

Sobre uma Heurística de Redução de Padrões de Corte para o Problema de Corte de Estoque¹

K.C. POLDI², M.N. ARENALES³, Instituto de Ciências Matemáticas e de Computação, USP, 13560-970 São Carlos, SP, Brasil.

Resumo. Neste artigo tratamos o problema de gerar soluções inteiras para o problema de corte de estoque unidimensional e, também, de reduzir o número de padrões de corte. Estudamos métodos heurísticos propostos na literatura e algumas variações diretas deles para o problema de arredondamento. Estudamos, também, uma heurística de redução de padrões de corte e apresentamos os testes computacionais realizados com esta heurística.

1. Introdução

Problemas de Corte de Estoque consistem na otimização do processo de cortagem de peças maiores, disponíveis em estoque, para a produção de peças menores, em quantidades encomendadas. Tais problemas vêm sendo investigados nas últimas 4 décadas, desde os trabalhos pioneiros de Gilmore e Gomory [3, 4]. Destacamos duas questões importantes: a geração de soluções inteiras para o problema de corte de estoque e a redução do número de padrões de corte.

Este artigo descreve métodos heurísticos para obtenção da solução inteira, e para a redução do número de padrões de corte. O propósito é analisar a qualidade das soluções encontradas por esta heurística, comparando-as com os resultados apresentados no trabalho de Foerster e Wäscher [2].

2. Definição do Problema

O *problema de corte de estoque unidimensional inteiro* pode ser definido como:

Considere que temos disponível em estoque um número suficientemente grande de objetos (barras, bobinas etc) de um determinado comprimento L , que chamaremos de *peças em estoque*, e um conjunto de pedidos, com demanda conhecida $d_i, i = 1, \dots, m$, de *itens* de comprimentos $l_i, i = 1, \dots, m$ (os comprimentos dos itens são tais que: $l_i \leq L$). O problema então consiste em produzir os itens demandados a partir do corte das peças em estoque, atendendo à demanda e de modo

¹Apoio FAPESP e CNPq.

²kelly@icmc.usp.br.

³arenales@icmc.usp.br.

que o número de objetos em estoque necessários para satisfazê-la seja minimizado. Outros critérios, tais como perda, custo, etc., podem ser definidos, sem perda de generalidade deste estudo; apenas a seção 4. é específica para o critério de número mínimo de objetos.

Problemas de corte de estoque unidimensional têm várias aplicações industriais, como por exemplo, na indústria de papel, onde bobinas jumbos devem ser cortadas em tamanhos encomendados pelos clientes; na indústria de aço, onde bobinas de aço devem ser cortadas em tiras, etc.

Não é permitido excesso de produção e os objetos em estoque devem ser cortados completamente, ou seja, não é permitido cortar uma parte de uma barra e devolver o restante para o estoque. Pedacos do corte que não sejam os itens demandados são considerados perda.

3. Modelagem Matemática

Definição 1. Chamamos de padrão de corte a maneira como um objeto em estoque é cortado para a produção dos itens demandados.

A um padrão de corte associamos um vetor m -dimensional que contabiliza os itens produzidos:

$$\mathbf{a} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T, \quad (3.1)$$

onde α_i é a quantidade de itens do tipo i no padrão de corte.

Observe que um vetor $\mathbf{a} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$ corresponde a um padrão de corte se, e somente se, satisfizer as restrições do Problema da Mochila (considerando apenas as restrições físicas (3.2) e (3.3))

$$\text{sujeito } a : l_1\alpha_1 + l_2\alpha_2 + \dots + l_m\alpha_m \leq L, \quad (3.2)$$

$$\alpha_i \geq 0, i = 1, \dots, m \text{ e } \textit{inteiros}. \quad (3.3)$$

Sejam

$$\mathbf{a}_1 = \begin{pmatrix} \alpha_{11} \\ \alpha_{21} \\ \vdots \\ \alpha_{m1} \end{pmatrix}, \quad \mathbf{a}_2 = \begin{pmatrix} \alpha_{12} \\ \alpha_{22} \\ \vdots \\ \alpha_{m2} \end{pmatrix}, \quad \dots, \quad \mathbf{a}_n = \begin{pmatrix} \alpha_{1n} \\ \alpha_{2n} \\ \vdots \\ \alpha_{mn} \end{pmatrix},$$

os vetores associados aos possíveis padrões de corte, onde α_{ij} é o número de vezes que o item de comprimento $l_i, i = 1, \dots, m$, aparece no padrão de corte j .

