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

BRKGA híbrido com Reinforcement Learning aplicado a alocação de estações de compartilhamento de bicicletas

Gabriel Souto, Caio Davi, Luidi Simonetti, Pedro Henrique González
Universidade Federal do Rio de Janeiro
R. Antônio Barros de Castro, 119 - Cidade Universitária, Rio de Janeiro - RJ, 21941-853
{soutog,luidi,pegonzalez}@cos.ufrj.br,
caiodavimonteiro.20231@poli.ufrj.br

RESUMO

Problemas de otimização combinatória surgem em diversas aplicações reais e exigem estratégias eficientes para tomada de decisão. Entre eles, o Problema da Mochila com Penalidades (KPF) modela conflitos e incompatibilidades gerando custos adicionais quando itens mutuamente excludentes são selecionados simultaneamente. Este estudo revisita avanços recentes na literatura e enfatiza a eficácia de métodos híbridos que mesclam técnicas de busca local e aprendizado. Formalizamos o KPF por meio de uma modelagem matemática capaz de incorporar penalidades de modo eficiente. Propomos o HDM-ILS-VND, um algoritmo que integra Iterated Local Search(ILS), Variable Neighborhood Descent(VND), mineração de padrões frequentes e Local Branching. A abordagem inicia-se com uma construção gulosa e alterna entre fases de intensificação e diversificação, aprendendo padrões de excelência a partir das melhores soluções geradas. Experimentos conduzidos em instâncias benchmark mostram que o HDM-ILS-VND atinge soluções de alta qualidade, frequentemente superando métodos consagrados na literatura, evidenciando o potencial de sua estratégia híbrida.

PALAVRAS CHAVE. Problema da Mochila com Conjuntos de Penalidades, BRKGA, *Reinforcement Learning*, Cidades Inteligentes.

MH - Metaheurísticas, OA – Outras Aplicações em PO

ABSTRACT

The optimization of bike-sharing station placements in urban areas is an essential issue in the context of smart cities. One way to model this problem is through the Knapsack Problem with Forfeit Sets (KPFS). In this model, each bike-sharing station is treated as an item with specific costs and associated profits. Placing multiple stations close can lead to underutilization and increased maintenance expenses, representing forfeit sets where penalties are incurred if predefined allowances are exceeded. To address this problem in smart cities optimization, we propose a hybrid metaheuristic that combines a Biased Random-Key Genetic Algorithm (BRKGA) with Q-learning and Local Branching, referred to as QVND-HBRKGA. This approach integrates a reinforcement learning-based variable neighborhood descent to explore neighborhood solutions. Computational experiments demonstrate that QVND-HBRKGA effectively balances resource allocation and minimizes penalties, outperforming existing methods in the literature.

KEYWORDS. Knapsack Problem with Forfeits Sets, BRKGA, *Reinforcement Learning*, Smart Cities.

MH - Metaheuristics, OA – Other Applications in OR

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

1. Introdução

No contexto de cidades inteligentes, a alocação eficiente de recursos e a gestão de conflitos são fundamentais para garantir sustentabilidade, distribuição equitativa de serviços e uso ideal dos sistemas urbanos Batty et al. [2012]. Conflitos surgem, por exemplo, quando infraestruturas se sobrepõem, como múltiplas estações de bicicletas compartilhadas instaladas muito próximas, levando à subutilização em determinadas regiões. De forma análoga, iniciativas de expansão de espaços verdes podem competir por orçamento e espaço físico com projetos de transporte público Ma et al. [2016]; Yin e Liu [2020]; Ma et al. [2018].

Essa tensão entre decisões sobrepostas e uso de dados inspirou estudos recentes, como o conduzido em 2024 pela Universidade de Western Ontario, que foca em cidades canadenses e propõe uma metodologia para identificar locais ótimos para estações de compartilhamento de bicicletas, evidenciando a necessidade de decisões baseadas em dados e análises empíricas [ElNawawy et al., 2024].

Neste contexto, modelamos o problema como uma variação do clássico problema da mochila: o Knapsack Problem with Forfeits Sets (KPFS) [D'Ambrosio et al., 2023]. Nele, cada possível localização de estação é tratada como um item, associado a custos (instalação, manutenção) e benefícios esperados (acesso, redução de tráfego). A capacidade da "mochila" representa o orçamento disponível. O diferencial do KPFS está nos conjuntos de penalidade — os *forfeits sets* — que agrupam locais redundantes ou próximos, limitando o número de escolhas simultâneas e penalizando sobreposições excessivas Bruck et al. [2025]; Chen et al. [2024].

O KPFS tem inúmeras aplicações em sistemas urbanos, como planejamento de transporte (bicicletas e estações de recarga para veículos elétricos), gerenciamento de resíduos (locais de centros de reciclagem) e logística urbana (seleção de itens para transporte com custos extras de manuseio) [Cerulli et al., 2020; Capobianco et al., 2022]. No entanto, sua resolução eficiente ainda impõe desafios computacionais.

Para enfrentar esse desafio, propomos um algoritmo híbrido que combina um *Biased Random Key Genetic Algorithm* (BRKGA) Ericsson et al. [2002]; Gonçalves e Resende [2011]; Goncalves e Resende [2004] com uma técnica de *Reinforcement Learning* conhecida como *Q-Learning* Watkins e Dayan [1992]; Sutton e Barto [2018]; Kumar et al. [2020]; Zhang et al. [2022]; Song et al. [2023]. Complementando essa abordagem com uma fase de *Local Branching*, voltada à intensificação das melhores soluções encontradas Fischetti e Lodi [2003]. Esse conjunto de técnicas permite equilibrar exploração e refinamento de soluções de maneira dinâmica, aumentando a qualidade das decisões e a eficiência computacional.

