



APLICANDO MÉTODOS INOVADORES PARA RESOLVER O PROBLEMA DA MOCHILA COM CONFLITOS

LAÍS GONÇALVES VIANA – lais.viana@ufv.br
UNIVERSIDADE FEDERAL DE VIÇOSA – UFV – CAMPUS RIO PARANAÍBA

HELEN VITALINE DE CASTRO SANTOS - helen.castro@ufv.br
UNIVERSIDADE FEDERAL DE VIÇOSA – UFV – CAMPUS RIO PARANAÍBA

THIAGO HENRIQUE NOGUEIRA - thiagoh.nogueira@ufv.br
UNIVERSIDADE FEDERAL DE VIÇOSA – UFV – CAMPUS RIO PARANAÍBA

LARISSA SOUSA CAMPOS - larissa.sousa@ufv.br
UNIVERSIDADE FEDERAL DE VIÇOSA – UFV – CAMPUS RIO PARANAÍBA

RAIANE RIBEIRO MACHADO GOMES - raianemachado@gmail.com
UNIVERSIDADE FEDERAL DE VIÇOSA – UFV – CAMPUS RIO PARANAÍBA

ÁREA: 3. PESQUISA OPERACIONAL

SUBÁREA: 3.1 - MODELAGEM, SIMULAÇÃO E OTIMIZAÇÃO

RESUMO: O PROBLEMA DA MOCHILA COM CONFLITOS (PMC) É UMA QUESTÃO NP-COMPLETA QUE VISA MAXIMIZAR O VALOR TOTAL DOS ITENS EM UMA MOCHILA, RESPEITANDO TANTO O LIMITE DE PESO QUANTO AS RESTRIÇÕES DE CONFLITOS ENTRE ITENS. ESTE ARTIGO EXPLORA MÉTODOS INOVADORES PARA SOLUCIONAR O PMC, INCLUINDO O USO DA HEURÍSTICA CONSTRUTIVA RVND. DUAS INSTÂNCIAS DO PMC DE COMPLEXIDADES VARIADAS, PREVIAMENTE APRESENTADAS EM OUTROS ESTUDOS, SÃO ANALISADAS. OS RESULTADOS DEMONSTRAM QUE A EFICIÊNCIA E A QUALIDADE DAS SOLUÇÕES SÃO INFLUENCIADAS PELAS DIFERENTES ABORDAGENS ALGORÍTMICAS E CRITÉRIOS DE ORDENAÇÃO APLICADOS, VARIANDO CONFORME AS CARACTERÍSTICAS DE CADA INSTÂNCIA. DESSA FORMA, A ESCOLHA CRITERIOSA DESSES ELEMENTOS PODE OTIMIZAR A RESOLUÇÃO DE PROBLEMAS SIMILARES AO PMC.

PALAVRAS-CHAVES: PMRC, RVND, CONSTRUTIVA.

1. INTRODUÇÃO

Segundo Borges (2020), o PMRC consegue auxiliar na resolução de inúmeros problemas da realidade, sendo bastante usual para propor soluções como por exemplo, o problema de embalagens industriais, no que diz respeito ao seu empacotamento, abrangendo as n variáveis, que auxiliam no planejamento da produção. Outrossim, Lee (2019), aborda mais um exemplo de sua utilidade, trazendo consigo, um breve comentário da sua origem, que se deu em cenários militares cuja finalidade era otimizar o preenchimento das mochilas dos soldados com diferentes itens de valores e pesos variáveis, respeitando suas restrições físicas.

Todavia, a literatura propõe que são necessárias abordagens e métodos para que estes problemas sejam resolvidos (OLIVAS et al., 2021), que segundo Wei (2021) podem ser classificados em Algoritmos Exatos, de Aproximação e Algoritmos Heurísticos. Os algoritmos exatos incluem métodos de programação dinâmica, ramificação e limite de abordagem, enquanto nos problemas de grandes instâncias são utilizadas meta-heurísticas, como Busca Tabu e Algoritmos Genéticos (ASSI, HARATY, 2018).

Soukaina et al. (2018), alertam sobre o uso destes Métodos Exatos que mesmo resolvendo pequenas e moderadas instâncias na mochila ainda podem não ter prática quando utilizado um número maior. Isso faz com que os métodos utilizando heurísticas e metaheurísticas sejam aconselhados para este tipo de problema, uma vez que podem trazer soluções satisfatórias em um tempo razoável atendendo a todas as variáveis e restrições.