Seja x_j o número de vezes que o objeto é cortado usando o padrão j . O problema de corte de estoque, onde o total de objetos em estoque cortado deva ser minimizado, é dado por:

$$\begin{aligned} &\textit{minimizar} && f(\mathbf{x}) = x_1 + x_2 + \dots + x_n, \\ &\textit{sujeito } a : && \begin{cases} \mathbf{a}_1x_1 + \mathbf{a}_2x_2 + \dots + \mathbf{a}_nx_n = \mathbf{d}, \\ x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0 \text{ e } \textit{inteiros}. \end{cases} \end{aligned} \quad (3.4)$$

Em notação matricial o problema (3.4) é escrito como segue:

$$\begin{aligned} & \text{minimizar} && f(\mathbf{x}) = \sum_{j=1}^n x_j, \\ & \text{sujeito a:} && \mathbf{Ax} = \mathbf{d}, \quad \mathbf{x} \geq \mathbf{0} \quad \text{e} \quad \text{inteiro}, \end{aligned} \quad (3.5)$$

onde cada coluna da matriz $\mathbf{A} \in \mathbf{R}^{m \times n}$ é um vetor associado a um padrão de corte.

A condição de integralidade sobre as variáveis x_j torna o problema (3.5) difícil, senão impossível de ser resolvido computacionalmente, para problemas de dimensões moderadas. Em problemas práticos, m (que é a quantidade de tipos de itens) é da ordem de dezenas, enquanto que n (o qual depende de m, L e $l_i, i = 1, \dots, m$) pode ser da ordem de centenas de milhares, o que inviabiliza a resolução direta do problema.

Uma abordagem prática para resolver o problema (3.5) consiste em relaxar a condição de integralidade e resolver o problema relaxado (problema de otimização linear) pelo *Método Simplex* utilizando um processo de *Geração de Colunas*, proposto por Gilmore e Gomory [3], em 1961, de modo que não é preciso gerar todas as colunas antecipadamente. O procedimento começa com um conjunto de m padrões de corte homogêneos (colunas) que representam uma base da matriz \mathbf{A} . A cada iteração simplex, um destes padrões básicos é substituído por um padrão de corte que melhora a solução básica atual. Tal padrão (coluna) é determinado pela solução de um problema da mochila onde os coeficientes da função objetivo são os multiplicadores simplex da iteração atual (pequenas alterações ocorreriam se diferentes funções objetivos fossem usadas. Veja Gilmore e Gomory [4] e Arenales [1]). A solução ótima obtida para o problema (3.5) relaxado é, em geral, fracionária. Então, heurísticas têm sido desenvolvidas para determinar o arredondamento da solução. Serão descritos três métodos heurísticos para este arredondamento, sendo dois conhecidos na literatura e um novo (mais detalhes em Poldi [8]).

4. Critério para Solução Inteira Ótima

Segundo Wäscher e Gau [9], para analisarmos a qualidade das soluções encontradas pelas heurísticas é essencial que tenhamos bons limitantes para o valor da função objetivo. Seja

$$\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$$

a solução ótima para o problema de corte de estoque unidimensional (3.5), com a condição de integralidade relaxada (obtida pelo método simplex com geração de colunas). A função objetivo é dada por

$$f(\tilde{\mathbf{x}}) = \sum_{j=1}^n \tilde{x}_j.$$

Um *limitante inferior* para a função objetivo do problema (3.5) é dado por

$$LI = \lceil f(\tilde{\mathbf{x}}) \rceil,$$

isto é, o arredondamento do valor da função objetivo para o inteiro superior. Se x^* é uma solução ótima de (3.5) então $f(x^*) \geq LI$.

Seja \mathbf{x} uma solução para o problema (3.5). Se $f(\mathbf{x}) = \lceil f(\tilde{\mathbf{x}}) \rceil$ então podemos dizer que a solução inteira obtida é ótima. Assim, o limitante inferior LI nos dá um excelente ponto de referência para analisarmos a qualidade das soluções obtidas pelas heurísticas.

5. Métodos Heurísticos para Arredondamento da Solução

Os procedimentos que serão apresentados aqui geram uma solução inteira para o problema de corte de estoque unidimensional inteiro a partir da solução ótima do problema (3.5) relaxado. Supomos que pelo menos uma componente da solução ótima seja não-inteira, pois, caso contrário, o problema de corte de estoque unidimensional inteiro já estaria resolvido.

