PROBLEMA DA MOCHILA COM ITENS IRREGULARES E INCERTEZA NOS DEFEITOS DA PLACA

Layane Rodrigues de Souza Queiroz e Marina Andretta
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo - Campus São Carlos,
Av. Trabalhador São Carlense, 400, Centro, 13566-590, São Carlos - SP, Brasil.
layaner@usp.br, andretta@icmc.usp.br

RESUMO

O presente trabalho lida com o problema da mochila bidimensional considerando o corte de itens irregulares em uma placa com defeitos. Enquanto os defeitos só são conhecidos no momento do corte, deve-se selecionar de antemão quais itens poderão ser cortados da placa. Os itens finais não podem ter qualquer defeito e o objetivo é maximizar o lucro com o corte da placa. Propomos um modelo de programação estocástica de dois estágios que faz uso de um conjunto discreto de cenários contendo a realização dos defeitos. As decisões de primeiro estágio envolvem selecionar os itens para o corte, enquanto as de segundo estágio consideram o posicionamento dos itens na placa considerando os defeitos existentes podendo, então, haver o cancelamento de itens selecionados. Realizamos testes computacionais em instâncias adaptadas da literatura que consideram três tipos de defeitos e oito cenários.

PALAVRAS CHAVE. Programação Estocástica de Dois Estágios. Problema da Mochila com Itens Irregulares. Defeitos.

POI - PO na Indústria. OC - Otimização Combinatória.

ABSTRACT

The present work deals with the two-dimensional knapsack problem considering the cut of irregular items in a defective plate. While defects are only known at the time of cutting, we should select in advance which items can be cut from the plate. The final items must not contain defects, and the objective is to maximize the profit by cutting the plate. We propose a two-stage stochastic programming model that uses a discrete set of scenarios containing the realization of defects. The first stage decisions involve selecting the items to be cut. In contrast, the second stage decisions consider the positioning of the items on the plate considering the existing defects and, therefore, there may be the cancellation of selected items. We perform computational tests on instances adapted from the literature that consider three types of defects and eight scenarios.

KEYWORDS. Two-Stage Stochastic Programming. Knapsack Problem with Irregularly shaped Itens. Defects.

ORI - OR in Industry. CO - Combinatorial Optimization.

1. Introdução

Nas empresas de corte de couro, de tecido e metal-mecânica, uma placa (isto é, uma peça grande de couro, de tecido ou de metal) está disponível para ser cortada e obter itens, que podem apresentar formato irregular. Esses itens são usados para a produção de roupas, bolsas, calçados, artefatos de esporte, entre outros. É comum que a placa apresente defeitos em algumas partes, seja oriundos da extração, manuseio ou transporte, inviabilizando o corte de itens sobre essas partes [Baldacci et al., 2014].

Nesse trabalho consideramos uma empresa que realiza o corte de itens irregulares a partir de uma placa retangular que pode apresentar defeitos. A empresa faz o pedido da placa com antecendência, devido a questões de preparo e transporte por parte do fornecedor. O pedido da placa inclui suas dimensões (e características de material) e está relacionado com os itens que a empresa selecionou inicialmente para obter da placa. Todavia, os defeitos na placa são identificados somente após a entrega e quando ela adentra o estágio de corte, momento em que alguns itens previamente selecionados podem ter a sua produção cancelada.

A empresa deseja um plano de corte (para a placa) que dê o máximo lucro líquido, que depende do lucro obtido com a produção e entrega de itens e do custo dado pelo possível cancelamento de alguns itens selecionados. O plano de corte contém os itens previamente selecionados para serem obtidos da placa e deve ser elaborado no momento de compra da placa. Porém, como os possíveis defeitos na placa não são conhecidos *a priori*, pode ser necessário o cancelamento de alguns itens quando do efetivo corte da placa. Dessa forma, tratamos os defeitos como dados incertos. Esse é um Problema do tipo Mochila bidimensional com itens Irregulares e Incerteza sobre os Defeitos da placa, que denominamos de 2PMI-ID.

Problemas de corte de itens irregulares vêm sendo estudados na literatura de problemas de corte e empacotamento, com um número de contribuições relativamente menor comparado aos problemas com itens regulares, especialmente aqueles com formato retangular [Wäscher et al., 2007]. Restrições comuns em problemas dessa natureza dizem respeito a garantir a não sobreposição entre itens e que o posicionamento dos itens não venha a extrapolar as dimensões da placa. Devido à geometria irregular dos itens, torna-se necessário o uso de ferramentas adicionais para que sejam escritas essas restrições. Algumas ferramentas incluem a discretização dos itens em matrizes de pontos (método *raster*), o cálculo da posição relativa, com o uso de funções de distância (*phi-functions*), a construção de polígonos de obstrução entre pares de itens (*no-fit polygon*) e a verificação da interseção entre segmentos e de inclusão de pontos (trigonometria direta) [Bennell e Oliveira, 2008].