A estrutura deste artigo está organizada da seguinte forma: a Seção 2 apresenta os trabalhos relacionados; a Seção 3 detalha o KPFS e sua formulação matemática; a Seção 4 descreve o método proposto e seus componentes; em seguida, discutimos os experimentos computacionais e seus resultados na Seção 5; por fim, a Seção 6 sintetiza as principais conclusões e propõe direções para pesquisas futuras.

2. Trabalhos Relacionados

O Knapsack Problem with Forfeits Sets (KPFS) tem se mostrado uma abordagem promissora para lidar com desafios de alocação em cidades inteligentes, como a instalação eficiente de estações de bicicletas compartilhadas. Embora sua aplicação nesse contexto ainda seja limitada, estudos correlatos abordam estratégias complementares, como o reequilíbrio dinâmico de frotas e incentivos para usuários, visando reduzir custos operacionais Chiariotti et al. [2020], além da maximização da cobertura populacional com número restrito de estações Hrabec et al..

O KPFS é uma generalização de variantes tradicionais do problema da mochila, como o Knapsack Problem with Conflicts (KPCG) Pferschy e Schauer [2009]; Li et al. [2021]; Carrabs et al.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

[2021] e o Knapsack Problem with Penalties (KPF) Cerulli et al. [2020]; Capobianco et al. [2022]; Souto et al. [2024], que lidam com conflitos binários e penalidades entre pares de itens. A introdução dos *forfeits sets* amplia esse escopo, permitindo modelar conflitos entre grupos de itens, o que o torna especialmente adequado a contextos urbanos com sobreposição de localização ou função.

O trabalho seminal de D'Ambrosio et al. [2023] formalizou o KPFS por meio de um modelo de programação inteira mista (MIP), propondo heurísticas eficientes como o *Carousel Greedy* e um algoritmo memético híbrido. A partir desse marco, novas abordagens foram desenvolvidas, como o *Biased Random Key Genetic Algorithm* (BRKGA) adaptado por Cerulli et al. [2024], que incorporou decodificadores especializados e estratégias de busca local, obtendo desempenho superior em instâncias de *benchmark*.

Mais recentemente, Jovanovic e Voß [2024] propuseram a *Matheuristic Fixed Set Search* (MFSS), que identifica padrões recorrentes em soluções de alta qualidade e os utiliza para guiar a busca, delegando a resolução de subproblemas a modelos exatos. Essa abordagem reforça a tendência de uso de técnicas híbridas para enfrentar a complexidade do KPFS, destacando seu potencial em cenários urbanos com restrições práticas e orçamentárias.

3. Definição do Problema

O *Knapsack Problem with Forfeits Sets* (KPFS), proposto por D'Ambrosio et al. [2023], estende o Problema da Mochila clássico ao incorporar penalidades por seleção excessiva de itens pertencentes a subconjuntos específicos.

Seja $X = \{1, 2, ..., n\}$ o conjunto de itens, cada um com lucro $p_j > 0$ e peso $w_j > 0$. A capacidade da mochila é $b > 0$. Define-se um conjunto $F = \{F_i\}_{i=1}^{l}$, onde cada $F_i \subseteq X$ com $|F_i| \geq 2$, está associado a uma penalidade $d_i > 0$ e um parâmetro de tolerância h_i, com $0 \leq h_i \leq |F_i|$. Para uma solução $S \subseteq X$, se $n_i^S = |F_i \cap S| > h_i$, aplica-se uma penalidade proporcional ao excesso. Um limite global $k \geq 0$ restringe o total de violações.

A formulação matemática do KPFS é dada por:

$$\max \quad \sum_{j}^{n} p_j x_j - \sum_{i}^{l} d_i v_i \tag{1}$$

$$\sum_{j}^{n} w_j x_j \leq b \tag{2}$$

$$\sum_{i}^{l} v_i \leq k \tag{3}$$

$$\sum_{j \in F_i} x_j - v_i \leq h_i \qquad \forall i = 1, ..., l \tag{4}$$

$$x_j \in \{0, 1\} \qquad \forall j = 1, ..., n \tag{5}$$

$$v_i \in \{0, ..., |F_i| - h_i\} \; \forall i = 1, ..., l \tag{6}$$

A função objetivo (1) busca maximizar o lucro total, subtraindo eventuais penalidades. A Restrição (2) assegura que a capacidade da mochila não seja excedida. A Restrição (3) limita o número total de violações de tolerância. A Restrição (4) regula, para cada conjunto de penalidade, o número de itens selecionados acima do limite permitido h_i. As variáveis x_j indicam a inclusão dos itens na mochila, enquanto v_i representam o número de violações relevantes em cada conjunto penalizado.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

4. Metodologia Proposta

Nesta seção, propomos uma abordagem híbrida para resolução do KPFS, integrando componentes heurísticos e exatos. Inicialmente, o *framework* principal baseado no *Biased Random Key Genetic Algorithm (BRKGA)* [Gonçalves e Resende, 2011], descrito na Seção 4.1, que funciona como mecanismo de controle populacional e diversificação da busca. Em seguida, falamos da Heurística Construtiva, responsável por gerar soluções iniciais (Seção 4.2); chamada dentro do decoder do BRKGA (Seção 4.3) uma fase de Busca Local que utiliza uma estratégia de VND, guiada por um *Q-Learning*, de modo a orientar dinamicamente a escolha de estruturas de vizinhança (QVND) (Seção 4.4); e uma etapa de Refinamento com *Local Branching*, voltada à intensificação de soluções promissoras (Seção 4.5).