5.1. Problema Residual

O *Problema Residual* parte de uma solução ótima do problema (3.5) relaxado. Todos os componentes do vetor solução são arredondados para o número inteiro abaixo. Fazendo isto, teremos o vetor

$$\tilde{\mathbf{y}} = (\lfloor \tilde{x}_1 \rfloor, \lfloor \tilde{x}_2 \rfloor, \dots, \lfloor \tilde{x}_n \rfloor)$$

de inteiros. Esta solução não é uma solução factível pois, como foi arredondada para baixo, não completa a demanda, isto é, $\mathbf{A}\tilde{\mathbf{y}} \leq \mathbf{d}$. A correspondente função objetivo será dada por

$$f(\tilde{\mathbf{y}}) = \sum_{j=1}^n \lfloor \tilde{x}_j \rfloor.$$

Defina agora a demanda residual por: $\mathbf{r} = \mathbf{d} - \mathbf{A}\tilde{\mathbf{y}}$ e resolva o problema (3.5) novamente - chamado agora de *Problema Residual* - substituindo a demanda \mathbf{d} pela demanda residual \mathbf{r} . Uma nova solução (possivelmente fracionária) é obtida e um arredondamento para o interior inferior é novamente feito, gerando nova demanda residual. Repete-se o processo, até que o arredondamento gere soluções nulas. Se ainda restarem itens com demandas não atendidas, então teremos um último problema residual, que pode ser resolvido por alguma heurística que forneça uma solução inteira. Particularmente, utilizamos as heurísticas *FFD* e *Gulosa*, que serão descritas a seguir.

5.2. Heurística FFD

A heurística *FFD* (*First-Fit-Decreasing*) sempre encontra uma solução inteira para o problema de corte de estoque unidimensional. Ela consiste, de forma geral, em colocar o item maior num padrão de corte tantas vezes quanto for possível, ou seja, até que não se consiga mais colocar este item ou, até que sua demanda já tenha sido atendida. A seguir, colocar o segundo maior e assim por diante. Quando o último item (menor comprimento) for examinado, um padrão de corte foi construído. Este

padrão é usado tantas vezes quanto possível e a demanda é atualizada. Repete-se este procedimento até que toda a demanda seja atendida.

5.3. Heurística Gulosa

Analogamente à heurística *FFD*, a heurística *Gulosa* produz uma solução inteira ao ser aplicada ao problema residual. Ela consiste em resolver o Problema da Mochila com uma função objetivo apropriada, por exemplo, o *valor de utilidade* v_i de cada item na mochila como sendo o próprio comprimento do item (l_i), de modo que a perda no padrão é minimizada. Observe que a heurística *FFD* é um procedimento heurístico para resolver este problema da mochila.

$$\text{maximizar } g(\mathbf{y}) = l_1 y_1 + l_2 y_2 + \dots + l_m y_m, \quad (5.6)$$

$$\text{sujeito a : } l_1 y_1 + l_2 y_2 + \dots + l_m y_m \leq L, \quad (5.7)$$

$$y_i \geq 0, \quad i = 1, \dots, m \quad \text{e} \quad \text{inteiros.} \quad (5.8)$$

Então, a heurística gulosa consiste em, a cada iteração, resolver o problema acima para gerar um padrão de corte. Este padrão é usado tantas vezes quanto possível e atualiza-se a demanda. Repete-se este procedimento até que toda a demanda seja atendida.

5.4. Heurística Nova

Esta heurística, como as anteriores *FFD* e *Gulosa*, produz uma solução inteira. A cada iteração desta heurística resolve-se um problema de corte de estoque relaxado e ordena-se o vetor solução de forma decrescente, isto é, dá-se prioridade aos padrões que são mais utilizados. Para cada posição deste vetor, nesta ordem, arredonda-se a solução para o número inteiro acima do fracionário obtido e testa-se a factibilidade desta solução (no sentido de que excessos de itens não foram gerados). Caso não seja factível (isto é, houve excesso de itens), a solução é reduzida de uma unidade até que excessos sejam eliminados. Quando o último padrão de corte gerado for examinado, atualiza-se a demanda (problema residual). O problema residual é resolvido e o procedimento de arredondamento (para o inteiro superior ou inteiros inferiores) continua até que a toda a demanda seja satisfeita. Deve-se observar que a geração de colunas neste procedimento deve ser restrita, pois cada padrão de corte gerado pode ser utilizado pelo menos uma vez.