Para Badrloo et al. (2019), os métodos utilizando as heurísticas têm como objetivo descrever uma abordagem não precisa para resolver um problema, enquanto os algoritmos metaheurísticos o utilizam para chegar o mais perto do ótimo. Sendo assim, a heurística presente no artigo será a construtiva, que segundo Adouani et al. (2022), busca solucionar de maneira viável, por meio de iterações, sem ficar preso em áreas semelhantes e na busca por regiões que podem ser consideradas promissoras.

A meta-heurística que explora a vizinhança local de forma determinística é a descida de vizinhança variável (VND), na qual entende que diferentes tipos de estruturas não possuem o mesmo mínimo local (FREITAS, PENNA, 2022). Quando a ordem das vizinhanças é obtida de maneira aleatória é chamada RVND, tendo em vista que nesta há menos risco de o resultado cair em uma falsa solução ótima, pois próximo a ela podem existir picos e vales locais (GOKALP, UGUR, 2019).

Por fim, o presente estudo objetiva-se em discutir as nuances do método PMCR, partindo, inicialmente, de duas instâncias, denominadas de Set 1 e Set 2: o Set 1 foi retirada por duas distintas literaturas, a primeira, proposta em 2006 por Hifi e Michrafy, contendo os primeiros 50 dados, a segunda, proposta em 2017 por Quan e Wu, contém outros 50 dados, assim tem-se o Set 1, com o total de 100 dados. Por sua vez, os dados do Set 2 é composto por 930 dados que foram retirados de um artigo de Bettinelli et al. em 2017. A diversidade dos conjuntos de dados permite aprofundamento no método PMCR, justamente por serem duas instâncias de diferentes tamanhos e níveis de complexibilidade.

2. REVISÃO DE LITERATURA

Para o presente trabalho foram utilizados dois artigos científicos base relacionados ao tema. Um deles, dos autores Alves e Hashino (2022), apresenta diferentes heurísticas, como a construtiva e gulosa; metaheurísticas e o RVND como busca local para resolver o PMRC. O outro, escrito por Andrade (2015), propõe uma heurística construtiva gulosa, um simulated annealing, um variable neighborhood search e uma meta-heurística híbrida para resolver o problema. Em seguida foi necessária adequação dos mesmos para que os métodos se ajustassem ao problema estudado.

Abordando as possíveis heurísticas, Ezugwu (2019), explicita que a partir de uma iterativa escolha a heurística gulosa seleciona, progressivamente, as melhores opções locais sempre na tentativa de encontrar um local ótimo, aumentando seu “apetite”, porém por vezes o resultado do algoritmo não é o esperado, salientando a utilização de outros métodos como, por exemplo, a heurística construtiva. Neste sentido, o presente estudo utiliza a heurística construtiva onde Cerulli (2020), pondera que a heurística construtiva cria soluções passo a passo, acrescentando as variáveis existentes no problema de forma gradual, sendo considerado uns algoritmos simples e eficientes para a resolução de PMRC.

Quanto às metaheurísticas alguns métodos são recomendados para a resolução do PMRC um deles é o Algoritmo Genético, que gera aleatoriamente uma população inicial realizando mutações e cruzamentos dos dados, que criam novas soluções. (LAMBORA; CHOPRA, 2019). Outrossim, a utilização de Algoritmo Genético não é recomendada para problemas que apresentem elevados números de dados, uma vez que sua utilização computacional é dita como lenta, sendo recomendado, assim, outros meios (LEON, 2019).

Assim, De Freitas (2020), propõe em sua pesquisa a busca por vizinhança variável, como metaheurística, onde salienta que é capaz de solucionar e otimizar a solução computacional, minimizando seu tempo de processamento, sendo caracterizado como um método flexível e eficiente, proporcionando uma melhora iterativa do algoritmo.

Em 2018, Soukaina et al. combinam um algoritmo genético com busca local, utilizando a heurística construtiva para gerar a população inicial do algoritmo para resolver o problema da mochila com restrições de conflito. No ano posterior, Rezoug et al. desenvolvem uma heurística baseada em programação linear inteira para o problema da mochila, no qual aplicam ao problema de maximizar a vida útil de redes de sensores sem fio, utilizando a heurística construtiva para resolver uma relaxação linear do problema.