A Figura 1 ilustra o fluxo geral da abordagem QVND-HBRKGA, destacando os principais estágios e a interação entre os componentes do método.

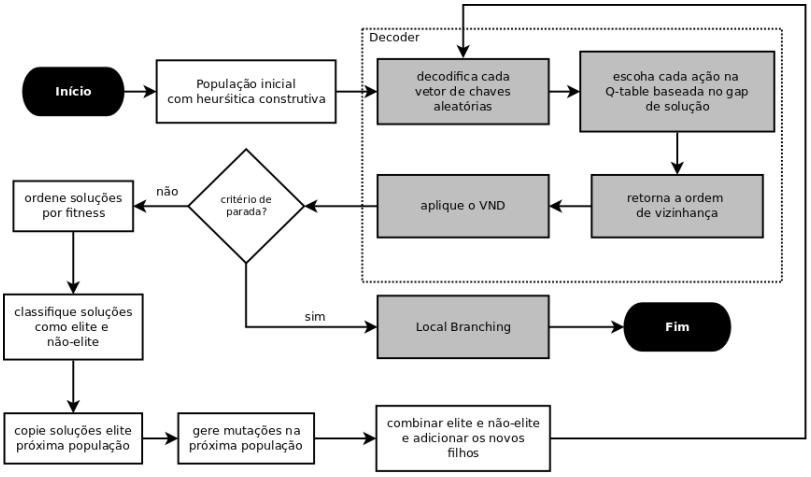


Figura 1: Fluxo geral do algoritmo híbrido QVND-HBRKGA

4.1. Biased Random Key Genetic Algorithm

O *Biased Random Key Genetic Algorithm* (BRKGA), introduzido por Gonçalves e Resende [2011], é uma metaheurística evolutiva que estende o modelo clássico do *Random Key Genetic Algorithm* (RKGA) [Bean, 1994], adicionando um viés no cruzamento para acelerar a convergência em espaços de busca complexos.

Neste modelo, cada solução é representada por um vetor de números reais no intervalo $[0, 1]$, chamados de *random keys*. Esses vetores são mapeados para soluções viáveis por um decodificador específico ao problema, responsável por interpretar as chaves e avaliar a qualidade da solução gerada. A população é organizada em dois grupos — elite e não-elite — com base na aptidão das soluções. As soluções do grupo elite são preservadas entre gerações, enquanto os novos indivíduos são gerados por cruzamento enviesado: um dos pais é sempre elite, e o outro é selecionado aleatoriamente do grupo não-elite, com probabilidade maior de herança dos genes do indivíduo elite. Mutações aleatórias são aplicadas a uma fração da população para manter diversidade e evitar estagnação prematura.

A flexibilidade do BRKGA tem sido validada em diversos domínios da otimização combinatória, incluindo problemas de localização de instalações, variantes do caixeiro-viajante, *job shop scheduling*, balanceamento de linhas de montagem e otimização de hiperparâmetros em aprendizado de máquina [Gonçalves e De Almeida, 2002; Brandao et al., 2015; Brandão et al., 2017; Gonzalez e Brandão, 2018; Souto et al., 2021; Morais et al., 2022; Japa et al., 2022; Souto et al., 2024]. Sua

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

estrutura modular permite integração com técnicas complementares, como as abordagens baseadas em aprendizado por reforço e busca local, tornando-o uma peça central no método proposto.

4.2. Heurística Construtiva

A fase construtiva é responsável por gerar a população inicial do BRKGA, fornecendo soluções viáveis que servem como ponto de partida para o processo evolutivo. Para isso, empregamos uma heurística semi-gulosa baseada na estratégia proposta em Souto et al. [2024], que combina seleção orientada por custo-benefício com um componente aleatório controlado.

O procedimento segue uma abordagem iterativa, onde, a cada passo, identifica-se o conjunto de itens candidatos — ou seja, aqueles que podem ser adicionados à mochila sem violar a capacidade total ou outras restrições do problema. Para cada item viável, calcula-se uma métrica de custo-benefício dada pela razão entre lucro e peso. A partir desse ranqueamento, constrói-se uma *Lista Restrita de Candidatos* (RCL), formada pelos itens com as melhores relações custo-benefício.

A seleção dos itens a serem inseridos é então realizada de forma aleatória dentro da RCL, balanceando exploração e direcionamento por meio de um parâmetro de aleatoriedade α. Esse processo se repete enquanto houver candidatos viáveis, garantindo que a solução inicial respeite as restrições do problema e mantenha diversidade na população.

4.3. Decoder

No BRKGA, o decodificador é responsável por mapear cromossomos (vetores de números reais no intervalo $[0, 1]$) em soluções viáveis para o problema em estudo. No caso do KPFS, esse processo utiliza uma abordagem gulosa, onde os itens são ordenados de acordo com seus alelos e inseridos na mochila respeitando a capacidade máxima.

A construção da solução é representada por um vetor binário de tamanho n, no qual cada elemento indica a inclusão (1) ou exclusão (0) de um item. O decodificador prioriza itens com menores valores de alelo, realizando a inserção até que novas adições violem as restrições do problema.

