

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Uma abordagem não-linear para o problema de corte unidimensional bi-objetivo

Angelo Aliano Filho¹

Departamento Acadêmico de Matemática, Universidade Tecnológica Federal do Paraná – UTFPR

Luiz Antônio Farani de Souza²

Departamento de Engenharia Civil, Universidade Tecnológica Federal do Paraná – UTFPR

Antonio Carlos Moretti³

Instituto de Matemática, Estatística e Computação Científica, IMECC – UNICAMP

Resumo.

Neste trabalho, resolvemos o problema de corte de estoque unidimensional inteiro bi-objetivo mediante uma abordagem não-linear do mesmo. Grande volume de trabalhos resolve este problema fornecendo as combinações de corte *a priori*, gerando soluções sub-ótimas (caso mono-objetivo) ou soluções sub-eficientes (no caso multi-objetivo). Nesse sentido, determinamos soluções eficientes para este problema gerando implicitamente os padrões de corte em cada passo. Para isto, linearizamos a formulação empregando “truques” de programação inteira, permitindo determinar o conjunto eficiente em instâncias de pequeno e médio porte em um tempo computacional inferior à 1 hora.

Palavras-chave. Pesquisa Operacional, Otimização Multi-objetivo, Problema de Corte.

1 Introdução

O Problema de Corte de Estoque Unidimensional Inteiro é um dos problemas combinatórios mais estudados dentro do campo da otimização combinatória, devido, principalmente, a sua aplicabilidade no mundo da engenharia de produção, fazendo parte do planejamento de uma diversidade de indústrias que cortam papel, madeira, vidro, plásticos, tecido, entre outras matérias-primas. Este problema tem um elevado nível de complexidade, sendo classificado na literatura como \mathcal{NP} -Completo, de acordo com [5].

Este problema pode ser entendido da seguinte forma: as matérias-primas a serem utilizadas são estocadas em forma de grandes rolos ou bobinas-mestre de larguras padronizadas, e apenas mais tarde são reduzidas a tamanhos menores e variados (itens) para serem usadas pela indústria de acordo com as demandas externas. No entanto, esse procedimento causa o inconveniente de preparar as máquinas para a operação de corte destes materiais, ocasionando, inevitavelmente, perdas, ajustes de máquinas e um aumento no tempo de

¹angeloaliano@utfpr.edu.br

²lasouza@utfpr.edu.br

³moretti@ime.unicamp.br

produção com sucessivas paradas no maquinário. Desse modo, há necessidade de planejar a maneira de efetuar estes cortes, a fim de minimizar o desperdício e efeitos negativos ocasionados com esta operação.

É útil incorporar ao problema de corte o *setup* de máquina. Ele está associado à cada troca de um padrão de corte para outro. Do ponto de vista operacional, o *setup* envolve uma série de reparações e parada nas máquinas para que as facas possam ser trocadas para efetuar outros cortes, e constitui um importante aspecto a ser levado em conta.

Neste presente estudo, damos um enfoque bi-objetivo a este problema, no tocante em obtermos soluções eficientes (no senso de Pareto), estabelecendo um compromisso entre (i) o desperdício ocasionado com os cortes e (ii) o *setup* de máquina. Além disto, pretendemos abordá-lo de maneira não-linear, no sentido do próprio modelo matemático gerar os padrões de corte conforme as soluções eficientes forem determinadas. Na literatura, há trabalhos, como em [6], [4], que fizeram esta consideração não-linear deste problema, porém, não o abordaram como bi-objetivo (no senso de determinar diferentes soluções ótimas de Pareto) e nem fizeram o uso de métodos exatos para sua resolução. O trabalho em [2] usou a abordagem multi-objetiva mas usou um algoritmo genético especializado. O trabalho recente em [1], considera o problema de corte resolvendo instâncias de dimensão mais elevada, mas gerando *a priori* os padrões de corte para posteriormente determinar as soluções eficientes.

Os procedimentos matemáticos necessários para resolver este problema são descritos na próxima seção.

2 Modelagem matemática e procedimento de linearização

Considere um rolo-mestre em estoque de largura L (largura padrão) e m o número de itens demandados. Cada item i tem largura $\ell_i < L$ e ao menos d_i unidades precisam ser produzidas a fim de atender a demanda, $i \in \{1, \dots, m\}$. Os objetivos consistem em minimizar o desperdício da produção e o número de diferentes padrões de cortes utilizados.