Em outro trabalho, a fim de melhorar as soluções encontradas, Haddad et al. (2018), aplicaram o Random Variable Neighborhood Descent (RVND) com a metaheurística Iterated Local Search (ILS) em diferentes vizinhanças em ordem aleatória, buscando melhorias locais na solução atual. Em seguida, uma nova estrutura de vizinhança é utilizada para otimizar a divisão e implantação de cada coleta e entrega, refinando ainda mais a solução.

O importante trabalho proposto por Hifi e Michrafy (2006), apresenta a utilização da busca local para o PMRC, utilizando especiais restrições disjuntivas, onde tem-se um par de itens excludentes, onde apenas um dos recursos são utilizados. Ainda sobre o estudo, são utilizados dois algoritmos gulosos com forma de partida que ao decorrer do estudo são afinados por meio da busca local, tal estudo é base para diversos trabalhos na área, fornecendo os primeiros 50 dados da presente pesquisa.

No ano seguinte, Hifi e Michrafy (2007), expõem mais uma contribuição para o tema, estudando três versões baseadas em reduzir e, por consequência, otimizar a busca local, este importante estudo torna-se base de pesquisas sobre o tema. Seguindo a mesma área de pesquisa Quan e Wu (2017), apoiam-se na computação paralela para a resolução de PMRC's, trazendo em seu estudo um algoritmo cooperativo a base de busca em multi-vizinhanças. Parte dos dados base do trabalho é utilizado no presente estudo.

Por fim, torna-se válido citar e explicar a contribuição do trabalho que origina os restantes dados utilizados nesta pesquisa. Bettinelli et al. (2017), então promove uma grande contribuição para a resolução de PMRC's, a novidade em questão é a utilização de branch-and-bound fato que encurtou o tempo de computação para os problemas similares, por meio da derivação de de soluções ótimas, caracterizando seu estudo como notável estudo para os problemas de mochila.

Os estudos realizados por diferentes pesquisadores demonstraram a importância e eficácia de diversas técnicas para a resolução de problemas de otimização. Algoritmos genéticos combinados com busca local e heurísticas construtivas mostraram-se promissores na

resolução do problema da mochila com restrições de conflito. A utilização Random Variable Neighborhood Descent (RVND) em conjunto da aplicação de busca local em problemas do PMRC, com restrições disjuntivas e a utilização de algoritmos construtivos, serviu como base para estudos subsequentes. A computação paralela também se mostrou útil para a resolução dos problemas, empregando algoritmos cooperativos e aplicação de relaxação em restrições essas abordagens avançadas têm impulsionado a pesquisa e contribuído para a resolução eficiente de problemas de otimização em diferentes áreas.

3. METODOLOGIA

Procurando a otimização na resolução do Problema da Mochila com Restrições de conflitos (PMRC), a metodologia do presente estudo apresenta de maneira detalhada o funcionamento dos algoritmos para a heurística e metaheurística propostas, bem com o modelo matemático base para a resolução do desafio junto a metaheurística de busca local.

3.1 Construção da solução inicial

Em primeiro plano, explicita que foi necessário a padronização dos dados e a divisão do PMRC dois subproblemas, o primeiro, sendo o problema do conjunto independente e o segundo o problema da mochila 0-1, para o primeiro foi apresentado a heurística construtiva, a abordagem para a resolução dessa heurística é utilizada da seguinte forma.

Faz-se uma ordenação dos valores de forma decrescente inicia-se uma lista vazia, denominada de lista de itens selecionados, e uma segunda denominada de itens conflitantes utilizada para armazenar itens conflitantes que foram selecionados, por conseguinte é feita uma verificação, para cada lista, verifica se o item possui conflito, se não houver conflito adiciona a primeira lista e atualiza a segunda. Por fim, retorna a lista de itens selecionados, como mostra o pseudocódigo a seguir.