Observação:

A cada iteração, um novo padrão é determinado pela heurística, e fará parte da solução do problema, de modo que deve ser armazenado (por colunas) numa matriz, digamos, \mathbf{A} . A heurística pode gerar um padrão já existente na matriz \mathbf{A} . Assim, estabelecemos uma *ordem lexicográfica* para manter as colunas da matriz ordenadas de forma que, quando uma coluna tiver de ser inserida na matriz, saberemos onde inseri-la, caso seja ‘nova’, ou caso já exista, apenas atualizamos a coordenada de x correspondente, sem ter de percorrer toda a matriz.

6. Heurística de Redução de Padrões de Corte

O número de trocas de padrões de corte para atender a uma dada demanda pode afetar consideravelmente o tempo do processo de corte. Este é um caso onde o equipamento de corte tem de ser preparado a cada novo padrão de corte - um procedimento que consome um tempo durante o qual o processo de corte tem de ser interrompido. Em outras palavras, o tempo de preparação de máquina (*setup*), não considerado na construção dos padrões de corte, pode ser relevante na programação da produção.

Agora, apresentamos uma heurística para resolver o problema de redução de padrões em problemas de corte de estoque proposta por Limeira e Yanasse [7]. O método baseia-se na observação de que para se ter uma redução no número de padrões, deve-se impor que dois (ou mais) itens precisam ter suas demandas completadas quando a quantidade total de um padrão é cortada. Esta quantidade é o número de vezes que um padrão é repetido em uma solução. Outro requisito é que a eficiência do padrão gerado seja maior que a determinada na solução inicial inteira do problema.

A heurística proposta por *Limeira e Yanasse* [7] consiste basicamente em, a partir do problema original P , definir um problema P_1 com os mesmos itens do problema original P mas com as demandas ajustadas de acordo com um fator F_1 . O problema P_1 é obtido através da divisão de d_1, d_2, \dots, d_m por F_1 e tomando-se apenas a parte inteira. Qualquer padrão de corte viável para o problema P_1 é também viável para P e, além disso, este padrão pode ser cortado F_1 vezes sem que haja excesso de produção de um item.

Para o problema P_1 , um padrão é aceito se completar simultaneamente as demandas de dois ou mais itens contidos nele e, também, se possuir uma perda aceitável (nível de aspiração). Se estas duas condições forem satisfeitas, gera-se um padrão e ele fará parte da solução global do problema P com frequência F_1 .

Se nenhum padrão for escolhido, nenhuma redução será realizada. Um novo fator F_2 é definido e o processo é repetido. O processo termina quando o fator $F_k = 1$, para algum k , for alcançado e nenhum outro padrão alcançar o nível de aspiração desejado. Caso se tenha, no final, um problema residual, ou seja, demandas de alguns itens ainda não foram atendidas, então esse problema residual deve ser resolvido de maneira independente. No nosso caso, utilizamos as heurísticas descritas na seção 5.

Os fatores $F_1 \geq F_2 \geq \dots \geq 1$ são definidos da seguinte maneira: como os padrões desejados, quando cortados, precisam finalizar a demanda de pelo menos dois itens, as quantidades de tais padrões precisam ser submúltiplas das demandas de alguns itens. Portanto, os candidatos a fatores são números submúltiplos das demandas de todos os itens. Esses valores são ordenados de forma decrescente, excluindo-se os valores iguais, e são testados um a um.

Um dos níveis de aspiração para os padrões é que o desperdício (perda em cada padrão de corte) precisa ser mantido baixo, o que equivale a manter a *eficiência* do padrão (quanto está sendo aproveitado da barra segundo o padrão de corte considerado).

6.1. Exemplo

Consideremos o seguinte problema de corte de estoque unidimensional: temos em estoque uma quantidade ilimitada de barras de comprimento $L = 1000$ e desejamos atender a uma demanda de 10 itens, ou seja, $m = 10$. O comprimento de cada item $i, i = 1, \dots, 10$ e a sua respectiva demanda estão na Tabela 1.