Com relação ao problema da mochila com itens irregulares, Martins e Tsuzuki [2010] propuseram uma heurística baseada em recozimento simulado para os casos com placas retangulares e irregulares. O posicionamento dos itens é baseado no cálculo de polígonos de obstrução e de regiões livres de sobreposição. Valle et al. [2012] desenvolveram heurísticas baseadas em GRASP (*Greedy Randomized Adaptive Search Procedure*), com a resolução dos casos em que existe apenas uma cópia de cada item e quando há ilimitadas cópias de cada item. O posicionamento dos itens é feito observando os polígonos de obstrução. Silveira [2013] utilizou um algoritmo genético cujos cromossomos representam a sequência em que os itens devem ser posicionados na placa. Os itens são posicionados por uma heurística de agrupamento que faz uso dos polígonos de obstrução. Dalalah et al. [2014] consideraram placas retangulares e irregulares e fizeram a proposta de uma heurística construtiva, que posiciona os itens conforme a utilização da envoltória convexa.

Por sua vez, Baldacci et al. [2014] resolveram o caso com placa irregular que surge na indústria de couro. Os autores consideraram a placa discretizada em uma matriz de pontos e apre-

sentaram um modelo de programação linear inteira e uma relaxação lagrangeana. Mundim et al. [2018] desenvolveram uma heurística geral para problemas de corte de itens irregulares, como o da mochila e de corte de estoque. Os autores propuseram diferentes regras de posicionamento, algumas delas baseadas na *bottom-left*, e conseguiram melhorar grande parte dos resultados anteriormente apresentados na literatura. Souza Queiroz e Andretta [2020b] propuseram duas heurísticas, a primeira baseada em um algoritmo genético de chaves aleatórias viciadas e a outra é uma busca em vizinhança variável. Nas heurísticas a solução é codificada através de um vetor e o posicionamento dos itens considera o cálculo de polígonos de obstrução e três regras inspiradas na *bottom-left*.

Alguns poucos estudos têm considerado defeitos e zonas de qualidade nas placas. Heistermann e Lengauer [1995] lidaram com um caso real encontrado na indústria de couro, em que a placa é irregular e contém zonas de qualidade e defeitos. A heurística dos autores considera um conjuntos de itens candidatos e tenta posicioná-los sem gerar sobreposição caso seja possível, senão posiciona apenas o item que resulta na melhor ocupação da placa. Em Yuping et al. [2005], as placas possuem zonas de qualidade e os autores adotaram uma estratégia de penalização para que os itens sejam posicionados de forma viável. Em Lee et al. [2008], o problema considera múltiplas placas irregulares e com defeitos, sendo que os itens são posicionados observando o seu valor, em ordem decrescente.

A resolução de um problema de corte de couro para a indústria automobilística foi feita em Alves et al. [2012]. Os autores consideraram a maximização do uso da placa para obter os itens finais, sendo apresentada uma heurística de busca em vizinhança variável. Os autores consideraram quatro operadores para gerar posicionamentos diferente dos itens na placa e, assim, gerar uma solução final melhor. Mundim e Andretta [2014] consideraram o problema com vários recipientes irregulares, visando minimizar o número de recipientes necessários para o corte dos itens demandados. Os autores apresentaram duas heurísticas baseadas na regra de posicionamento *bottom-left*. A não sobreposição entre os itens e com os defeitos é garantida pelos polígonos de obstrução. Pinto et al. [2016] melhoraram os resultados de Alves et al. [2012], propondo uma heurística construtiva que simula o posicionamento de itens na placa, então selecionando o item e ponto que geram a melhor ocupação da placa. Os autores propuseram uma busca local para testar itens não posicionados e, assim, aumentar a ocupação da placa.

A literatura é ainda mais escassa quando se busca pelo tratamento de incertezas dentro do problema. Mundim [2017] considerou incertezas na demanda dos itens para o problema com múltiplas placas, propondo um modelo de programação estocástica de dois estágios. No modelo se penaliza a falta ou o excesso de itens produzidos a mais que o necessário, conforme um conjunto amostrado de cenários. Em Souza Queiroz e Andretta [2020a] há também um modelo de programação estocástica de dois estágios, porém para o problema de corte em faixa com incertezas associadas a demanda dos itens. Os resultados obtidos com o modelo indicaram que desconsiderar as incertezas acarretaria em soluções de maior custo total.

Nesse trabalho, propomos um modelo de programação estocástica de dois estágios para o 2PMI-ID. O primeiro estágio considera decisões sobre quais itens selecionar, para obter da placa, enquanto o segundo estágio contém os cenários com a realização dos defeitos na placa e, então, alguns itens poderão ser cancelados por não mais haver um posicionamento viável na placa. O posicionamento dos itens é feito observando os polígonos de obstrução que são discretizados em uma matriz de pontos, isto é, da junção do método *raster* com o *no-fit polygon*, resultando no *no-fit raster* [Toledo et al., 2013]. A definição do problema é dada na Seção 2, enquanto a Seção 3 apresenta o modelo proposto. A Seção 4 contém o estudo computacional realizado sobre o modelo, que é resolvido por um algoritmo *branch-and-cut* e aplicado para resolver instâncias adaptadas da

literatura de problemas de corte de itens irregulares. Por fim, conclusões finais e direções para a continuação desse trabalho são dadas na Seção 5.

