e que eles particionam o conjunto de nós, tal que $\bigcup_{p \in K} N_p = N$.

O principal objetivo consiste em identificar um subgrafo composto por uma rota primária (tour) que, com as opções de estabelecer caminhos ou atribuições, garanta que cada nó seja visitado ou atribuído. Além disso, a rota primária começa e termina no depósito, visitando um ou mais clusters (denominados clusters primários). Adicionalmente, caminhos secundários só podem ser estabelecidos entre um nó incluído no tour e um cluster não visitado (denominado cluster secundário),

DOI: 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

visitando ao menos um nó dentro do respectivo cluster secundário. Por fim, os nós não diretamente visitados nem pelo tour primário nem pelos caminhos secundários devem ser atribuídos a um nó que tenha sido visitado. Essas decisões são modeladas por um conjunto de variáveis binárias $(x, y, w)$, respectivamente. As duas primeiras consideram um conjunto de custos primários $(c_{i,j})$ e secundários $(d_{i,j})$ para todo $(i, j) \in A$, enquanto a última considera um custo de atribuição $(r_{i,j})$ para todo $(i, j) \in N_p$ para cada $p \in K$. Por definição, $r_{i,j} = 0$ se $i = j$.

$$\min_{x,y,w,f} \sum_{i\in N}\sum_{j\in N:i\neq j} c_{i,j}x_{i,j} + \sum_{i\in N}\sum_{j\in N:i\neq j} d_{i,j}y_{i,j} + \sum_{p\in K}\sum_{i\in N_p}\sum_{j\in N_p} r_{i,j}w_{i,j} \tag{1}$$

Sujeito a:

$$\sum_{j\in N:j\neq 1} x_{1,j} = 1 \tag{2}$$

$$\sum_{i\in N:i\neq j} x_{i,j} = \sum_{i\in N:i\neq j} x_{j,i} \qquad \forall j \in N \tag{3}$$

$$\sum_{i\in N\setminus N_p}\sum_{j\in N_p} (x_{i,j} + y_{i,j}) = 1 \qquad \forall p \in K \tag{4}$$

$$\sum_{i\in N:i\neq j} (x_{i,j} + y_{i,j}) + \sum_{i\in N_p:i\neq j} w_{i,j} = 1 \qquad \forall p \in K, j \in N_p \tag{5}$$

$$\sum_{j\in N_q} y_{i,j} \leq \sum_{h\in N\setminus N_q:h\neq i} x_{h,i} \qquad \forall p, q \in K, i \in N_p : p \neq q \tag{6}$$

$$\sum_{j\in N_p:j\neq i} y_{i,j} \leq \sum_{h\in N:h\neq i} y_{h,i} \qquad \forall p \in K, i \in N_p \tag{7}$$

$$y_{i,j} + \sum_{h\in N\setminus N_p}\sum_{k\in N_p} x_{h,k} \leq 1 \qquad \forall p \in K, i, j \in N_p : i \neq j \tag{8}$$

$$\sum_{j\in N_p} w_{j,h} = 1 \qquad \forall p \in K, h \in N_p \tag{9}$$

$$w_{j,h} \leq \sum_{i\in N:i\neq j} (x_{i,j} + y_{i,j}) \qquad \forall p \in K, j, h \in N_p \tag{10}$$

$$\sum_{j\in N:j\neq 1} f_{1,j} = |N| - 1 \tag{11}$$

$$\sum_{i\in N:i\neq j} f_{i,j} = 1 + \sum_{h\in N:h\neq j} f_{j,h} \qquad \forall j \in N, j \neq 1 \tag{12}$$

$$f_{i,j} \leq (|N| - 1)x_{i,j} + (|N_p| - 1)y_{i,j} + w_{i,j} \qquad \forall p \in K, i, j \in N_p : i \neq j \tag{13}$$

$$f_{i,j} \leq (|N| - 1)x_{i,j} + |N_p|y_{i,j} \qquad \forall p \in K, i \in N \setminus N_p, j \in N_p \tag{14}$$

$$x_{i,j}, y_{i,j} \in \{0, 1\} \qquad \forall i, j \in N : i \neq j \tag{15}$$

$$w_{ij} \in \{0, 1\} \qquad \forall i, j \in N_p,\ p \in K \tag{16}$$

$$f_{i,j} \geq 0 \qquad \forall i, j \in N : i \neq j \tag{17}$$

A função objetivo (1) é composta pela soma de três custos: primário (tours), secundário (caminhos) e terciário (atribuições). O tour deve conter o nó depósito conforme indicado pela Equação (2). As Equações (3) estabelecem o balanço de fluxo nodal, enquanto (4) determinam