Item	Comprimento	Demanda
1	10	6
2	62	18
3	115	7
4	132	6
5	140	3
6	158	11
7	161	12
8	172	18
9	176	8
10	178	11

Tabela 1: Dados do problema

Primeiramente, resolvemos o problema pelo método simplex com geração de colunas e obtivemos uma solução fracionária na qual:

Total de objetos cortados = 13,325

Número de padrões de corte = 10

A seguir, aplicamos uma heurística de arredondamento e encontramos uma solução na qual:

Total de objetos cortados = 14

Número de padrões de corte = 11

Eficiência média = 952 unidades

O resultado final obtido após a execução da heurística de redução de número de padrões de corte e do problema residual é o seguinte:

Solução:	Padrão de Corte:									
9	0	2	0	0	0	1	1	2	0	1
3	2	0	2	2	1	0	1	0	1	0
1	0	0	1	0	0	2	0	0	3	0
1	0	0	0	0	0	0	0	0	2	2

Total de objetos cortados = 14

Número de padrões de corte = 4

Eficiência média = 952 unidades

Podemos notar a grande redução de 11 para 4 padrões de corte, mantendo o

nível de eficiência. Eficiência média é quanto está sendo realmente aproveitado da barra (média do aproveitamento dos 4 padrões).

7. Testes Computacionais

Realizamos alguns testes computacionais para comparar os resultados obtidos pelas três heurísticas de arredondamento e a heurística de redução de padrões de corte. Para isto, geramos, aleatoriamente, 18 classes com 30 exemplos em cada classe, totalizando 540 exemplos. A seguir, mostramos em detalhes os critérios usados para gerar os dados e os resultados obtidos.

7.1. O Gerador Aleatório

Definimos o seguinte gerador aleatório, segundo Foerster e Wäscher [2]:

- *Dimensão da barra em estoque:* $L = 1000$.
- *Número de tipos de itens a serem produzidos:* foram utilizados diferentes valores de $m = 10, 20$ e 40 .
- *Dimensão dos itens a serem produzidos:* os comprimentos dos itens l_i foram gerados aleatoriamente nos intervalos $[v_1 L, v_2 L]$, onde v_1 e v_2 definem o menor e o maior valor, respectivamente, que $l_i, i = 1, \dots, m$ pode ter. Utilizou-se: $v_1 = 0.01$ e 0.2 e $v_2 = 0.2$ e 0.8 . Combinando estes valores, gerou-se classes com problemas *pequenos, médios e grandes*.
- *Demanda:* foram utilizados dois geradores diferentes para os valores da demanda dos itens; o primeiro com valores próximos a 10 e o segundo com valores próximos a 100.

Foram criadas 18 classes de problemas combinando L, m, d e gerados 30 exemplos para cada classe.

7.2. Análise dos Resultados

A Tabela 2 mostra a média dos 30 exemplos de cada classe. Nesta tabela podemos notar que a heurística *Nova*, em geral, resolve o problema de corte de estoque com um número menor de padrões de corte.

Com a aplicação da Heurística de Redução de Padrões de Corte as soluções encontradas apresentaram-se muito próximas para as três heurísticas de arredondamento, o que sugere que a Heurística de Redução de Padrões de Corte é pouco sensível à solução inicial. Entende-se por solução inicial, a solução do problema de corte de estoque inteira sem a aplicação da heurística de redução de padrões de corte.

A redução do número de padrões de corte fica prejudicada para algumas classes (como: C8, C14, C16) pois os itens de comprimentos médio e grande exigem que

				Solução Original			Solução Reduzida		
	m	l_i	d	<i>FFD</i>	Gulosa	<i>Nova</i>	<i>FFD</i>	Gulosa	<i>Nova</i>
C1	10	P	10	9,33	9,37	8,30	3,57	3,60	3,60
C2	10	P	100	14,07	14,03	12,90	10,80	10,83	10,13
C3	20	P	10	18,50	18,47	16,90	5,17	5,67	5,53
C4	20	P	100	27,43	27,40	25,30	15,20	15,57	13,90
C5	40	P	10	37,23	37,20	34,80	7,53	8,63	7,50
C6	40	P	100	54,80	54,77	51,50	40,50	38,23	30,00
C7	10	M	10	10,43	10,63	9,60	8,10	8,10	7,90
C8	10	M	100	11,93	11,90	11,93	12,43	12,50	12,07
C9	20	M	10	20,70	21,70	18,40	15,37	15,23	14,93
C10	20	M	100	24,27	25,37	22,77	21,80	21,93	21,87
C11	40	M	10	41,40	43,80	35,67	28,50	28,30	27,87
C12	40	M	100	48,20	51,10	43,77	39,13	39,30	39,17
C13	10	G	10	9,80	9,80	9,60	9,40	9,43	9,37
C14	10	G	100	10,77	10,77	11,27	11,43	11,43	11,43
C15	20	G	10	19,63	19,63	19,17	16,33	16,33	16,20
C16	20	G	100	21,83	22,00	22,30	22,60	22,70	22,53
C17	40	G	10	39,83	39,97	36,23	32,20	32,20	32,07
C18	40	G	100	44,37	44,50	42,73	42,13	42,13	42,13

Tabela 2: Número médio de padrões de corte

mais padrões sejam utilizados, não sendo, então, possível resolver o problema com um número menor de padrões de corte, tal como Haessler [5] afirma em seu artigo. Nestes poucos casos, nos quais observou-se um pequeno aumento no número de padrões de corte, ficamos com a solução original com o número menor de padrões.