2. Definição do Problema

O problema da mochila bidimensional com itens irregulares e incerteza sobre os defeitos da placa (isto é, 2PMI-ID) é uma generalização do problema da mochila bidimensional, que por sua vez é NP-Difícil [Garey e Johnson, 1979]. O problema é definido sobre o plano Cartesiano, com uma placa posicionada no primeiro quadrante e com origem em $(0, 0)$. O eixo das abscissas (eixo x) está associado com a dimensão da largura (W) e o eixo das ordenadas (eixo y) está associado com a dimensão do comprimento (H), que são valores fixos e conhecidos *a priori*. Existe um conjunto de itens I disponíveis para seleção e cada item $i \in I$ possui área a_i e um vértice de referência $r_i = (r_i^x, r_i^y)$ que é o menor retângulo, em termos de dimensões e sem rotação, que circunscreve o item. Assumimos que os itens possuem orientação fixa (isto é, não podem ser rotacionados) e são posicionados nessa orientação através do seu vértice de referência, que é o vértice de menor ordenada e, em caso de empate, de menor abscissa. Além disso, cada item i tem um lucro p_i, dada a sua seleção e produção, e um custo de cancelamento c_i.

O problema considera um conjunto D de possíveis defeitos que a placa pode ter. Cada defeito $d \in D$ é modelado como um item irregular de dimensões, área e vértice de referência conhecidos. Por outro lado, não se conhece quais defeitos e seu posicionamento dentro da placa, sendo tratados como dados incertos. O objetivo do problema é determinar um plano de corte viável que retorne o máximo lucro ao cortar a placa. Um plano de corte é viável quando os itens a serem produzidos podem ser posicionados sem sobreposição, inteiramente dentro da placa e não contêm partes com defeito.

A seleção dos itens que poderão fazer parte do plano de corte ocorre no momento inicial, em que se faz o pedido da (ou se define a) placa. Nesse momento, a empresa define os itens do conjunto I que deveriam ser produzidos a partir do corte da placa. Como a placa está sujeita a defeitos, que são conhecidos somente em um segundo momento (de entrega e produção dos itens), alguns itens selecionados poderão ser cancelados, incorrendo um custo de cancelamento. O plano de corte somente é definido nesse segundo momento, onde há o conhecimento dos defeitos e, assim, pode-se confirmar quais dos itens selecionados se pode posicionar de forma viável, enquanto alguns poderão vir a ser cancelados. Dessa forma, deseja-se um plano de corte viável que resulte no máximo lucro.

Com relação ao posicionamento dos itens, usamos o *no-fit raster* para garantir a não sobreposição entre itens e o *inner-fit raster* para assegurar que os itens estejam completamente dentro da placa [Toledo et al., 2013]. O *no-fit raster* NFR_{ij} consiste no cálculo do polígono de obstrução para cada par de itens i e j, sendo um item fixo (i) e o outro orbital (j). O item orbital é transladado ao redor do item fixo, sempre encostando no item fixo, mas sem gerar sobreposição, formando um polígono cujo interior indica sobreposição entre os itens. Esse polígono é, então, discretizado em uma matriz de pontos, com valores iguais a "1" indicando a sobreposição e, caso contrário, valores iguais a "0". Por outro lado, o *inner-fit raster* IFR_i consiste no polígono obtido ao transladar cada item i, internamente, sempre encostando uma das bordas da placa. Esse polígono é, então, discretizado em uma matriz de pontos, com valores iguais a "0" indicando que o item pode ser posicionado sem extrapolar as dimensões da placa e, caso contrário, com valores iguais a "1".

3. Modelo de Programação Estocástica

Apresentamos um modelo de programação estocástica de dois estágios para o 2PMI-ID. Os defeitos (e suas localizações na placa) são considerados variáveis aleatórias com realizações discretas conforme uma distribuição de probabilidade conhecida. Seja $\Omega = \{1, 2, \ldots, S\}$ o conjunto

de possíveis estados, isto é, de cenários $s \in \Omega$ amostrados e contendo a realização dos defeitos na placa.

No modelo desenvolvido, o primeiro estágio do problema está relacionado com decisões *here-and-now* envolvendo a seleção dos itens conforme a placa a ser entregue pelo fornecedor. Por outro lado, no segundo estágio são tomadas decisões *wait-and-see* envolvendo o posicionamento dos itens selecionados e, devido à realização das variáveis aleatórias, isto é, o conhecimento dos defeitos na placa, temos ainda decisões envolvendo o possível cancelamento de itens. O plano de corte da placa deve ser viável e ter o máximo lucro esperado, que envolve o lucro com os itens selecionados e produzidos menos o custo esperado com o cancelamento de itens.