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

que cada cluster $p$ deve receber um arco primário ou secundário. A conexão de cada nó $i$ dentro de um cluster $p$ é garantida pelas Equações (5). As restrições (6) impõem que, se dois clusters diferentes $(p, q)$ forem conectados por um arco secundário $y_{i,j} = 1$, então o nó $i$ pertence ao tour primário. Já as restrições (7) garantem que, se um arco secundário deixar o nó $i$, então algum outro arco secundário deve entrar no nó $i$. As restrições (8) proíbem arcos secundários dentro de clusters primários. Todas as atribuições são garantidas em (9), enquanto (10) garantem que cada nó seja atribuído a um nó envolvido em uma rota primária ou secundária. Consequentemente, como $r_{i,i} = 0$ e o modelo naturalmente favorece a autoatribuição para minimizar os custos, os nós incluídos no percurso principal ou em caminhos secundários serão atribuídos a si mesmos.

As restrições (11) – (14) previnem o aparecimento de subtours usando variáveis contínuas $f_{i,j}$ que representam o fluxo entre os nós $i$ e $j$. Essas restrições estabelecem unidades de fluxo que devem partir do depósito garantindo a conectividade da rede. Sua formulação é baseada no trabalho de Gavish e Graves [1978]. Em detalhe, a Equação (11) impõe que $|N| - 1$ unidades de fluxo devem sair do depósito. As Equações (12) impõem um balanço onde, para cada nó exceto o depósito, o fluxo que entra em $j$ deve ser exatamente uma unidade maior do que o fluxo que sai. As restrições (13) permitem o fluxo no arco $(i, j)$ dentro do cluster $p$, conforme sua classificação. Similarmente, (14) permitem o fluxo entre clusters com base na classificação do arco. Por fim, (15)–(17) definem o domínio das variáveis.

O modelo representa adequadamente o PRMGDN e previne a formação de subtours. No entanto, sua formulação baseada em fluxos impõe um alto custo computacional, devido à criação de $\mathcal{O}(|N|^2)$ variáveis de fluxo e ao uso de constantes elevadas em algumas restrições, o que compromete a qualidade da relaxação linear. Dessa forma, a formulação completa torna-se viável apenas para instâncias de pequeno porte, evidenciando a importância do uso de métodos de decomposição.

## 3. Estrutura de Decomposição de Benders

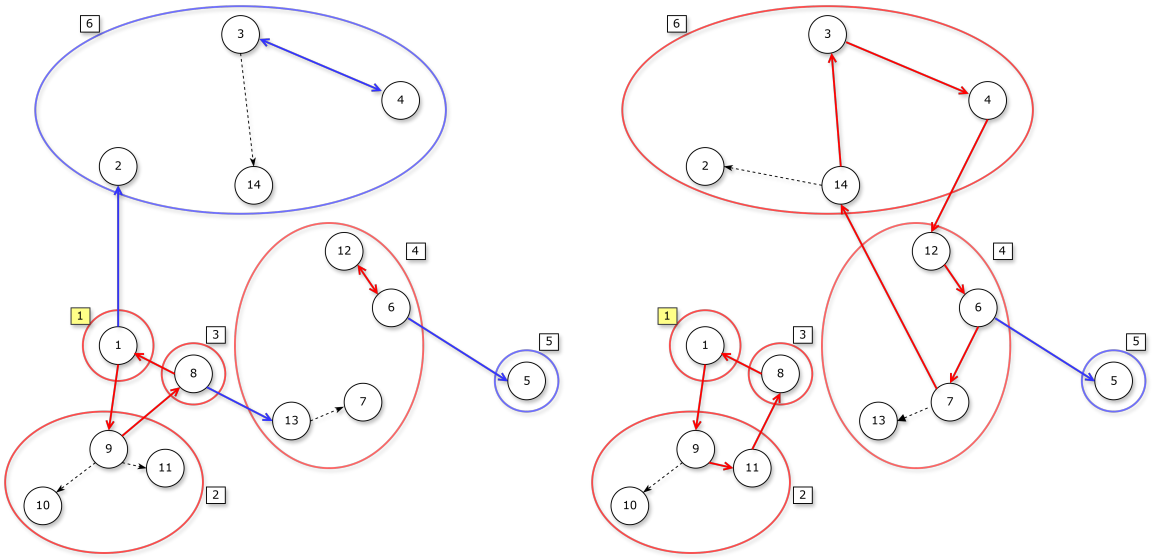
A decomposição de Benders é uma técnica clássica e amplamente aplicada para resolver problemas de otimização combinatória de larga escala, em especial modelos PLIM [Benders, 1962]. A ideia central é decompor o problema original em dois componentes interligados: um Problema Mestre (PM) e um ou mais Subproblemas (SPs). O PM normalmente retém as variáveis inteiras de decisão, frequentemente denominadas variáveis complicadoras, enquanto as variáveis contínuas são projetadas e tratadas no SP. A cada iteração, o PM é resolvido para fornecer valores às variáveis inteiras, que são então fixadas no SP, o qual se torna um problema de programação linear mais simples e resolvido de forma eficiente. Através da teoria da dualidade forte, a solução do SP fornece planos de corte para o PM. Esse processo iterativo continua até que se atinja a convergência.