Algoritmo 1: Heurística Construtiva

```
1: Criar um modelo PLI vazio
2: Vetor: Vs
3: Lista de itens com conflito: Vc
4: enquanto houver itens na lista i faça
5:   seleciona item i
6:   para cada item j em i faça
7:     se existe conflito entre i e j então
9:       remove item k
12:    fim se
13:  fim para
14: fim enquanto
15: resolvePLI (itens sem conflito)
```

- Parâmetros:
Vs: Vetor itens sem conflitos
Vc: Vetor com conflito.
i : item atual selecionado no vetor conflito.
j: item atual em dentro de i
k : item removido do vetor conflito

Posteriormente a execução da heurística construtiva e com a aquisição dos itens que não conflitam entre si é possível a aplicação do método clássico do Problema da Mochila 0-1, no qual foi proposto, pela primeira, por Dantzig (1957), o segundo subproblema do desafio

proposto pela presente pesquisa, sendo assim, resolvendo o problema de conflito entre os itens é possível a simplificação do método a:

$$\text{Maximizar } \sum_{i=1}^n v_i x_i$$

Sujeito a:

1. $\sum_{i=1}^n p_i x_{ij} \leq C$
2. $x_i \in \{0,1\}, \forall i \in \{1,2, \dots, n\}$

Fonte: Dantzig (1957)

- Parâmetros:

v_i : valor do item i

C : Capacidade máxima.

p_i : Peso do item i

i : item

- Variáveis de Decisão:

x_i : Variável binária que indica se o item i é selecionado ou não.

x_{ij} : Variável binária que indica se o item i é selecionado ou não, em relação ao valor

j .

3.2 RVND

O RVND utiliza uma estratégia de busca local iterativa, explorando diferentes vizinhanças para encontrar soluções melhores (BUCCO et al., 2017). Essa abordagem oferece diversidade e maior capacidade de escapar de ótimos locais, tornando-se uma escolha adequada para problemas de otimização (SILVA; LEAL, 2020). Para sua implementação torna-se válido determinar inicialmente uma estrutura de vizinhança.

3.2.1 Estruturas de vizinhança

A fim de investigar as possibilidades de solução do problema, definiu-se duas categorias de ações para serem executadas em uma solução. Cada ação gera uma estrutura de vizinhança, denotada por $N_k(s)$, em que $k = 1, 2, 3$ e é definida da seguinte maneira:

- $N_1(s)$ Vizinhança de troca: Consiste em trocar um item selecionado na mochila por um item não selecionado;
- $N_2(s)$ Vizinhança de remoção: Envolve a remoção de um item selecionado da mochila, abrindo espaço para outros itens serem inseridos em seu lugar

3.2.2 Algoritmo genérico

Utilizando um pseudocódigo genérico, o algoritmo começa com uma solução inicial e busca iterativamente por soluções vizinhas que melhorem a função objetivo. Ao encontrar um vizinho melhor, a solução é atualizada e a busca reiniciada.

Algoritmo 2: RVND

```
1: função s: solução inicial
2:  $N = \{N_1, \dots, N_{\max}\}$ 
3:  $k \leftarrow 1$ 
4: enquanto  $k \leq k_{\max}$  faça
5:    $s' \leftarrow \text{procurar\_vizinho}(N_k, s)$ 
6:   se  $f(s') < f(s)$  então
7:      $s \leftarrow s'$ 
9:      $k \leftarrow 2$ 
10:  se não
11:     $k \leftarrow k + 1$ 
12:  fim se
13: fim enquanto
14: retorna s
```

Fonte: Mladenovic e Hansen (1997)

O algoritmo recebe uma solução inicial "s" para um problema e uma lista de vizinhanças "N" em ordem aleatória. Em um loop, o algoritmo explora cada vizinhança, procurando por um vizinho melhor usando a função "procurar vizinho". Se um vizinho melhor for encontrado, a solução é atualizada e o loop reinicia. Caso contrário, o algoritmo passa para a próxima vizinhança. Após explorar todas as vizinhanças, a solução final é retornada. O objetivo é encontrar a melhor solução localmente otimizada usando diferentes vizinhanças.

O critério de parada para o algoritmo será estabelecido por um número máximo de iterações. Após atingir esse limite, o algoritmo encerra a busca por melhores vizinhos e retornaria à solução atual. Isso evitaria a execução indefinida do algoritmo e permitiria avaliar seu desempenho dentro de um número pré-determinado de iterações.