Os parâmetros e as variáveis de decisão do modelo de programação estocástica são:

- I: conjunto de itens irregulares disponíveis para a seleção;
- v_i: lucro associado com a seleção do item i no primeiro estágio;
- c_i: custo associado ao cancelamento do item i no segundo estágio;
- π^s: probabilidade de ocorrência do cenário $s \in \Omega$;
- D^s: defeitos na placa, dado o cenário $s \in \Omega$;
- y_i: variável binária, de primeiro estágio, que recebe o valor 1 se o item $i \in I$ faz parte dos itens selecionados para produção; ou, o valor 0, caso contrário;
- x^s_{ipq}: variável binária, de segundo estágio, que recebe o valor 1 se o item i escolhido (isto é, $y_i = 1$), tem o seu vértice de referência posicionado no ponto (p, q) da malha associada a placa; ou, o valor 0, caso contrário; para o cenário $s \in \Omega$;
- z^s_i: variável binária, de segundo estágio, que recebe o valor 0 se o item i selecionado deve ser cancelado; ou, o valor 1, caso contrário; para o cenário $s \in \Omega$;
- M: número suficientemente grande.

O modelo para o 2PMI-ID é de programação linear inteira com dois estágios e risco neutro, assumindo que a solução terá desempenho de longo prazo. O objetivo é maximizar o lucro esperado, considerando a seleção dos itens a produzir (primeiro estágio) e o posicionamento e possível cancelamento de itens conforme os defeitos na placa (segundo estágio). A função objetivo (1) do primeiro estágio considera o lucro pela seleção de itens ($v_i y_i$) e o custo esperado pelo cancelamento de itens no segundo estágio ($Q(y, \boldsymbol{\xi})$), em que $\boldsymbol{\xi} = [\xi_s]$, com $\xi_s = \{x^s_{ipq}, z^s_i\}$, representa o vetor aleatório do cenário s. As restrições (2) definem o domínio das variáveis de decisão de primeiro estágio.

$$\text{Maximizar } v_i y_i + Q(y, \boldsymbol{\xi}) \tag{1}$$

$$y_i \in \{0, 1\}, \qquad \forall\, i \in I. \tag{2}$$

O modelo de segundo estágio possui função objetivo (3), que busca pela minimização da soma dos custos de cancelamento de itens considerando a probabilidade de ocorrência π^s de cada cenário $s \in \Omega$. Como itens selecionados, porém cancelados, não podem ter seu lucro contabilizado na solução, assumimos que $c_i \geq v_i$, para cada item i.

$$Q(y, \boldsymbol{\xi}) = \text{Minimizar} \sum_{s \in \Omega} \pi_s \left(\sum_{i \in I} c_i (y_i - z_i^s) \right) \tag{3}$$

$$z_i^s \leq y_i, \qquad \forall\, s \in \Omega,\ i \in I; \tag{4}$$

$$\sum_{j \in I} \sum_{(u,v) \in NFR_{ij}^{(p,q)}} x_{juv}^s \leq (1 - x_{ipq}^s) M, \qquad \forall\, s \in \Omega,\ i \in I,\ (p,q) \in IFR_i^s; \tag{5}$$

$$\sum_{(p,q) \in IFR_i^s} x_{ipq}^s = z_i^s, \qquad \forall\, s \in \Omega,\ i \in I; \tag{6}$$

$$z_i^s \in \{0,1\}, \qquad \forall\, s \in \Omega,\ i \in I; \tag{7}$$

$$x_{ipq}^s \in \{0,1\}, \qquad \forall\, s \in \Omega,\ i \in I,\ (p,q) \in IFR_i^s. \tag{8}$$

As restrições (4) do modelo de segundo estágio impõem que um item não selecionado no primeiro estágio deve ser marcado com o mesmo estado que os itens cancelados, dado cada cenário $s \in \Omega$. As restrições (5) asseguram, para cada cenário $s \in \Omega$, que se o item i está posicionado em (p, q) na placa, então o item j não pode ser posicionado em qualquer um dos pontos (u, v) do conjunto $NFR_{ijs}^{(p,q)}$. As restrições (6) impõem que somente os itens i selecionados e não cancelados devem ser posicionados na placa, para cada cenário $s \in \Omega$. Por fim, as restrições (7) e (8) definem que as variáveis de segundo estágio são binárias.

Nas restrições (5) e (6), definimos o $NFR_{ij}^{(p,q)}$ como sendo o conjunto de pontos (u, v) que estão no interior do NFR_{ij}, entre i, o item fixo, e j, o item orbital, dado que i está com o seu vértice de referência posicionado em (p, q). Os pontos (u, v) são aqueles em que, se o item j tem o seu vértice de referência posicionado em algum deles, faz com que haja sobreposição com i, que está posicionado em (p, q). Por outro lado, o conjunto $IFR_i^s \subseteq IFR_i$ contém os pontos da placa, dado o cenário $s \in \Omega$, onde o vértice de referência do item i pode ser posicionado de forma que o item não contenha partes com defeito e que esteja inteiramente contido na placa.