No contexto desta pesquisa, o PM é construído usando a função objetivo (1) junto das restrições (2)–(10), (15) e (16), sendo responsável por determinar a estrutura do tour principal, os caminhos associados e as decisões de atribuição, com o objetivo de minimizar o custo total. No entanto, devido à complexidade inerente do problema, as soluções geradas pelo PM podem conter subtours.

Exemplos de subtours que podem surgir em soluções candidatas estão ilustrados na Figura 2. Em particular, a Figura 2a mostra uma solução composta por três subtours: um inter-cluster envolvendo os Clusters 1, 2 e 3, e dois intra-cluster: um subtour primário conectando os Nós 12 e 6, e um subtour secundário envolvendo os Nós 3 e 4. Por outro lado, a Figura 2b apresenta uma solução com dois subtours primários: o primeiro envolvendo os Clusters 1, 2 e 3, e o segundo abrangendo os Clusters 4 e 6.

Para lidar com isso, um único SP composto por (11)–(14) e (17) é formulado para validar a solução do PM, identificando tais subtours. Quando detectados, o SP gera cortes de viabilidade que são adicionados ao PM para eliminar a solução atual e parte do espaço de soluções correspondente.

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br




(a) Solução com 3 subtours

(b) Solução com 2 subtours

Figura 2: Exemplo de soluções do PRMGDN contendo subtours

Por meio desse processo iterativo, o algoritmo refina progressivamente a região viável do PM ao excluir soluções com subtours, ao mesmo tempo em que guia a busca em direção à otimalidade. A convergência é alcançada quando o PM fornece uma solução inteira que esteja livre de subtours e ao menor custo possível. Essa estratégia baseada em decomposição permite a resolução eficiente do PRMGDN ao separar as variáveis de fluxo em um SP reduzido e tratável, mantendo no PM apenas variáveis binárias, sem comprometer a otimalidade global.

Entretanto, uma implementação direta da decomposição de Benders frequentemente apresenta desvantagens, como iterações demoradas, cortes de viabilidade fracos, ineficiência nas iterações iniciais e convergência lenta quando próximo à otimalidade [Rahmaniani et al., 2017]. Para superar esses desafios e melhorar o desempenho geral, é essencial incorporar uma combinação de estratégias de aceleração. Neste estudo, adota-se um conjunto de melhorias bem fundamentadas na literatura devido ao seu bom desempenho. Nesse sentido, as seções a seguir têm como objetivo descrever a estrutura tradicional da decomposição de Benders e detalhar as técnicas de aceleração desenvolvidas.

### 3.1. Decomposição de Benders Tradicional

Dada uma solução $(\hat{x}, \hat{y}, \hat{w})$ obtida na iteração atual do PM, o SP serve como verificação de viabilidade, verificando a presença de subtours na solução. Associando as variáveis duais $\alpha, \beta, \gamma, \pi$ às Restrições (11) – (14), respectivamente, o Subproblema Dual de Benders (SPD) é definido pelas por (18) – (21).

$$\begin{aligned}\max_{\alpha,\beta,\gamma,\pi} \quad & \Big(|N|-1\Big)\alpha + \sum_{j\in N, j\neq 1} \beta_j + \\ & \sum_{p\in K} \sum_{i,j\in N_p, i\neq j} \Big((|N|-1)\hat{x}_{i,j} + (|N_p|-1)\hat{y}_{i,j} + \hat{w}_{i,j}\Big)\gamma_{p,i,j} + \\ & \sum_{p\in K} \sum_{i\in N\setminus N_p, j\in N_p} \Big((|N|-1)\hat{x}_{i,j} + |N_p|\hat{y}_{i,j}\Big)\pi_{p,i,j}\end{aligned} \tag{18}$$