4. RESULTADOS

Os algoritmos foram codificados em linguagem de programação Python usando o compilador Visual Studio Code. Os experimentos foram realizados em um computador Desktop contendo uma Intel(R) Core (TM) i5-8265U CPU @ 1.60GHz 1.80 GHz, equipado com 8 GB RAM executando sobre o sistema operacional Windows.

As metaheurísticas propostas foram testadas em instâncias de problemas contidos na literatura. As instâncias são divididas em dois conjuntos: set I e set II. O set I é composto por 100 instâncias denominadas jIk, onde j varia de 1 a 20 e k varia de 1 a 5. As primeiras 50 instâncias são denominadas 1Ik a 10Ik (HIFI; MICHRAFY, 2006), enquanto as outras 50 instâncias 11Ik a 20Ik (QUAN; WU, 2017). O set II é composto por 930 instâncias denominadas C15, R15, SR e SC (BETTINELLI et al., 2017). Seus dados estão definidos na Tabela 1, incluindo o número de itens (n), a capacidade da mochila (C), a densidade dos conflitos (d) e o total de instâncias.

TABELA 1 – Dados das instâncias

instance_name	n	C	d	#total
11k a 101k	{500,1000}	{1800,2000}	[0.05,0.1]	50
111k a 201k	{1500,2000}	{4000}	[0.04,0.2]	50
R15	$120 \leq n \leq 500$	15000	[0.1,0.9]	180
C15	$120 \leq n \leq 500$	15000	[0.1,0.9]	270
SC	$500 \leq n \leq 1000$	$1000 \leq C \leq 2000$	[0.001,0.05]	240
SR	$500 \leq n \leq 1000$	$1000 \leq C \leq 2000$	[0.001,0.05]	240

Fonte: Alves; Hoshino (2022)

Na Tabela 2 são apresentadas as configurações das metaheurísticas utilizadas. Nela está descrito o nome da heurística, o método utilizado e o critério de ordenação empregado em cada método.

TABELA 2 - Configurações das metaheurísticas

Heurística	Método	Critério
PLI-RVND	Mochila PLI + RVND	-
C-PLI-v	Construtiva + Mochila PLI	valor
C-PLI-p	Construtiva + Mochila PLI	peso
C-PLI-c	Construtiva + Mochila PLI	conflito

Fonte: Autores (2024)

Foram conduzidos experimentos utilizando diferentes metaheurísticas para resolver o problema, onde cada heurística possuía um método e critério de ordenação distintos. Os algoritmos foram testados em várias instâncias do problema, sendo executados 25 vezes para cada. Essa abordagem permitiu uma análise mais robusta dos resultados.

Inicialmente, foram realizados testes utilizando o algoritmo da heurística construtiva, sem a técnica de busca local (RVND), com o objetivo de identificar qual critério de ordenação proporciona os melhores resultados.

TABELA 3 - Resultados da heurística construtiva

Heurística	Total de soluções ótimas	
	11K-101K	111K-201K
C-PLI	2	4
C-PLI-v	1	1
C-PLI-p	0	2
C-PLI-c	7	12

Fonte: Autores (2024)

Para a heurística C-PLI, na instância 11K-101K foram apresentadas 2 soluções ótimas em um grupo amostral de 25 possibilidades. Já na instância 111K-201K foram apresentadas 4 soluções ótimas. Quando aplicadas aos critérios específicos (valor, peso e conflito), os resultados foram 1, 0 e 7, respectivamente, para a instância 1. Já para a segunda instância, os resultados obtidos foram 1, 2 e 12 resultados ótimos.

Após analisar os dados da Tabela 3, verificou-se que a heurística com o critério de ordenação baseado em conflitos alcançou os melhores resultados para ambas as instâncias. Deste modo, com o critério de ordenação estabelecido, foi realizada a execução do algoritmo com a inclusão da técnica de busca local (RVND) para refinar os resultados, conforme mostra a Tabela 4.