Após a construção inicial, o custo da solução gerada é comparado com a melhor solução conhecida S_{melhor}, calculando-se um *gap* relativo entre elas. Esse valor define o estado atual da busca, que é usado para selecionar, via *Q-Learning*, a ordem de aplicação das estruturas de vizinhança no processo de busca local (QVND). A Tabela Q, previamente treinada, orienta essa escolha, equilibrando exploração e intensificação.

Por fim, o cromossomo é sincronizado com a solução gerada, atualizando os alelos de acordo com os itens efetivamente selecionados. O procedimento do decodificador é resumido no Algoritmo 1.

Algorithm 1: Decoder

Data: Q_{table}, $cromossomo$, S_{melhor}
$S \leftarrow decode(cromossomo)$
$gap \leftarrow (1 - \frac{S_{\text{melhor}}}{getSolution(S)}) \times 100$
$estado \leftarrow getState(gap)$
$acao \leftarrow argmax(Q_{\text{table}}[estado])$
$ordem \leftarrow getNeighOrder(acao)$
$S' \leftarrow QVND(S, ordem)$
$syncChromosome(S')$
return S'

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

4.4. Variable Neighborhood Descent com Q-learning (QVND)

O *Variable Neighborhood Descent* (VND) [Hansen e Mladenovic, 2003] é uma estratégia de busca local que explora múltiplas estruturas de vizinhança para evitar ótimos locais prematuros. No KPFS, utilizamos três vizinhanças: (i) *Troca* de um item da solução por outro fora dela; (ii) *Inserção* de um novo item viável; e (iii) *Remoção* de um item selecionado.

Uma questão central no uso de VND é a escolha da ordem em que essas vizinhanças devem ser exploradas. Para guiar essa decisão de forma adaptativa, integramos uma política de *Reinforcement Learning*, baseada em Q-learning, ao processo de busca local, formando o componente QVND da metodologia proposta. Essa integração permite selecionar dinamicamente a estrutura de vizinhança mais promissora, considerando o desempenho atual da busca.

O agente de Q-learning utiliza como estado um indicador de desempenho da solução corrente, calculado pelo gap relativo em relação à melhor solução conhecida:

$$Q_{gap} = 1 - \frac{S_{\text{melhor}}}{S_{\text{atual}}}.$$

A partir desse estado, uma ação (isto é, a ordem de exploração das vizinhanças) é escolhida a partir da tabela Q previamente treinada, definindo a ordem de aplicação das vizinhanças no VND. Vale ressaltar também que para cada faixa de Q_{gap}, foi atribuído um valor inteiro, de modo que quanto mais próximo da melhor solução, maior o valor inteiro, isto é, a recompensa. Esse mecanismo permite balancear exploração e intensificação, promovendo melhorias na qualidade da solução gerada pelo decodificador do BRKGA.

Diversos estudos têm demonstrado a eficácia da combinação entre técnicas de *Reinforcement Learning* e métodos heurísticos para problemas de otimização combinatória [Hassan e Pillay, 2022; Song et al., 2023; Souto et al., 2024], reforçando a escolha dessa abordagem no presente trabalho.

Essa integração entre VND e Q-learning permite direcionar a busca local de maneira adaptativa, ajustando a ordem das vizinhanças conforme o desempenho atual da solução. No entanto, mesmo com essa estratégia, pode ser necessário explorar regiões mais amplas do espaço de busca para escapar de ótimos locais mais profundos. Para esse fim, incorporamos uma fase adicional de intensificação baseada na técnica de *Local Branching*, descrita a seguir.

4.5. Local Branching

A técnica de *Local Branching* (LB), introduzida por Fischetti e Lodi [2003], é utilizada para refinar soluções viáveis por meio da exploração de vizinhanças definidas por restrições lineares adicionais ao modelo de programação inteira mista (MIP). Essa abordagem permite delimitar uma região ao redor de uma solução corrente, potencializando a intensificação da busca com o auxílio de um solver MIP [Gonzalez et al., 2016; Gonzalez e Brandão, 2018; Morais et al., 2022; Souto et al., 2024].

No contexto do KPFS, dada uma solução $s \in P$, onde P representa o poliedro definido pelas Restrições (2)–(6), a vizinhança é definida pela seguinte desigualdade:

$$\sum_{j \in X | \bar{x}_j = 0} x_j + \sum_{j \in X | \bar{x}_j = 1} (1 - x_j) \leq \Delta, \tag{7}$$

onde Δ controla o número máximo de variáveis x_j que podem ser modificadas.

Essa fase de intensificação complementa as etapas anteriores, permitindo uma exploração mais aprofundada em torno das soluções promissoras geradas pelo QVND. Dessa forma, o LB conclui os três pilares da abordagem proposta, integrando construção de soluções (Heurística Construtiva), melhoria adaptativa (QVND com Q-learning) e refinamento via *Local Branching*, sob o framework do QVND-HBRKGA.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

5. Experimentos Computacionais

Nesta seção, apresentamos os experimentos realizados para avaliar a eficácia do QVND-HBRKGA na resolução do KPFS, aplicado à alocação de estações de compartilhamento de bicicletas. Na Seção 5.1 detalhamos a configuração da implementação e os parâmetros adotados. Em seguida, as instâncias *benhcmark* são descritas na Seção 5.2, por último, temos na Seção 5.3 a análise dos principais resultados, comparando nossa abordagem com métodos da literatura.

5.1. Detalhes de Implementação e Parâmetros

O QVND-HBRKGA foi implementado em C++ com suporte do GCC 11.4.0 e IBM CPLEX 22.1.1, utilizando a API BRKGA de Toso e Resende [2015]. Os experimentos foram executados em um processador Intel Core i7-11800H (4,60 GHz) com 16 GB de RAM.