4. Resultados Computacionais

O modelo apresentando na seção anterior foi implementado na linguagem C++, sendo resolvido com o algoritmo *branch-and-cut* presente no resolvedor *Gurobi Optimizer*, versão 9.1, com as configurações padrões. O computador usado nos experimentos possui sistema operacional linux Ubuntu 16.04.7 LTS, processador Intel Xeon E3-1245v5 de 3,5 GHz e 32 GB de RAM.

As instâncias utilizadas nos testes foram adaptadas da literatura do problema da mochila bidimensional com itens irregulares e disponibilizadas por Souza Queiroz e Andretta [2020b]. Os principais dados de cada instância são apresentados na Tabela 1, contendo o nome, a quantidade total de itens disponíveis para a seleção e as dimensões da placa. Adotamos a escala de 1 ponto a cada 1 unidade de distância para obter as matrizes com os *no-fit raster* e *inner-fit raster*. Com relação à função objetivo de primeiro e segundo estágios do modelo, consideramos $v_i = a_i$, com a_i sendo o valor da área do item, e $c_i = 1,5v_i$, para cada item $i \in I$.

A geração dos cenários considerou uma árvore de possibilidades [Ma et al., 2010], com três tipos diferentes defeitos (um triângulo, um quadrado e um losango). Cada cenário considera a presença ou não de cada um dos defeitos e a posição do defeito na placa foi determinada de forma aleatória usando uma distribuição uniforme. Há um total de 8 cenários por instância, que são as possibilidades entre não ter qualquer defeito na placa até ter todos os defeitos presentes na placa.

Tabela 1: Instâncias adaptadas para os testes computacionais.

Instância	Trabalho	#itens	H	W	Instância	Trabalho	#itens	H	W
blasz2	Oliveira et al. [2000]	16	15	16	blazewicz1	Toledo et al. [2013]	7	15	6
rco1	Toledo et al. [2013]	7	15	7	shapes2	Toledo et al. [2013]	8	40	14
fu5	Alvarez-Valdes et al. [2013]	5	38	14	shirts1-2	Rodrigues e Toledo [2017]	13	40	13
three	Alvarez-Valdes et al. [2013]	3	7	4	threep2	Alvarez-Valdes et al. [2013]	6	7	7
poly1c	Rodrigues e Toledo [2017]	15	40	13	threep3	Alvarez-Valdes et al. [2013]	9	7	10

Cada instância está associada a três casos de probabilidade para a ocorrência dos cenários, que são: *equiprovável*, com a probabilidade de 50% associada a ter o defeito d na placa e os outros 50% para não ter tal defeito presente na placa; *otimista*, com a probabilidade de 25% associada a ter o defeito d na placa e os outros 75% para não ter tal defeito presente na placa; e *pessimista* com a probabilidade de 75% associada a ter o defeito d na placa e os outros 25% para não ter tal defeito presente na placa. Por exemplo, para o caso *otimista*, o cenário onde não há qualquer defeito na placa possui probabilidade de ocorrência igual a $75\% \times 75\% \times 75\% = 42.19\%$, onde a probabilidade de 75% está associada ao evento de *não* ter o defeito d na placa. Por outro lado, no cenário cuja placa contém todos os defeitos, a sua probabilidade de ocorrência é dada por $25\% \times 25\% \times 25\% = 1.56\%$, com a probabilidade de 25% associada ao evento de *sim*, ou seja, de ter o defeito d na placa, para $d = 1, 2, 3$. A Tabela 2 apresenta a árvore de cenários com a probabilidade de ocorrência de cada cenário.

Tabela 2: Árvore de cenário para as instâncias usadas nos testes computacionais.

				Probabilidades		
Defeito 1	Defeito 2	Defeito 3	Cenário	Equiprovável	Otimista	Pessimista
NÃO	NÃO	NÃO	#1 NÃO-NÃO-NÃO	12.50%	42.19%	1.56%
		SIM	#2 NÃO-NÃO-SIM	12.50%	14.06%	4.69%
	SIM	NÃO	#3 NÃO-SIM-NÃO	12.50%	14.06%	4.69%
		SIM	#4 NÃO-SIM-SIM	12.50%	4.69%	14.06%
SIM	NÃO	NÃO	#5 SIM-NÃO-NÃO	12.50%	14.06%	4.69%
		SIM	#6 SIM-NÃO-SIM	12.50%	4.69%	14.06%
	SIM	NÃO	#7 SIM-SIM-NÃO	12.50%	4.69%	14.06%
		SIM	#8 SIM-SIM-SIM	12.50%	1.56%	42.19%

As Tabelas 3 a 5 contêm os resultados após aplicar o modelo sobre as instâncias considerando, respectivamente, os três casos, equiprovável, otimista e pessimista. Os resultados indicam que todas as instâncias foram resolvidas na otimalidade, com gap igual a zero. Além disso, o tempo computacional requerido pelo Gurobi ficou abaixo dos 410 segundos no pior caso (instância blasz2). De forma geral, é perceptível a variação em termos de função objetivo quando se passa de um caso ao outro, em que se observa um melhor valor objetivo para o caso otimista (cujos cenários com menos defeitos na placa possuem maior probabilidade de ocorrência), seguido do equiprovável e, por fim, do pessimista. Notamos que somente nas instâncias poly1c, shapes2 e shirts1-2 todos os itens são selecionados e há o mesmo valor de função objetivo, o que pode ser explicado pela pouca influência dos cenários (defeitos) sobre o posicionamento dos itens.