As duas definições a seguir são importantes para apresentarmos a formulação matemática deste problema.

Definição 2.1. *Uma instrução que lista os itens demandados a serem cortados do rolo-mestre é chamada de padrão de corte, que pode ser associado ao vetor m -dimensional:*

$$\mathbf{a}_j = (a_{1j}, a_{2j}, \dots, a_{mj})^T,$$

onde cada entrada $a_{ij} \in \mathbb{Z}_+$, denota a quantidade do item i presente no padrão $j \in \{1, \dots, n\}$. O número n denota o número total de combinações de corte que podem ser feitas usando o rolo-mestre de largura L .

Definição 2.2. *Um padrão de corte $\mathbf{a}_j \in \mathbb{Z}_+^m$ é admissível se satisfaz as seguintes condições:*

$$\begin{aligned} L - \delta &\leq \sum_{i=1}^m \ell_i \cdot a_{ij} \leq L, & j = 1, \dots, n, \\ \sum_{i=1}^m a_{ij} &\leq f^{max}, & j = 1, \dots, n, \end{aligned} \tag{1}$$

onde $\delta = \min_{1 \leq i \leq m} \{\ell_i\}$ (cortando até que o desperdício seja igual a largura do menor item) e f^{max} é o número máximo de facas permitido no maquinário.

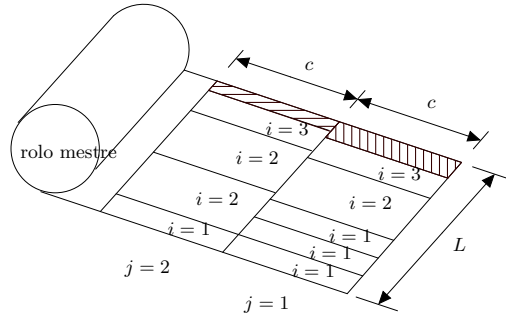


Figura 1: Exemplo de dois padrões de cortes

Na Figura 1, ilustramos um rolo-mestre de largura L , quatro itens demandados e dois padrões de cortes. No padrão $j = 1$, temos três vezes o item 1 sendo cortado, uma vez os itens 2 e 3 e nenhuma vez o item quatro, logo ele pode ser representado pelo vetor $\mathbf{a}_1 = (3, 1, 1, 0)^T$. Um outro padrão de corte poderia ser dado pelo vetor $\mathbf{a}_2 = (1, 2, 1, 0)^T$, em que os itens 1 e 3 são cortados uma só vez e o item 2 sendo cortado duas vezes.

Nesta formulação, partimos do princípio que os padrões de corte não são fornecidos *a priori* pelo usuário, como muitos trabalhos na área supõem, como em [3] e [1]. Definimos como variáveis decisórias o seguinte:

- $a_{ij} \in \mathbb{Z}_+$: número de vezes que o item i está presente no padrão de corte j , $i \in \{1, \dots, m\}$ e $j \in \{1, \dots, 2m\}$;
- $x_j \in \mathbb{Z}_+$: número de vezes que o padrão de corte j é utilizado;
- $y_j = \begin{cases} 1, & \text{se } x_j > 0 \\ 0, & \text{caso contrário.} \end{cases}$

A formulação bi-objetiva não-linear inteira que visa minimizar o desperdício e a troca de padrões de cortes distintos, é apresentada a seguir.

$$\text{minimize } z_1 = \sum_{j=1}^{2m} \left(L - \sum_{i=1}^m \ell_i \cdot a_{ij} \right) \cdot x_j \tag{2}$$

$$\text{minimize } z_2 = \sum_{j=1}^{2m} y_j \tag{3}$$

$$\text{sujeito a } \sum_{j=1}^{2m} a_{ij} \cdot x_j \geq d_i, \quad i = 1, \dots, m, \tag{4}$$

$$L - \delta \leq \sum_{i=1}^m \ell_i \cdot a_{ij} \leq L, \quad j = 1, \dots, 2m, \tag{5}$$

$$\sum_{i=1}^m a_{ij} \leq f^{max}, \quad j = 1, \dots, 2m, \tag{6}$$

$$y_j \leq x_j \leq N \cdot y_j, \quad j = 1, \dots, 2m, \tag{7}$$

$$a_{ij} \in \mathbb{Z}_+, \quad x_j \in \mathbb{Z}_+, \quad y_j \in \{0, 1\}, \quad i = 1, \dots, m, \quad j = 1, \dots, 2m \tag{8}$$

Nela, impusemos um limitante superior de $n = 2m$ para o número máximo de distintos padrões de corte. Nos resultados computacionais veremos que esta escolha foi razoável.