A Tabela 1 resume os parâmetros definidos empiricamente para cada componente do método, assegurando reprodutibilidade dos resultados.

Tabela 1: Parâmetros QVND-HBRKGA

Parâmetro	Valor
População (p)	30
Elite (p_e)	0.10
Mutação (p_m)	0.10
α (RCL)	0.50
Δ (Local Branching)	$\lvert X\rvert/2$
Taxa de aprendizado (κ)	0.1
Épocas Q-learning (n_{ep})	5000
Desconto (γ)	0.90

5.2. Instâncias *Benchmark*

As instâncias *benchmark* seguem a proposta de D'Ambrosio et al. [2023], com tamanhos de $n = 300, 500, 700, 800$ e 1.000 itens, distribuídos em dois cenários e três tipos de correlação entre peso, lucro e penalidade: Not Correlated (NC), Correlated (C) e Fully Correlated (FC).

Os cenários diferem pelo número de conjuntos penalizados (l) e pelo tamanho médio desses conjuntos:

- **Cenário 1:** $l = 5n$, $|C_i| \in [2, n/50]$, $h_i = 1$.
- **Cenário 2:** $l = 3n$, $|C_i| \in [2, n/20]$, $h_i = 1$.

A dificuldade das instâncias decorre da alta sobreposição entre os conjuntos de penalidade e do limite de tolerância ($h_i = 1$), o que intensifica os conflitos entre itens. A correlação entre pesos, lucros e penalidades também influencia a complexidade: enquanto em NC os atributos são aleatórios, nos tipos C e FC lucros maiores estão associados a pesos elevados e, consequentemente, a penalidades mais severas.

Essas características tornam as instâncias desafiadoras, exigindo estratégias robustas para equilibrar viabilidade e ganho, especialmente em cenários com forte interdependência entre os itens.

5.3. QVND-HBRKGA: Resultados e discussões

Nesta seção, avaliamos o desempenho da metodologia QVND-HBRKGA na resolução do KPFS, considerando instâncias *benchmark* aplicadas à alocação de estações de compartilhamento de bicicletas. A análise inclui a comparação com dois dos principais métodos da literatura: o BRKGA Cerulli et al. [2024] e a abordagem MFSS Jovanovic e Voß [2024]. Os resultados são discutidos com base na qualidade das soluções obtidas, tempo computacional e comportamento em diferentes cenários e tipos de instâncias.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

5.3.1. Critérios de Avaliação

A principal métrica utilizada foi a lacuna da solução (*gap*) em relação aos melhores valores conhecidos, calculada como:

$$\text{gap} = 1 - \left(\frac{Sol_{\text{heurística}}}{Sol_{\text{CPLEX}}} \right),$$

onde $Sol_{\text{heurística}}$ é o valor da solução obtido pelo método avaliado e Sol_{CPLEX} é o melhor valor reportado em D'Ambrosio et al. [2023], obtido com IBM CPLEX em até 3 horas de tempo limite (sem garantia de otimalidade em alguns casos). Para uma comparação justa entre os tempos computacionais, aplicamos fatores de conversão baseados em *Single Thread Rating* das CPUs disponível em `https://www.cpubenchmark.net/`. Especificamente, um fator de 1,86 foi aplicado ao tempo computacional da abordagem de Cerulli et al. [2024], enquanto os autores do método MFSS não reportaram o tempo computacional.

5.3.2. Comparativo com a literatura

Os resultados demonstram que o QVND-HBRKGA supera consistentemente as abordagens MFSS e BRKGA em termos de qualidade da solução, especialmente em instâncias com maior grau de correlação entre pesos, lucros e penalidades (tipos C e FC). A Figura 2 ilustra as diferenças absolutas de desempenho (Delta) entre nossa abordagem (QVND-HBRKGA) e os métodos comparados (MFSS e BRKGA), indicando vantagem sempre que os valores estão acima de zero.

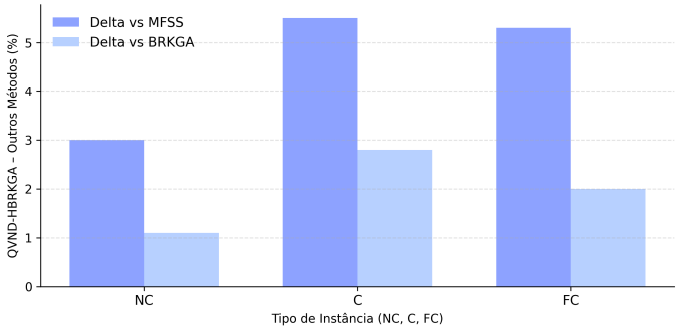


Figura 2: Diferença na qualidade da solução entre QVND-HBRKGA e os métodos MFSS e BRKGA, por tipo de instância.

No Cenário 1, o QVND-HBRKGA obteve soluções superiores em 12 de 15 tipos de instâncias em relação ao MFSS, e em 14 de 15 quando comparado ao BRKGA. Já no Cenário 2, a metodologia atingiu ou superou os melhores valores conhecidos em 13 de 15 casos, mantendo desempenho competitivo ou superior na maioria das situações.

Apesar do tempo computacional do QVND-HBRKGA ser ligeiramente maior que o do BRKGA, as melhorias obtidas na qualidade da solução — frequentemente entre 1% e 2% — justificam esse investimento adicional de tempo, especialmente em aplicações como o planejamento de infraestrutura urbana, onde ganhos pequenos podem ter impacto significativo.