Analisando o cancelamento de itens conforme os casos nas Tabelas 3 a 5, observamos que o cenário 1 não requer cancelamento em qualquer caso, visto que neste cenário não há defeitos. Nos demais cenários, o cancelamento de itens ocorre com maior frequência quando há mais defeitos e no caso otimista. Nesse caso, em particular, a probabilidade de ocorrência dos cenários com maior

número de defeitos é menor. O caso com menor quantidade de cancelamentos é o pessimista, já que os cenários com maior número de defeitos têm maior probabilidade de ocorrência. Em geral, os cenários 4, 6, 7 e 8 possuem o maior número de itens cancelados, independente do caso.

Tabela 3: Resultados do caso equiprovável.

instância	#itens	#itens selecionados	objetivo	gap (%)	tempo (s)	#itens cancelados no cenário							
						#1	#2	#3	#4	#5	#6	#7	#8
blasz2	16	12	165,8	0,0	130,4	0	1	1	1	0	0	1	0
blazewicz1	7	5	49,5	0,0	2,0	0	0	0	0	0	1	1	1
fu5	5	4	347,1	0,0	2,4	0	0	0	0	0	0	1	0
poly1c	15	15	315,5	0,0	277,4	0	0	0	0	0	0	0	0
rco1	7	6	61,2	0,0	9,2	0	1	1	2	0	1	1	1
shapes2	8	8	320,0	0,0	14,6	0	0	0	0	0	0	0	0
shirts1-2	13	13	275,0	0,0	5,8	0	0	0	0	0	0	0	0
three	3	1	5,6	0,0	0,0	0	0	0	1	0	1	0	0
threep2	6	4	23,0	0,0	0,1	0	1	1	1	0	0	2	2
threep3	9	5	37,1	0,0	0,3	0	0	0	1	0	0	1	0

Tabela 4: Resultados do caso otimista.

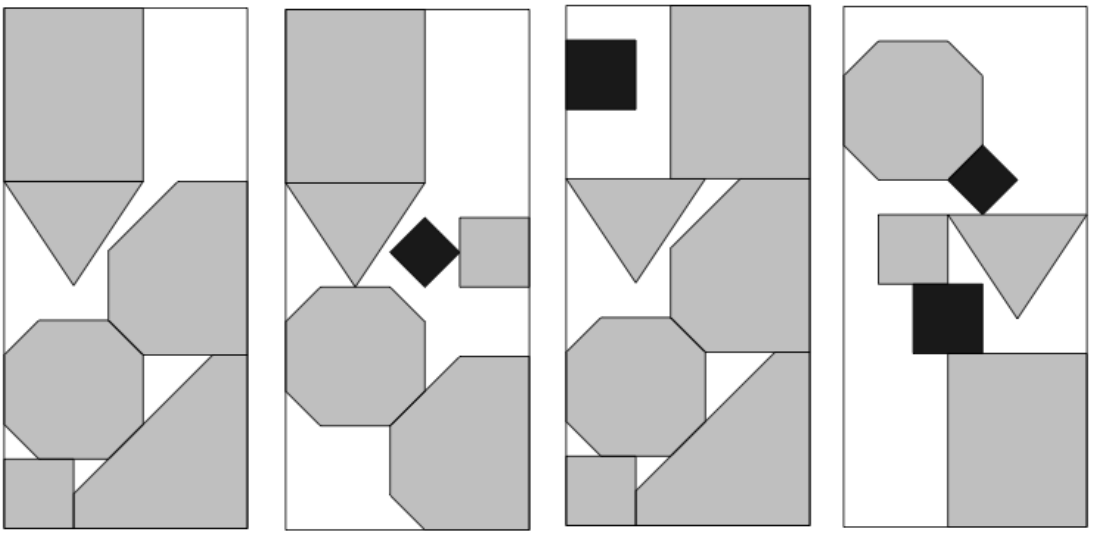
instância	#itens	#itens selecionados	objetivo	gap (%)	tempo (s)	#itens cancelados no cenário							
						#1	#2	#3	#4	#5	#6	#7	#8
blasz2	16	12	174,4	0,0	404,9	0	0	0	0	0	0	0	0
blazewicz1	7	5	52,9	0,0	5,9	0	0	0	0	0	1	1	1
fu5	5	4	380,8	0,0	1,3	0	0	0	0	0	0	1	1
poly1c	15	15	315,5	0,0	291,8	0	0	0	0	0	0	0	0
rco1	7	6	70,2	0,0	6,6	0	1	1	2	0	1	1	1
shapes2	8	8	320,0	0,0	14,3	0	0	0	0	0	0	0	0
shirts1-2	13	13	275,0	0,0	5,8	0	0	0	0	0	0	0	0
three	3	2	9,0	0,0	0,0	0	1	1	1	1	2	1	1
threep2	6	4	27,6	0,0	0,1	0	1	1	1	0	0	2	2
threep3	9	6	42,3	0,0	0,3	0	1	1	1	1	0	1	0

Tabela 5: Resultados do caso pessimista.