DOI: 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

Sujeito a:

$$\alpha + \beta_i - \beta_j - \sum_{p \in K} \gamma_{p,i,j} - \sum_{p \in K} \pi_{p,i,j} \leq 0 \qquad \forall i, j \in N, i \neq j \tag{19}$$

$$\gamma_{p,i,j}, \pi_{p,i,j} \geq 0 \qquad \forall p \in K, i, j \in N, i \neq j \tag{20}$$

$$\alpha, \beta_j \in \mathbb{R} \qquad \forall j \in N \tag{21}$$

Seja $D$ o poliedro definido pelo SPD, e seja $\mathcal{R}_D$ o conjunto de seus raios extremos. Com base nisso, o PM é formulado da seguinte forma:

$$\min_{x,y,z,f} \quad \sum_{i \in N} \sum_{j \in N: j \neq i} c_{i,j} x_{i,j} + \sum_{i \in N} \sum_{j \in N: j \neq i} d_{i,j} y_{i,j} + \sum_{p \in K: p \neq 1} \sum_{i \in N_p} \sum_{j \in N_p} r_{i,j} w_{i,j} \tag{22}$$

Sujeito a:

$$\begin{aligned} &\textbf{Eqs.}(2) - (17) \\ &\Big(|N| - 1\Big)\alpha + \sum_{j \in N, j \neq 1} \beta_j + \\ &\sum_{p \in K} \sum_{i,j \in N_p, i \neq j} \Big((|N| - 1)x_{i,j} + (|N_p| - 1)y_{i,j} + w_{i,j}\Big)\gamma_{p,i,j} + \\ &\sum_{p \in K} \sum_{i \in N \setminus N_p, j \in N_p} \Big((|N| - 1)x_{i,j} + |N_p| y_{i,j}\Big)\pi_{p,i,j} \geq 0 \qquad \forall (\alpha, \beta, \gamma, \pi) \in \mathcal{R}_D \end{aligned} \tag{23}$$

No entanto, é bem conhecido que o conjunto $\mathcal{R}_D$ pode ser extremamente grande, frequentemente até maior do que o número de restrições do problema original. Assim, a ideia central da estrutura de Benders é gerar os cortes de forma incremental, ao invés de todos de uma vez. Cada corte, representado pela Inequação (23), é adicionado iterativamente. Na prática, a solução ótima geralmente é encontrada após gerar apenas um subconjunto pequeno dos raios extremos de $\mathcal{R}_D$, evitando assim a necessidade de explorar o conjunto completo.

### 3.1.1. Abordagem em Duas Fases

A estratégia de decomposição de Benders em duas fases, originalmente proposta por McDaniel e Devine [1977], aproveita o fato de que soluções obtidas a partir da relaxação linear do PM ainda podem gerar cortes válidos e úteis. A abordagem começa com a primeira fase (linear), resolvendo o PM sem restrições de integralidade, permitindo resoluções rápidas e eficientes. Uma vez que nenhum novo corte é encontrado no espaço relaxado, a segunda fase (inteira) é iniciada, onde as restrições de integralidade são reintroduzidas. Neste ponto, o PM já incorporou uma coleção de cortes, que servem para guiar o otimizador de forma mais eficaz ao longo da árvore de *Branch-and-Bound* (B&B), melhorando os limites iniciais e acelerando a convergência na fase inteira subsequente.

### 3.1.2. Árvore Única

No esquema tradicional de decomposição de Benders, o PM é resolvido até a otimalidade a cada iteração. Essa abordagem exige a reconstrução de uma nova árvore de B&B a cada iteração, frequentemente resultando em esforço computacional redundante ao revisitar soluções candidatas já eliminadas anteriormente. Uma estratégia alternativa consiste em manter uma árvore B&B unificada

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

durante todo o processo de solução e gerar cortes válidos para as soluções encontradas dentro da árvore, alcançando a mesma solução ótima [Rahmaniani et al., 2017]. Essa estratégia é comumente chamada de *Branch-and-Benders-Cut* (B&BC) na literatura de Benders e é adotada na implementação deste trabalho. Para aplicações adicionais de B&BC, remetemos o leitor aos trabalhos de Moreno et al. [2019] e Senna et al. [2024].

### 3.1.3. Normalização do Subproblema

Os cortes de viabilidade de Benders são normalmente derivados pela seleção de raios extremos do SPD. Contudo, como observado em trabalhos fundamentais como Benders [1962] e discutido por Fischetti et al. [2010], essa abordagem padrão possui uma limitação importante: ela gera um raio extremo arbitrário sem considerar a força ou profundidade do corte resultante. Como consequência, os cortes de viabilidade gerados podem ser fracos ou rasos, dificultando a convergência do algoritmo como um todo.