TABELA 4 - Resultados da heurística construtiva com busca local

Heurística	Total de soluções ótimas	
	11K-101K	111K-201K
C-PLI-v + (RVND)	2	0
C-PLI-p + (RVND)	0	0
C-PLI-c + (RVND)	11	20

Fonte: Autores (2024)

Através destes resultados, é visto que a lógica se mantém, onde mesmo com a aplicação da técnica de busca local, o critério de ordenação baseado no conflito se mostra melhor do que os demais, sofrendo um aumento em relação ao método anterior. Ao mesmo tempo, os critérios de valor e peso apresentaram uma leve baixa quando comparados ao método sem busca local. Munidos de tal resultado, implementa-se ao set2 a heurística C-PLI-c + RVND, como é ilustrado na Tabela 5.

TABELA 5 - Resultados da heurística construtiva com busca local para o set2

Instâncias	Total de soluções ótimas
R15	9
C15	18
SC	0
SR	1

Fonte: Autores (2024)

Os resultados obtidos mostram um número diferente de soluções ótimas encontradas. Entre as instâncias avaliadas, a R15 apresentou 9 soluções ótimas, a C15 teve 18 soluções ótimas, enquanto as instâncias SC e SR apresentaram 0 e 1 soluções ótimas nos testes realizados.

Os resultados nas instâncias R15 e C15 podem ser atribuídos a algumas características específicas desses problemas. Essas instâncias apresentam um número menor de itens em comparação com as outras, o que pode facilitar a otimização devido a menos combinações possíveis. Além disso, ter uma capacidade fixa da mochila maior nessas instâncias oferece mais flexibilidade na escolha dos itens, aumentando as chances de encontrar soluções ótimas. A densidade de conflito mais alta nessas instâncias também implica em mais restrições e possíveis conflitos entre os itens, o que pode direcionar a busca para soluções mais precisas.

5. CONCLUSÃO

Os resultados obtidos a partir dos experimentos com as metaheurísticas aplicadas ao problema da mochila 0-1 demonstraram a influência significativa dos critérios de ordenação e da técnica de busca local na obtenção de soluções ótimas. A heurística construtiva, quando combinada com o critério de ordenação baseado em conflitos, apresentou os melhores resultados para as instâncias analisadas, tanto no set I quanto no set II. Esses resultados evidenciam a importância de considerar as características específicas de cada instância, como o número de itens, a capacidade da mochila e a densidade de conflito, para determinar a abordagem mais adequada na resolução do problema.

Além disso, a inclusão da técnica de busca local (RVND) mostrou-se eficaz no refinamento das soluções encontradas, proporcionando melhorias nos resultados em relação à heurística construtiva isolada. No entanto, é importante ressaltar que os critérios de valor e peso apresentaram um desempenho ligeiramente inferior em comparação ao critério de conflito, indicando que a consideração dos conflitos entre os itens pode ser mais relevante na busca por soluções ótimas.

Esses resultados reforçam a importância de explorar diferentes abordagens algorítmicas e critérios de ordenação no contexto do problema da mochila 0-1, levando em consideração as particularidades de cada instância. A combinação adequada de heurísticas, critérios de ordenação e técnicas de busca local pode resultar em soluções mais precisas e eficientes, contribuindo para a otimização de problemas de natureza similar.

REFERÊNCIAS

ANDRADE, Luana Alves et al. Métodos heurísticos aplicados ao problema de agrupamento de entregas em veículos de uma frota heterogênea. 2015.

ASSIMI, Hirad et al. Otimização bi-objetiva evolutiva para o problema dinâmico da mochila com restrição de chance baseado em objetivos limitados pela cauda. arXiv preprint arXiv:2002.06766 , 2020.

ADOUANI, Yassine; JARBOUI, Bassem; MASMOUDI, Malek. A matheuristic for the 0–1 generalized quadratic multiple knapsack problem. Optimization Letters, p. 1-22, 2022.

ASSI, Maram; HARATY, Ramzi A. A survey of the knapsack problem. In: 2018 International Arab Conference on Information Technology (ACIT). IEEE, 2018. p. 1-6.

BADRLOO, Soheila; HUSSEINZADEH KASHAN, Ali. Combinatorial optimization of permutation-based quadratic assignment problem using optics inspired optimization. Journal of applied research on industrial engineering, v. 6, n. 4, p. 314-332, 2019.

Bettinelli, A., Cacchiani, V., e Malaguti, E. (2017). A branch-and-bound algorithm for the knapsack problem with conflict graph. INFORMS Journal on Computing, 29.