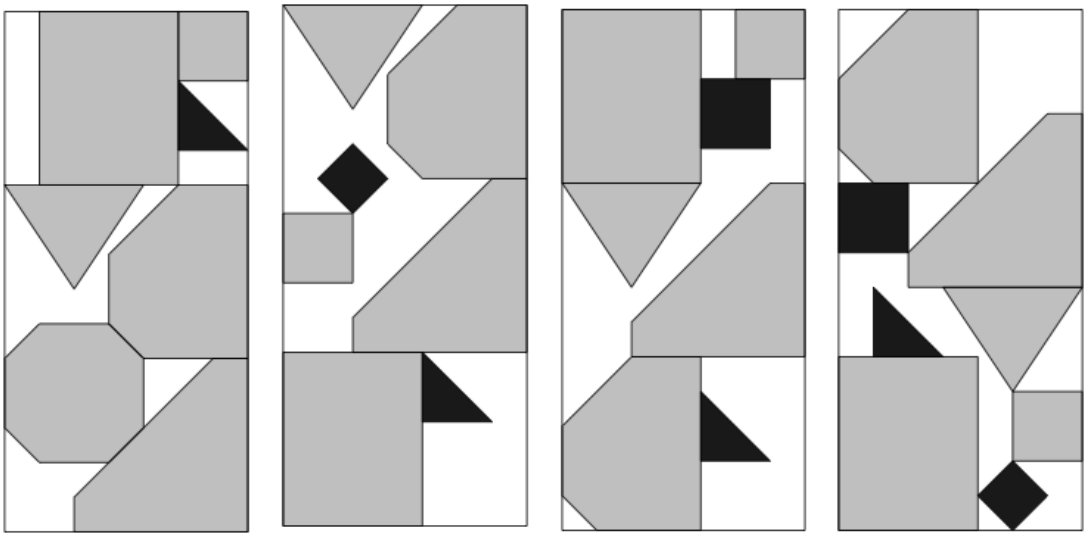
instância	#itens	#itens selecionados	objetivo	gap (%)	tempo (s)	#itens cancelados no cenário							
						#1	#2	#3	#4	#5	#6	#7	#8
blasz2	16	12	160,6	0,0	58,1	0	0	0	1	0	0	0	0
blazewicz1	7	5	46,4	0,0	0,9	0	0	0	0	0	1	1	1
fu5	5	4	301,4	0,0	1,2	0	0	0	0	0	0	1	0
poly1c	15	15	315,5	0,0	296,6	0	0	0	0	0	0	0	0
rco1	7	3	59,4	0,0	8,3	0	0	0	1	0	0	0	0
shapes2	8	8	320,0	0,0	6,2	0	0	0	0	0	0	0	0
shirts1-2	13	13	275,0	0,0	5,8	0	0	0	0	0	0	0	0
three	3	1	5,2	0,0	0,0	0	0	0	1	0	1	0	0
threep2	6	3	18,9	0,0	0,1	0	0	0	0	0	0	1	1
threep3	9	5	32,8	0,0	0,3	0	0	0	1	0	0	1	0

A Figura 1 ilustra a solução obtida para a instância rco1 no caso otimista, em que os itens da cor preta representam os defeitos na placa. Essa instância possui um total de 7 itens, sendo 6 deles selecionados no primeiro estágio, com o cancelamento de itens ocorrendo nos cenários 2, 3, 4,

6, 7 e 8. Observamos que os itens são reposicionados na placa conforme a localização dos defeitos, embora haja o cancelamento de dois itens no cenário 4.



(a) Cenário 1 (b) Cenário 2 (c) Cenário 3 (d) Cenário 4



(e) Cenário 5 (f) Cenário 6 (g) Cenário 7 (h) Cenário 8

Figura 1: Solução para a instância rco1 no caso otimista.

5. Conclusões

Este trabalho apresentou um modelo de programação estocástica de dois estágios para o problema da mochila com itens irregulares e defeitos na placa, com incertezas associadas aos defeitos. O primeiro estágio do modelo decide quais itens selecionar para o corte da placa, enquanto o segundo estágio lida com as decisões para definir um plano de corte viável. O modelo, que é de programação linear inteira, foi resolvido com o algoritmo *branch-and-cut* do resolvedor Gurobi, fazendo uso das ferramentas *no-fit raster* e *inner-fit raster* para garantir um posicionamento viável dos itens na placa.

Os testes computacionais sobre diferentes configurações de instâncias mostram que o modelo se comporta bem, permitindo ao resolvedor obter a solução ótima para todas as instâncias em todos os casos. Além disso, os cenários otimistas, com menos defeitos e maior probabilidade de ocorrência, são os que apresentam maior número de itens cancelados por instância. Por sua vez, o caso otimista é o que fornece os melhores valores de função objetivo, enquanto o caso pessimista tem os menores valores de função objetivo e menos itens cancelados.