Para contornar essa limitação, técnicas de normalização surgiram como uma estratégia poderosa para melhorar a qualidade dos cortes. Em vez de otimizar sobre um cone ilimitado como na abordagem padrão, a normalização envolve resolver um programa linear sobre um poliedro limitado. Em outras palavras, ela adiciona uma condição de normalização que trunca o cone dual e o transforma em um politopo equivalente. Neste trabalho, adota-se a estratégia proposta por Fischetti et al. [2010], também implementada na versão automática do Benders do CPLEX [Bonami et al., 2020]. Incorporando esse conceito no algoritmo proposto, as variáveis de penalidade $z_1$ e $z_2$ são criadas, gerando um novo SP.

$$\min_{z_1,z_2,f} \quad z_1 + z_2 \tag{24}$$

Sujeito a:

$$\sum_{i\in N:i\neq j} f_{i,j} + z_1 = z_2 + 1 + \sum_{h\in N:h\neq j} f_{j,h} \qquad \forall j \in N, j \neq 0 \tag{25}$$

$$z_1, z_2 \geq 0 \tag{26}$$

**Eqs.** $(11) - (14), (17)$

A nova função objetivo (24) minimiza a soma das variáveis de penalidade, que são introduzidas nas Equações (12), resultando nas Equações (25). O domínio das novas variáveis é definido em (26). Essas variáveis de penalidade atuam como componentes de fluxo artificiais, garantindo que o SP permaneça viável mesmo quando as restrições originais forem violadas. Como consequência natural, a inclusão de duas variáveis leva à introdução de duas novas restrições na formulação dual, gerando um novo SPD.

$$\begin{aligned}\max_{\alpha,\beta,\gamma,\pi} \quad & \Big(|N|-1\Big)\alpha + \sum_{j\in N, j\neq 1} \beta_j + \\ & \sum_{p\in K} \sum_{i,j\in N_p, i\neq j} \Big((|N|-1)\hat{x}_{i,j} + (|N_p|-1)\hat{y}_{i,j} + \hat{w}_{i,j}\Big)\gamma_{p,i,j} + \\ & \sum_{p\in K} \sum_{i\in N\setminus N_p, j\in N_p} \Big((|N|-1)\hat{x}_{i,j} + |N_p|\hat{y}_{i,j}\Big)\pi_{p,i,j}\end{aligned} \tag{27}$$

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

Sujeito a:

$$\sum_{j \in N: j \neq 1} \beta_j \leq 1 \tag{28}$$

$$-\sum_{j \in N: j \neq 1} \beta_j \leq 1 \tag{29}$$

**Eqs.** (19) – (21)

A nova formulação dual se assemelha à original, com a principal diferença sendo a adição das Restrições (28) e (29). Juntas, as duas equivalem à Restrição (30).

$$\left| \sum_{j \in N: j \neq 1} \beta_j \right| \leq 1 \tag{30}$$

Com o SPD normalizado, os cortes de viabilidade continuam sendo gerados conforme a mesma estrutura da Inequação (23). No entanto, essa modificação permite a seleção de pontos extremos equivalentes a raios extremos, resultando em cortes mais fortes que eliminam soluções inviáveis de forma mais eficiente. Como resultado, a abordagem não apenas melhora a qualidade dos cortes, mas também acelera a convergência e aumenta a eficiência geral do processo de decomposição de Benders.

### 3.1.4. Desigualdades Válidas

Neste trabalho, um conjunto de Desigualdades Válidas (DVs) fortes é implementado com o objetivo simultâneo de reforçar a relaxação linear do PM e melhorar a qualidade das soluções inteiras iniciais. A introdução dessas DVs contribui para gerar limites inferiores iniciais mais apertados, reduzir o número total de iterações, melhorar a qualidade das soluções fracionárias e, por fim, promover uma exploração mais eficaz dos nós inteiros na árvore unificada de B&B.