5.3.3. Melhoria Percentual sobre o Estado da Arte

Para reforçar a comparação direta com o BRKGA, atual estado da arte para o KPFS, calculamos a melhoria percentual do QVND-HBRKGA em relação ao BRKGA por meio da expressão

$$\frac{\texttt{custo(QVND-HBRKGA)} - \texttt{custo(BRKGA)}}{\texttt{custo(BRKGA)}} \times 100\%.$$

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

A Figura 3 apresenta essa análise para cada tipo e tamanho de instância. A maioria dos pontos à direita da linha zero confirma a superioridade da nossa abordagem, com destaque para instâncias dos tipos Correlated (C) e Fully Correlated (FC), onde o problema apresenta maior complexidade estrutural.

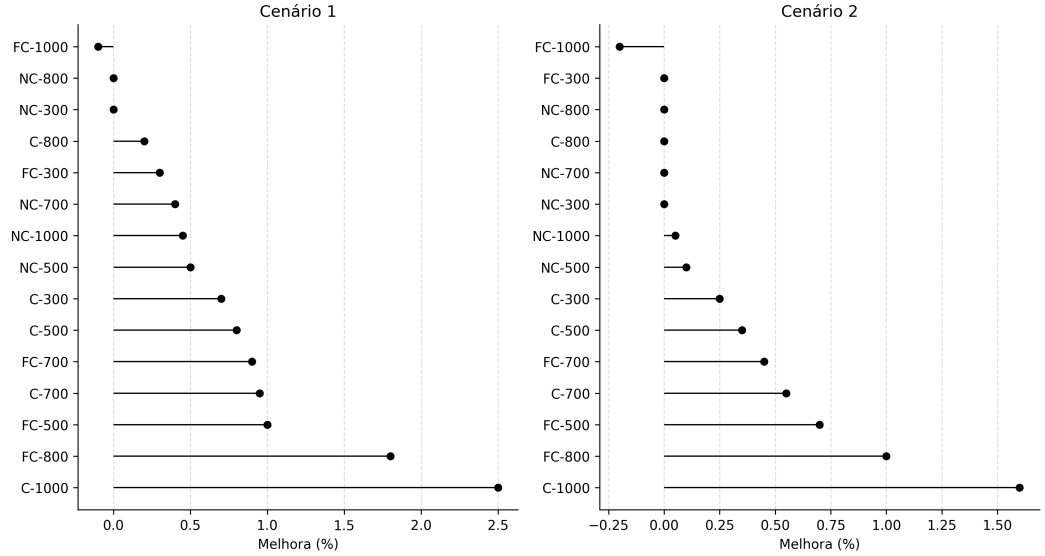


Figura 3: Melhoria percentual do QVND-HBRKGA em relação ao BRKGA (estado da arte).

A análise por cenário revela que o QVND-HBRKGA mantém vantagens consistentes à medida que o tamanho da instância aumenta, demonstrando boa escalabilidade. Essa robustez é particularmente relevante para aplicações em cidades inteligentes, nas quais a quantidade de pontos de alocação (itens) pode crescer substancialmente.

5.3.4. Síntese dos Resultados

A Tabela 2 resume os resultados detalhados para cada tipo de instância e cenário, incluindo valores da solução, tempo computacional e *gap*. Observa-se que, embora o tempo do QVND-HBRKGA seja superior em alguns casos, a melhoria na qualidade das soluções compensa essa diferença, principalmente em instâncias mais complexas.

Tabela 2: Comparação entre os métodos da literatura

		Cenário 1									**Cenário 2**								
Instâncias		**QVND-HBRKGA**			**BRKGA**			**MFSS**			**QVND-HBRKGA**			**BRKGA**			**MFSS**		
Tipo	*n*	Solução	Tempo(s)	Gap(%)	Solução	Time(s)	Gap(%)	Solução	Tempo(s)	Gap(%)	Solução	Tempo(s)	Gap(%)	Solução	Tempo(s)	Gap(%)	Solução	Tempo(s)	Gap(%)
NC	300	**684.00**	181.33	0.00	683.10	14.51	0.13	684.00	-	0.00	**299.60**	115.60	0.00	299.60	10.68	0.00	299.60	-	0.00
	500	**575.10**	192.13	-2.39	570.10	41.60	-1.50	570.20	-	-1.51	**227.60**	184.97	0.00	227.30	26.68	0.13	227.60	-	0.00
	700	**504.10**	199.38	-	502.70	81.44	-	499.70	-	-	**186.90**	188.62	0.00	186.80	45.97	0.05	186.90	-	0.00
	800	**465.10**	212.63	-	462.80	103.27	-	457.60	-	-	**174.30**	190.93	0.00	174.30	59.32	0.00	172.40	-	1.09
	1000	**415.10**	222.76	-	413.80	161.00	-	404.70	-	-	**145.90**	196.02	0.00	145.80	83.87	0.07	144.70	-	0.82
C	300	771.80	185.53	-0.30	**772.70**	15.18	-0.42	772.00	-	-0.32	**443.90**	185.42	0.00	443.30	13.70	0.14	443.90	-	0.00
	500	**854.30**	194.77	-2.42	846.90	52.50	-1.53	843.80	-	-1.16	**347.80**	189.52	-1.34	347.80	35.16	-1.34	346.90	-	-1.08
	700	**739.10**	202.48	-	726.70	100.99	-	733.40	-	-	**274.40**	193.33	-	274.40	62.36	-	273.60	-	-
	800	**683.10**	206.34	-	676.30	134.58	-	670.40	-	-	**251.10**	191.43	-	251.10	81.57	-	248.10	-	-
	1000	**611.00**	217.49	-	608.80	198.68	-	593.60	-	-	**214.60**	197.02	-	214.60	114.47	-	211.00	-	-
FC	300	752.90	183.52	-0.21	751.00	13.48	0.04	**753.00**	-	-0.23	**465.40**	183.52	-0.22	465.40	15.98	-0.22	464.60	-	-0.04
	500	**824.30**	208.40	-2.70	821.90	54.84	-2.40	812.80	-	-1.27	**406.60**	208.40	-0.79	406.10	38.65	-0.67	404.30	-	-0.22
	700	**749.80**	207.76	-	742.60	112.57	-	743.10	-	-	349.40	207.76	-	**349.60**	69.25	-	348.00	-	-
	800	**713.80**	215.34	-	709.20	150.41	-	702.40	-	-	**326.50**	215.34	-	326.40	84.62	-	323.60	-	-
	1000	**647.60**	248.22	-	642.90	229.34	-	634.10	-	-	288.30	248.22	-	**289.30**	119.16	-	285.60	-	-