Como continuação desse trabalho, pretendemos avaliar o impacto e a importância da aleatoriedade no problema, a partir da análise do *valor esperado da informação perfeita* e do *valor da solução estocástica*. Buscaremos ainda gerar mais instâncias e investigar outras formas de considerar os defeitos na placa. A proposta de heurísticas para acelerar a convergência do algoritmo *branch-and-cut* também faz parte dos trabalhos futuros desejados.

Agradecimentos

O presente trabalho foi realizado com o apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001 e a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP - 2016/01860-1 e 2018/07240-0).

Referências

Alvarez-Valdes, R., Martinez, A., e Tamarit, J. (2013). A branch & bound algorithm for cutting and packing irregularly shaped pieces. *International Journal of Production Economics*, 145(2):463 – 477.

Alves, C. M. M., Brás, P. A. F., Carvalho, J. M. V., e Pinto, T. M. P. (2012). A variable neighborhood search algorithm for the leather nesting problem. *Mathematical Problems in Engineering*, 183: 28.

Baldacci, R., Boschetti, M. A., Ganovelli, M., e Maniezzo, V. (2014). Algorithms for nesting with defects. *Discrete Applied Mathematics*, 163, Part 1:17 – 33.

Bennell, J. e Oliveira, J. (2008). The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, 184(2):397 – 415.

Dalalah, D., Khrais, S., e Bataineh, K. (2014). Waste minimization in irregular stock cutting. *Journal of Manufacturing Systems*, 33(1):27 – 40.

Garey, M. R. e Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco: Freeman.

Heistermann, J. e Lengauer, T. (1995). The nesting problem in the leather manufacturing industry. *Annals of Operations Research*, 57:147–173.

Lee, W.-C., Ma, H., e Cheng, B.-W. (2008). A heuristic for nesting problems of irregular shapes. *Computer-Aided Design*, 40(5):625 – 633.

Ma, Z., Kwon, R. H., e Lee, C.-G. (2010). A stochastic programming winner determination model for truckload procurement under shipment uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, 46:49 – 60.

Martins, T. C. e Tsuzuki, M. S. G. (2010). Simulated annealing applied to the irregular rotational placement of shapes over containers with fixed dimensions. *Expert Systems with Applications*, 37:1955–1972.

Mundim, L. R. e Andretta, M. (2014). Problema de corte de itens irregulares na fabricação de luvas de couro. In *XLVI Simpósio Brasileiro de Pesquisa Operacional*, p. 1–12, Salvador-BA, Brasil.

Mundim, L. R., Andretta, M., Carravilla, M. A., e Oliveira, J. F. (2018). A general heuristic for two-dimensional nesting problems with limited-size containers. *International Journal of Production Research*, 56(1-2):709–732.

Mundim, L. R. (2017). *Mathematical models and heuristic methods for nesting problems*. Doutorado em ciências da computação e matemática computacional, Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, SP, Brasil.

Oliveira, J. F., Gomes, A. M., e Ferreira, J. S. (2000). Topos – a new constructive algorithm for nesting problems. *OR-Spektrum*, 22(2):263–284.

Pinto, T. M. P., Alves, C. M. M., Carvalho, J. M. V., e Brás, P. A. F. (2016). Heuristic methods for the leather nesting problem in the automotive industry. *International Journal of Business Excellence*, 9(3):332–347.

Rodrigues, M. O. e Toledo, F. M. (2017). A clique covering mip model for the irregular strip packing problem. *Computers & Operations Research*, 87:221 – 234.

Silveira, T. (2013). Problemas de empacotamento com itens irregulares: Heurísticas e avaliação de construtores de nfp. Mestrado em ciências da computação, Instituto de Computação, Universidade de Campinas, Campinas, SP, Brasil.

Souza Queiroz, L. R. e Andretta, M. (2020a). Modelo de programação estocástica para um problema de corte de itens irregulares. In *LII Simpósio Brasileiro de Pesquisa Operacional*, p. 1–12, JoÃ£o Pessoa-PB, Brasil.

Souza Queiroz, L. R. e Andretta, M. (2020b). Two effective methods for the irregular knapsack problem. *Applied Soft Computing*, 95:106485.

Toledo, F. M. B., Carravilla, M. A., Ribeiro, C., Oliveira, J. F., e Gomes, A. M. (2013). The dotted-board model: a new mip model for nesting irregular shapes. *International Journal of Production Economics*, 145:478–487.

Valle, A. M. D., Queiroz, T. A., Miyazawa, F. K., e Xavier, E. C. (2012). Heuristics for two-dimensional knapsack and cutting stock problems with items of irregular shape. *Expert Systems with Applications*, 39(16):12589 – 12598.

Wäscher, G., Haubner, H., e Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109 – 1130.

Yuping, Z., Shouwei, J., e Chunli, Z. (2005). A very fast simulated re-annealing algorithm for the leather nesting problem. *International Journal of Advanced Manufacturing Technology*, 25(11): 1113–1118.