As DVs propostas são apresentadas em (31) – (32). Em (31) são tratadas DVs intra-cluster, envolvendo variáveis de alocação, enquanto (32) são DVs inter-cluster.

$$x_{i,j} + x_{j,i} + y_{i,j} + y_{j,i} + w_{i,j} + w_{j,i} \leq 1 \quad \forall p \in K, i, j \in N_p : i < j \tag{31}$$

$$x_{i,j} + x_{j,i} + y_{i,j} + y_{j,i} \leq 1 \quad \forall p \in K, i \in N \setminus N_p, j \in N_p : i < j, 1 \notin N_p \tag{32}$$

Primeiramente, essas desigualdades reforçam a relaxação linear do PM ao eliminar parte das soluções fracionárias que atribuem múltiplos tipos de rota ao mesmo arco $(i, j)$ — por exemplo, casos em que tanto $x_{i,j} > 0$ quanto $y_{i,j} > 0$ ocorrem simultaneamente. Além disso, ao considerar ambas as direções de um arco dado, ou seja, $(i, j)$ e $(j, i)$, as desigualdades se tornam ainda mais restritivas.

Ademais, para soluções inteiras, essas mesmas desigualdades inviabilizam a formação de subtours de tamanho dois (por exemplo, os subtours apresentados na Fig. 2a). Para preservar a generalidade, o Nó 1 (depósito) é excluído das desigualdades em (32), já que uma solução ótima pode envolve-lo em um tour de comprimento dois. Para (31), tal exclusão não é necessária, uma vez que o cluster do depósito contém apenas ele mesmo. Por fim, a condição $i < j$ é imposta para evitar a geração de restrições redundantes, mantendo assim um modelo mais compacto e eficiente.

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

## 4. Implementação e Experimentos Computacionais

Esta seção apresenta os resultados numéricos obtidos com o algoritmo de decomposição de Benders proposto. Para fins de comparação, consideramos duas abordagens de referência: o algoritmo *branch-and-cut* (B&C) desenvolvido por Obreque et al. [2024] e a funcionalidade integrada de decomposição de Benders disponível na ferramenta de otimização CPLEX. Especificamente, implementamos a abordagem B&C proposta por Obreque et al. [2024], a qual utiliza uma formulação alternativa junto de restrições clássicas de eliminação de subtours e apresentou os melhores resultados em seus experimentos. De modo a assegurar uma comparação justa entre os métodos, adotou-se o mesmo limite de tempo utilizado no estudo original, fixado em 4 horas (14400 segundos).

A formulação matemática foi implementada na linguagem de programação Julia 1.9.1, utilizando o pacote de modelagem JuMP 1.25.0 e o otimizador Gurobi® 11.0.3. Para as comparações com a decomposição automática de Benders, foi utilizado o CPLEX® na versão 22.1.2. Todos os experimentos foram conduzidos em uma máquina com processador Intel® i7-10700K de 3.8GHz e 64GB de memória RAM, utilizando apenas uma thread.

Durante a exploração da árvore B&B unificada, são gerados cortes em todas as soluções encontradas e incorporados como *lazy constraints*. O problema mestre é resolvido, por padrão, utilizando o método simplex dual devido a sua sinergia com cortes de viabilidade. Já para a resolução dos subproblemas, o método de pontos interiores foi preferido devido ao alto grau de degenerescência observado, onde métodos baseados em simplex tendem a apresentar desempenho inferior. Por fim, a tolerância para separação de cortes foi fixada em $\epsilon = 10^{-4}$.

### 4.1. Seleção de Instâncias

Os experimentos neste artigo foram conduzidos sobre um conjunto de 32 instâncias também utilizadas por Obreque et al. [2024] que pertencem ao repositório aberto da GTSP-Lib. Essas instâncias foram originalmente introduzidas por Fischetti et al. [1997] aplicando um procedimento de clusterização sobre instâncias clássicas da TSPLIB. Foram selecionados problemas variando de 100 a 417 nós e 20 a 84 clusters. Por exemplo, na instância `50gr96`, 96 nós estão agrupados em 50 clusters. Todas as instâncias são euclidianas e simétricas, o que significa que a distância entre quaisquer dois nós $i$ e $j$ satisfaz $D_{i,j} = D_{j,i}$. Foram adotados custos diretamente proporcionais às distâncias, de modo que $D_{i,j} = c_{i,j} = d_{i,j} = r_{i,j}$.

### 4.2. Resultados

A Tabela 1 apresenta uma comparação detalhada entre três abordagens distintas para a resolução do PRMGDN: o algoritmo de B&C proposto por Obreque et al. [2024], a decomposição de Benders automática do CPLEX e o algoritmo de Benders proposto neste trabalho. Os resultados são reportados em termos de tempo de execução (em segundos) e gap de integralidade alcançado, considerando um limite de tempo de 4 horas (14400 segundos), conforme adotado também no trabalho original.