A função z_1 visa minimizar o desperdício total. O termo entre parêntesis em (2) é o desperdício no padrão j , que deve ser multiplicado pela sua respectiva frequência x_j e ser somado para todo j . A função objetivo (3) descreve a minimização da troca de padrões de cortes. As restrições (4) determinam o atendimento da demanda para cada item i e as restrições em (5) e (6) definem um padrão factível de acordo com a Definição 2.2. As restrições (7) ligam as variáveis x_j e y_j , lembrando que N é um limitante superior para x_j , que pode ser dado por $N = \max\{d_i | i = 1, \dots, m\}$. Finalmente, (8) determinam o domínio das variáveis decisórias deste problema.

Para linearizar o modelo (2)–(8), tanto na primeira função objetivo quanto nas restrições de demanda, procedemos do seguinte modo. Consideramos a decomposição binária de a_{ij} e assumimos que estas variáveis possuam limitantes superiores M_i , dados por:

$$M_i = \min \left\{ \left\lfloor \frac{L}{\ell_i} \right\rfloor, f^{max} \right\}, \quad i = 1, \dots, m. \quad (9)$$

Então, escrevemos a_{ij} em função de sua decomposição em variáveis binárias, β_{ijk} , do seguinte modo:

$$a_{ij} = \sum_{k=1}^{K_i} 2^{k-1} \cdot \beta_{ijk}, \quad i = 1, \dots, m, \quad j = 1, \dots, 2m, \quad (10)$$

onde $K_i = \lceil \log_2 M_i \rceil + 1$. Feito isto, a próxima etapa consiste em substituir a bilinearidade $\mathbb{R}_+ \ni z_{ijk} = \beta_{ijk} \cdot x_j$, onde z_{ijk} são novas variáveis que aparece quando substituimos as expressões (10) nas restrições (4). Para isto, necessitamos inserir restrições lineares que conectam z_{ijk} com x_j e β_{ijk} . Tais restrições são descritas por:

$$z_{ijk} \leq N \cdot \beta_{ijk}, \quad (11)$$

$$z_{ijk} \leq x_j, \quad (12)$$

$$x_j - N \cdot (1 - \beta_{ijk}) \leq z_{ijk}, \quad (13)$$

válidas para todo $i = 1, \dots, m$, $j = 1, \dots, 2m$, e $k = 1, \dots, K_i$. Substituindo as variáveis a_{ij} , por meio das equações (10) nas condições (1), na função objetivo (2), em seguida mudando para as variáveis z_{ijk} , que levarão consigo os conjuntos de restrições (11)–(13) na nova formulação, após algumas manipulações algébricas, teremos uma versão linearizada do problema (2)–(8), dada por (14)–(21).

$$\text{minimize } z_1 = L \cdot \sum_{j=1}^{2m} x_j - \sum_{i=1}^m \sum_{j=1}^{2m} \sum_{k=1}^{K_i} 2^{k-1} \cdot \ell_i \cdot z_{ijk} \quad (14)$$

$$\text{minimize } z_2 = \sum_{j=1}^{2m} y_j \quad (15)$$

$$\text{sujeito a } \sum_{j=1}^{2m} \sum_{k=1}^{K_i} 2^{k-1} \cdot z_{ijk} \geq d_i, \quad i = 1, \dots, m, \quad (16)$$

$$L - \delta \leq \sum_{i=1}^m \sum_{k=1}^{K_i} 2^{k-1} \cdot \ell_i \cdot \beta_{ijk} \leq L, \quad j = 1, \dots, 2m, \quad (17)$$