Esses resultados reforçam o potencial da abordagem QVND-HBRKGA para resolver problemas de alocação sob restrições complexas, como aqueles encontrados em cenários urbanos de planejamento inteligente. A combinação entre geração adaptativa de soluções, busca local orientada por *Reinforcement Learning* e refinamento via *Local Branching* contribui para a eficácia da proposta.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

6. Conclusões e Trabalhos Futuros

Neste trabalho, propusemos o QVND-HBRKGA, um método híbrido que integra o *Biased Random-Key Genetic Algorithm* (BRKGA), Q-learning e *Local Branching* para resolver o *Knapsack Problem with Forfeits Sets* (KPFS), um problema relevante para cenários de alocação sob restrições complexas, como na gestão de infraestruturas em cidades inteligentes. A principal inovação da abordagem está no uso de Q-learning para guiar dinamicamente a escolha das estruturas de vizinhança no processo de busca local, promovendo um equilíbrio entre exploração e intensificação.

Os experimentos computacionais demonstraram que o QVND-HBRKGA supera os principais métodos da literatura, incluindo o MFSS e o BRKGA, especialmente em instâncias de maior complexidade, com forte correlação entre lucros, pesos e penalidades. A metodologia apresentou resultados robustos e escaláveis, obtendo boas soluções em cenários mais complexos.

Como direções para trabalhos futuros, pretendemos investigar a aplicação do QVND-HBRKGA em outros problemas de otimização combinatória que compartilham características estruturais similares, como problemas de roteamento de veículos e escalonamento de tarefas. Além disso, avaliamos como promissora a incorporação de técnicas mais avançadas de *Reinforcement Learning*, como *Deep Q-learning* e *Policy Gradient Methods*, visando aprimorar o processo de tomada de decisão.

7. Agradecimentos

Esta pesquisa foi parcialmente apoiada pela Universidade Federal do Rio de Janeiro (23079.227622/2023-70), Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (307663/2021-3, 304680/2023-0) e Fundaçãoção Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro - FAPERJ (E-26/201.341/2022)

Referências

Batty, M., Axhausen, K. W., Giannotti, F., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., Ouzounis, G., e Portugali, Y. (2012). Smart cities of the future. *The European Physical Journal Special Topics*, 214:481–518.

Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA journal on computing*, 6(2):154–160.

Brandao, J. S., Noronha, T. F., Resende, M. G., e Ribeiro, C. C. (2015). A biased random-key genetic algorithm for single-round divisible load scheduling. *International Transactions in Operational Research*, 22(5):823–839.

Brandão, J. S., Noronha, T. F., Resende, M. G., e Ribeiro, C. C. (2017). A biased random-key genetic algorithm for scheduling heterogeneous multi-round systems. *International Transactions in Operational Research*, 24(5):1061–1077.

Bruck, B. P., Coutinho, W. P., e Munari, P. (2025). The robust bike sharing rebalancing problem: Formulations and a branch-and-cut algorithm. *European Journal of Operational Research*.

Capobianco, G., D'Ambrosio, C., Pavone, L., Raiconi, A., Vitale, G., e Sebastiano, F. (2022). A hybrid metaheuristic for the knapsack problem with forfeits. *Soft Computing*, p. 1–14.

Carrabs, F., Cerulli, R., Pentangelo, R., e Raiconi, A. (2021). Minimum spanning tree with conflicting edge pairs: a branch-and-cut approach. *Annals of Operations Research*, 298(1-2):65–78.

Cerulli, R., D'Ambrosio, C., e Raiconi, A. (2024). A biased random-key genetic algorithm for the knapsack problem with forfeit sets. *Soft Computing*, p. 1–21.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

Cerulli, R., D'Ambrosio, C., Raiconi, A., e Vitale, G. (2020). The knapsack problem with forfeits. In *Combinatorial Optimization: 6th International Symposium, ISCO 2020, Montreal, QC, Canada, May 4–6, 2020, Revised Selected Papers 6*, p. 263–272. Springer.

Chen, H., Wang, W., Cheng, L., e Li, P. (2024). A cooperative optimization method for the layout of shared bicycle parking areas and delivery quantity. *Scientific Reports*, 14(1):4171.

Chiariotti, F., Pielli, C., Zanella, A., e Zorzi, M. (2020). A bike-sharing optimization framework combining dynamic rebalancing and user incentives. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 14(3):1–30.