Os dados indicam um desempenho significativamente superior do algoritmo proposto em relação às abordagens de referência. Enquanto o método original não alcançou a solução ótima em todas as instâncias, apresentando um gap médio de 3.10% e levando em média 1766.46 segundos por instância, o algoritmo proposto atingiu otimalidade em todas as 30 instâncias com um tempo médio de apenas 395.69 segundos — uma redução superior a 77% no tempo de resolução médio.

Em relação à decomposição automática do CPLEX, os ganhos também foram substanciais. Embora o CPLEX tenha apresentado um gap médio baixo (0.08%), não obteve otimalidade em todas as instâncias e registrou um tempo médio de 1603,74 segundos, cerca de quatro vezes maior do que o tempo médio do algoritmo proposto. Além disso, o desempenho do CPLEX variou de forma expressiva entre instâncias, com tempos particularmente elevados em casos de porte médio,

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

como a `40kroB200`. O algoritmo deste trabalho, por outro lado, demonstrou robustez e consistência, mantendo tempos controlados mesmo em instâncias mais exigentes, como a `64lin318`, o que evidencia a eficácia das técnicas de aceleração adotadas.

Em resumo, a análise dos resultados evidencia que o algoritmo de decomposição de Benders proposto não apenas supera amplamente o trabalho original em termos de qualidade e tempo de solução, como também se mostra substancialmente mais eficiente que a implementação automatizada do CPLEX. Esses resultados reforçam o potencial da abordagem desenvolvida como uma alternativa escalável e robusta para a resolução do PRMGDN.

| Instância | B&C | | Benders CPLEX | | Benders Proposto | |
|---|---|---|---|---|---|---|
| | Tempo(s) | GAP(%) | Tempo(s) | GAP(%) | Tempo(s) | GAP(%) |
| 20kroA100 | 31.85 | 0.00 | 28.07 | 0.00 | 17.71 | 0.00 |
| 20kroB100 | 42.44 | 0.00 | 22.96 | 0.00 | 24.63 | 0.00 |
| 20kroC100 | 38.48 | 0.00 | 20.00 | 0.00 | 13.51 | 0.00 |
| 20kroD100 | 46.18 | 0.00 | 33.45 | 0.00 | 29.22 | 0.00 |
| 20kroE100 | 21.08 | 0.00 | 9.46 | 0.00 | 15.70 | 0.00 |
| 20rat99 | 35.65 | 0.00 | 18.91 | 0.00 | 24.31 | 0.00 |
| 20rd100 | 28.35 | 0.00 | 21.41 | 0.00 | 20.61 | 0.00 |
| 21eil101 | 78.29 | 0.00 | 68.91 | 0.00 | 24.31 | 0.00 |
| 21lin105 | 45.63 | 0.00 | 39.69 | 0.00 | 26.76 | 0.00 |
| 22pr107 | 14.75 | 0.00 | 5.21 | 0.00 | 10.15 | 0.00 |
| 25pr124 | 51.04 | 0.00 | 24.02 | 0.00 | 40.65 | 0.00 |
| 26bier127 | 288.04 | 0.00 | 268.96 | 0.00 | 197.88 | 0.00 |
| 28gr137 | 55.34 | 0.00 | 17.27 | 0.00 | 37.91 | 0.00 |
| 28pr136 | 176.31 | 0.00 | 79.44 | 0.00 | 48.23 | 0.00 |
| 29pr144 | 96.14 | 0.00 | 35.95 | 0.00 | 40.19 | 0.00 |
| 30kroA150 | 39.45 | 0.00 | 120.30 | 0.00 | 59.61 | 0.00 |
| 30kroB150 | 46.71 | 0.00 | 94.83 | 0.00 | 87.70 | 0.00 |
| 31pr152 | 327.36 | 0.00 | 254.86 | 0.00 | 151.72 | 0.00 |
| 32u159 | 79.87 | 0.00 | 158.65 | 0.00 | 102.83 | 0.00 |
| 39rat195 | 275.76 | 0.00 | 310.00 | 0.00 | 167.33 | 0.00 |
| 40d198 | 401.92 | 0.00 | 185.42 | 0.00 | 243.53 | 0.00 |
| 40kroA200 | 298.49 | 0.00 | 1346.91 | 0.00 | 147.64 | 0.00 |
| 40kroB200 | 654.92 | 0.00 | 6739.94 | 0.00 | 548.05 | 0.00 |
| 41gr202 | 2848.49 | 0.00 | 873.63 | 0.00 | 421.05 | 0.00 |
| 45ts225 | 1027.28 | 0.00 | 2047.06 | 0.00 | 1322.64 | 0.00 |
| 46gr229 | 14400.02 | 0.50 | 1516.95 | 0.00 | 513.05 | 0.00 |
| 46pr226 | 540.72 | 0.00 | 201.87 | 0.00 | 258.17 | 0.00 |
| 53gil262 | 966.65 | 0.00 | 1126.00 | 0.00 | 970.13 | 0.00 |
| 53pr264 | 14400.02 | 7.10 | 2880.65 | 0.00 | 622.54 | 0.00 |
| 60pr299 | 2330.41 | 0.00 | 14400.07 | 1.50 | 1945.52 | 0.00 |
| 64lin318 | 2438.96 | 0.00 | 14400.07 | 1.20 | 2350.56 | 0.00 |
| 84fl417 | 14400.28 | 91.60 | 3619.92 | 0.00 | 2174.34 | 0.00 |
| Média | 1766.46 | 3.10 | 1603.74 | 0.08 | 395.69 | 0.00 |