Quanto ao valor da função objetivo, que é o número total de objetos a serem cortados ocorreu um pequeno aumento em relação ao número obtido na solução inicial. O aumento médio no número de objetos a serem cortados decorrido da heurística de redução de padrões de corte foi de 4,32% com a heurística *FFD*, 4,24% com a heurística Gulosa e 4,21% com a heurística *Nova*.

Quanto ao número de soluções ótimas - determinadas segundo o critério da seção 4. - encontradas na solução original foi de 95,7% para a heurística *FFD*, 88,5% para a heurística Gulosa e 92% para a heurística *Nova*. Com a aplicação da heurística de redução de padrões de corte esta porcentagem passou a 40,2% para a heurística *FFD*, 40,2% para a heurística Gulosa e 45% para a heurística *Nova*. Esta diminuição na quantidade de soluções ótimas deve-se ao aumento no número de objetos a serem cortados comentado anteriormente.

O tempo computacional médio para cada problema teste é de 2,2 segundos para a heurísticas *FFD*, 2,3 segundos para a heurística Gulosa e 1 segundo para a heurística *Nova*. Os testes foram realizados em um computador Pentium III (866 MHz) com 256 MB de memória RAM.

8. Conclusão e Perspectivas Futuras

Concluimos que a heurística de redução mostrou-se muito eficiente, pois para a grande maioria das classes obteve redução do número de padrões de corte. Apesar da heurística *Nova* já iniciar com um número de padrões menor, a Heurística de Redução mostrou-se pouco sensível à solução inicial, conforme também observado por Foerster e Wäscher [2]. Um trabalho futuro consiste em estender as heurísticas para o caso bidimensional.

Abstract. This paper addresses the problem of generating integer solutions to the one-dimensional cutting stock problem and, in a second step, a reduction in the number of patterns. We study some heuristic approaches proposed in literature and some straightforward variants. These methods are compared with respect to solution quality and number of cutting patterns. Our evaluation is based on solving 540 randomly generated test problems.

Referências

- [1] M.N. Arenales, Modelos e Métodos Básicos dos problemas de corte, em “XX CNMAC”, Capítulo I do Livro-texto “O Problema de Corte e Empacotamento e Aplicações Industriais” de Mini curso editado por M.N. Arenales e R. Marabito, Gramado, RS, 1997.
- [2] H. Foerster, G. Wäscher, “Pattern reduction in one-dimensional cutting-stock problems”, *International Journal of Production Research*, **38** (2000), 1657-1676.
- [3] P.C. Gilmore, R.E. Gomory, “A linear programming approach to the cutting stock problem”, *Operations Research*, **9** (1961), 848-859.
- [4] P.C. Gilmore, R.E. Gomory, “A linear programming approach to the cutting stock problem - Part II”, *Operations Research*, **11** (1963), 863-888.
- [5] R.W. Haessler, “Controlling cutting pattern changes in one-dimensional trim loss problems”, *Operations Research*, **23** (1975), 483-493.
- [6] P. Lefrançois, A. Gascon, “Solving a one-dimensional cutting-stock problem in a small manufacturing firm: a case study”, *IIE Transactions*, **27** (1995), 483-496.
- [7] M.S. Limeira, H.H. Yanasse, Uma heurística para o problema de redução de padrões de corte, em “Anais da V Oficina Nacional de Problemas de Corte e Empacotamento”, São José dos Campos, SP, pp. 137-145, 2001.
- [8] K.C. Poldi, Heurísticas para o Problema de Corte de Estoque Unidimensional Inteiro, em “Anais do XXXIV SBPO”, Rio de Janeiro, RJ, 2002.
- [9] G. Wäscher, T. Gau, “Heuristics for the integer one-dimensional cutting stock problem: a computational study”, *OR Spektrum*, **18** (1996), 131-144.