D'Ambrosio, C., Laureana, F., Raiconi, A., e Vitale, G. (2023). The knapsack problem with forfeit sets. *Computers & Operations Research*, 151:106093.

ElNawawy, A., Lee, J., e Zaki, M. H. (2024). Optimizing smart bike-sharing station placement near public transit stops using gis: A case study of london, ontario. In *2024 IEEE International Conference on Smart Mobility (SM)*, p. 203–208. IEEE.

Ericsson, M., Resende, M. G. C., e Pardalos, P. M. (2002). A genetic algorithm for the weight setting problem in ospf routing. *Journal of combinatorial optimization*, 6(3):299–333.

Fischetti, M. e Lodi, A. (2003). Local branching. *Mathematical Programming*, 98:23–47.

Gonçalves, J. F. e De Almeida, J. R. (2002). A hybrid genetic algorithm for assembly line balancing. *Journal of heuristics*, 8(6):629–642.

Goncalves, J. F. e Resende, M. G. (2004). An evolutionary algorithm for manufacturing cell formation. *Computers & industrial engineering*, 47(2-3):247–273.

Gonçalves, J. F. e Resende, M. G. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17(5):487–525.

Gonzalez, P. H. e Brandão, J. (2018). A biased random key genetic algorithm to solve the transmission expansion planning problem with re-design. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, p. 1–7. IEEE.

Gonzalez, P. H., Simonetti, L., Michelon, P., Martinhon, C., e Santos, E. (2016). A variable fixing heuristic with local branching for the fixed charge uncapacitated network design problem with user-optimal flow. *Computers & Operations Research*, 76:134–146.

Hansen, P. e Mladenovic, N. (2003). A tutorial on variable neighborhood search. *Les Cahiers du GERAD ISSN*, 711:2440.

Hassan, A. e Pillay, N. (2022). Hybridizing a genetic algorithm with reinforcement learning for automated design of genetic algorithms. In *2022 IEEE Congress on Evolutionary Computation (CEC)*, p. 1–8. IEEE.

Hrabec, D., Nverlỳ, V., Víchová, K., Šohaj, K., Peterek, K., e Taraba, P. Location of bike-sharing stations: Optimization model and case studies with insights into population coverage. *Available at SSRN 4849664*.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598

Japa, L., Serqueira, M., Mendonça, I., Bezerra, E., Aritsugi, M., e González, P. H. (2022). A conjugated evolutionary algorithm for hyperparameter optimization. In *2022 IEEE Congress on Evolutionary Computation (CEC)*, p. 1–9. IEEE.

Jovanovic, R. e Voß, S. (2024). Matheuristic fixed set search applied to the multidimensional knapsack problem and the knapsack problem with forfeit sets. *OR Spectrum*, p. 1–37.

Kumar, A., Zhou, A., Tucker, G., e Levine, S. (2020). Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33:1179–1191.

Li, J., Lan, Y., Chen, F., Han, X., e Blazewicz, J. (2021). A fast algorithm for knapsack problem with conflict graph. *Asia-Pacific Journal of Operational Research*, 38(06):2150010.

Ma, M., Preum, S. M., Tarneberg, W., Ahmed, M., Ruiters, M., e Stankovic, J. (2016). Detection of runtime conflicts among services in smart cities. In *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, p. 1–10. IEEE.

Ma, M., Stankovic, J. A., e Feng, L. (2018). Cityresolver: a decision support system for conflict resolution in smart cities. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, p. 55–64. IEEE.

Morais, I., Souto, G., Ribeiro, G. M., Mendonça, I., e González, P. H. (2022). A hybrid brkga approach for the multiproduct two stage capacitated facility location problem. In *2022 IEEE Congress on Evolutionary Computation (CEC)*, p. 1–8. IEEE.

Pferschy, U. e Schauer, J. (2009). The knapsack problem with conflict graphs. *Journal of Graph Algorithms and Applications*, 13(2):233–249.

Song, Y., Wei, L., Yang, Q., Wu, J., Xing, L., e Chen, Y. (2023). Rl-ga: A reinforcement learning-based genetic algorithm for electromagnetic detection satellite scheduling problem. *Swarm and Evolutionary Computation*, 77:101236.

Souto, G., Aritsugi, M., Mendonça, I., Simonetti, L., e González, P. H. (2024). A q-learning hybrid brkga applied to the knapsack problem with forfeits. In *2024 IEEE Congress on Evolutionary Computation (CEC)*, p. 1–8. IEEE.

Souto, G., Morais, I., Faulhaber, L., Ribeiro, G. M., e González, P. H. (2021). A hybrid brkga approach for the two stage capacitated facility location problem. In *2021 IEEE Congress on Evolutionary Computation (CEC)*, p. 2007–2014. IEEE.

Sutton, R. S. e Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Toso, R. F. e Resende, M. G. (2015). A c++ application programming interface for biased random-key genetic algorithms. *Optimization Methods and Software*, 30(1):81–93.

Watkins, C. J. e Dayan, P. (1992). Q-learning. *Machine learning*, 8:279–292.

Yin, Q. e Liu, G. (2020). Resource scheduling and strategic management of smart cities under the background of digital economy. *Complexity*, 2020(1):6624307.

Zhang, X., Xia, S., Li, X., e Zhang, T. (2022). Multi-objective particle swarm optimization with multi-mode collaboration based on reinforcement learning for path planning of unmanned air vehicles. *Knowledge-Based Systems*, 250:109075.

DOI: 10.59254/sbpo-2025-212598
http://live-proceedings.pantheonsite.io/p/212598