Tabela 1: Resultados computacionais

## 5. Conclusão

Este trabalho propôs um algoritmo baseado em decomposição de Benders para a resolução do PRMGDN, incorporando técnicas modernas de aceleração, como a abordagem em duas fases, utilização de uma única árvore de busca, subproblemas normalizados e a adição de desigualdades válidas. O algoritmo demonstrou desempenho competitivo, alcançando a solução ótima em todas as 32 instâncias selecionadas da GTSP-Lib. Os resultados superaram tanto os obtidos no artigo original quanto a versão automática da decomposição de Benders implementada no otimizador CPLEX.

Como continuidade deste trabalho, estão sendo investigadas novas estratégias de aceleração, incluindo métodos de estabilização, remoção de cortes inativos e seleção de cortes fracionários

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br

durante a execução da árvore de B&B. Essas melhorias visam aumentar ainda mais a eficiência e escalabilidade do algoritmo proposto, possibilitando sua aplicação a instâncias de maior porte e complexidade.

**Referências**

Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252.

Bonami, P., Salvagnin, D., e Tramontani, A. (2020). Implementing automatic benders decomposition in a modern mip solver. In Bienstock, D. e Zambelli, G., editors, *Integer Programming and Combinatorial Optimization. IPCO 2020*, volume 12125 of *Lecture Notes in Computer Science*, p. 78–90. Springer, Cham.

Costa, A. M. (2005). A survey on benders decomposition applied to fixed-charge network design problems. *Computers & Operations Research*, 32(6):1429–1450. ISSN 0305-0548.

Fischetti, M., Salvagnin, D., e Zanette, A. (2010). A note on the selection of benders' cuts. *Mathematical Programming*, 124:175–182.

Fischetti, M., Salazar González, J. J., e Toth, P. (1997). A branch-and-cut algorithm for the symmetric generalized traveling salesman problem. *Operations Research*, 45(3):378–394.

Gavish, B. e Graves, S. (1978). The traveling salesman problem and related problems. *Massachusetts Institute of Technology, Operations Research Center*.

McDaniel, D. e Devine, M. (1977). A modified benders' partitioning algorithm for mixed integer programming. *Management Science*, 24:312–319.

Moreno, A., Munari, P., e Alem, D. (2019). A branch-and-benders-cut algorithm for the crew scheduling and routing problem in road restoration. *European Journal of Operational Research*, 275(1):16–34. ISSN 0377-2217. URL https://www.sciencedirect.com/science/article/pii/S0377221718309299.

Obreque, C., Paredes-Belmar, G., Miranda, P. A., Campuzano, G., e Gutiérrez-Jarpa, G. (2020). The generalized median tour problem: Modeling, solving and an application. *IEEE Access*, 8: 178097–178107.

Obreque, C., Paredes-Belmar, G., Miranda-Gonzalez, P. A., Campuzano, G., e Gutiérrez-Jarpa, G. (2024). Modeling and solving the two-level generalized median tour problem. *Computers & Operations Research*, 164:106558. ISSN 0305-0548.

Rahmaniani, R., Crainic, T. G., Gendreau, M., e Rei, W. (2017). The benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817. ISSN 0377-2217.

Senna, F., Coelho, L. C., Morabito, R., e Munari, P. (2024). An exact method for a last-mile delivery routing problem with multiple deliverymen. *European Journal of Operational Research*, 317(2): 550–562. ISSN 0377-2217.

**DOI:** 10.59254/sbpo-2025-212503
https://proceedings.science/p/212503?lang=pt-br