BORGES, Yulle GF et al. Exact algorithms for class-constrained packing problems. Computers & Industrial Engineering, v. 144, p. 106455, 2020.

BUCCO, Guilherme Brandelli; BORNIA-POULSEN, Camilo José; BANDEIRA, Denise Lindstrom. Desenvolvimento de um modelo de programação linear para o Problema da Construção de Grades Horárias em Universidades. Gestão & Produção, v. 24, p. 40-49, 2017.

CACCHIANI, Valentina et al. Knapsack problems-An overview of recent advances. Part II: Multiple, multidimensional, and quadratic knapsack problems. Computers & Operations Research, p. 105693, 2022.

CERULLI, Raffaele et al. The knapsack problem with forfeits. In: Combinatorial Optimization: 6th International Symposium, ISCO 2020, Montreal, QC, Canada, May 4–6, 2020, Revised Selected Papers 6. Springer International Publishing, 2020. p. 263-272.

DANTZIG, George B. Discrete-variable extremum problems. Operations research, v. 5, n. 2, p. 266-288, 1957.

DE FREITAS, Júlia Cária; PENNA, Puca Huachi Vaz. A variable neighborhood search for flying sidekick traveling salesman problem. International Transactions in Operational Research, v. 27, n. 1, p. 267-290, 2020.

EZUGWU, Absalom E. et al. A comparative study of meta-heuristic optimization algorithms for 0–1 knapsack problem: Some initial results. *IEEE Access*, v. 7, p. 43979-44001, 2019.

GOKALP, Osman; UGUR, Aybars. A multi-start ILS–RVND algorithm with adaptive solution acceptance for the CVRP. 2019.

GURSKI, Frank; REHS, Carolin. Solutions for the knapsack problem with conflict and forcing graphs of bounded clique-width. *Mathematical Methods of Operations Research*, v. 89, n. 3, p. 411-432, 2019.

HADDAD, Matheus Nohra et al. Large neighborhood-based metaheuristic and branch-and-price for the pickup and delivery problem with split loads. 2018.

Hifi, M. e Michrafy, M. (2006). A reactive local search-based algorithm for the disjunctively constrained knapsack problem. *Journal of the Operational Research Society*, 57.

HIFI, Mhand; MICHRAFY, Mustapha. Reduction strategies and exact algorithms for the disjunctively constrained knapsack problem. *Computers & operations research*, v. 34, n. 9, p. 2657-2673, 2007

LEE, Ji Hwan et al. 1H1-5 A study on the optimal dimension design of a Military Backpack considering body size characteristics. *Ningen Kougaku*, v. 55, n. Supplement, p. 1H1-5-1H1-5, 2019.

LEON, Paul et al. A Taboo-search Algorithm for 3D-binpacking Problem in Containers. In: *New Knowledge in Information Systems and Technologies: Volume 1*. Springer International Publishing, 2019. p. 229-240.

LI, Wei et al. An ant colony optimization algorithm with adaptive greedy strategy to optimize path problems. *Journal of Ambient Intelligence and Humanized Computing*, p. 1-15, 2022.

OLIVAS, Frumen et al. Enhancing hyperheuristics for the knapsack problem through fuzzy logic. *Computational Intelligence and Neuroscience*, v. 2021, 2021.

QUAN, Zhe; WU, Lei. Cooperative parallel adaptive neighbourhood search for the disjunctively constrained knapsack problem. *Engineering Optimization*, v. 49, n. 9, p. 1541-1557, 2017.

SILVA JÚNIOR, Hélio Santana da. Estudo comparativo de meta-heurísticas: otimização do modelo de alocação na Agência Nacional de Aviação Civil. 2021.

SOUKAINA, Laabadi et al. A hybrid genetic algorithm for solving 0/1 knapsack problem. In: *Proceedings of the International Conference on Learning and Optimization Algorithms: Theory and Applications*. 2018. p. 1-6.

XIE, Yue et al. Evolutionary algorithms for the chance-constrained knapsack problem. In: *Proceedings of the Genetic and Evolutionary Computation Conference*. 2019. p. 338-346.

WANG, Yun-long et al. Research on intelligent design method of ship multi-deck compartment layout based on improved taboo search genetic algorithm. Ocean Engineering, v. 225, p. 108823, 2021.