$$\sum_{i=1}^m \sum_{k=1}^{K_i} 2^{k-1} \cdot \beta_{ijk} \leq f^{max}, \quad j = 1, \dots, 2m, \quad (18)$$

$$y_j \leq x_j \leq N \cdot y_j, \quad j = 1, \dots, 2m, \quad (19)$$

$$\text{restrições (11) - (13)} \quad j = 1, \dots, 2m, \quad (20)$$

$$\beta_{ijk} \in \{0, 1\}, \quad y_j \in \{0, 1\}, \quad x_j \in \mathbb{Z}_+, \quad z_{ijk} \in \mathbb{R}_+, \quad i = 1, \dots, m, \quad j = 1, \dots, 2m, \quad k = 1, \dots, K_i. \quad (21)$$

A próxima seção descreve um método exato para gerar soluções eficientes a partir da formulação linear (14)–(21).

3 Aplicação do método ε –Restrito

O método ε –Restrito aplicado neste problema e que descrevemos nesta seção, foi originalmente proposto em [1]. É um procedimento que pode ser usado para gerar um conjunto de soluções eficientes para o problema em estudo. A escolha do mesmo está atrelada à facilidade na sua implementação e gera tanto soluções eficientes suportadas como não-suportadas. Primeiramente, otimizamos individualmente as funções z_1 e z_2 , isto é, determinamos as soluções ótimas dos seguintes problemas:

$$\mathbf{x}_1^* = \operatorname{argmin}\{z_1 + 0.01z_2 : \text{sa (16)–(21)}\} \quad \text{e} \quad \mathbf{x}_2^* = \operatorname{argmin}\{z_2 + 0.01z_1 : \text{sa (16)–(21)}\},$$

onde os termos $0.01z_1$ ou $0.01z_2$ são para evitarem múltiplas soluções. Sejam $\varepsilon^{min} = z_2(\mathbf{x}_2^*)$ e $\varepsilon^{max} = z_2(\mathbf{x}_1^*)$ os valores de máximo e mínimo da segunda função objetivo. Feito isto, consideramos o problema restrito:

$$\min\{z_1 + 0.01 \cdot z_2 | \text{s.a. (16) - (19) e } z_2 \leq \varepsilon\}, \quad (22)$$

onde ε assume valores inteiros no intervalo $\mathcal{I} = [z_2^-, z_2^+]$. A cada escolha de ε em \mathcal{I} , uma solução eficiente é determinada.

4 Resultados computacionais

Os os subproblemas restritos (22) foram resolvidos pelo CPLEX 12.5 em um computador Intel(R) Core(TM) i3 M350 de 2.27GHz, com 2 GB de memória em um sistema operacional Windows 7 Professional de 32 Bits. Os valores para ℓ_i foram gerados aleatoriamente no intervalo $[0.2 \cdot L, 0.6 \cdot L]$ enquanto os dados para as demandas foram gerados entre $[300, 1000]$.

Resolvemos 4 instâncias para este problema, considerando m igual a 5, 10, 15 e 20 itens. O número de facas máximo foi fixo igual a $f^{max} = \left\lceil \sum_{i=1}^m \frac{L}{m \cdot \ell_i} \right\rceil$. Em cada instância, apresentamos os valores do do excesso de produção (Exc.), desperdício (Desp.), quantidade de padrões de corte (Σx_j) e número de diferentes padrões de cortes (Σy_j). Além disso, devido a dificuldade em resolver cada sub-problema (22), nós apenas resolvemos para $\bar{\varepsilon} = \lceil \frac{1}{2}(\varepsilon^{max} + \varepsilon^{min}) \rceil$. Sendo assim, para cada instância, foram determinadas três distintas soluções eficientes. Estes resultados estão presentes na Tabela (1).

Tabela 1: Resultados computacionais para excesso de produção, desperdício, número de padrões cortados e *setup* para as quatro instâncias resolvidas

	$m = 5$				$m = 10$				$m = 15$				$m = 20$			
ε	Exc.	Desp.	Σx_j	Σy_j	Exc.	Desp.	Σx_j	Σy_j	Exc.	Desp.	Σx_j	Σy_j	Exc.	Desp.	Σx_j	Σy_j
ε^{min}	106	2198	355	3	196	2990	526	5	252	4142	704	6	210	4103	866	9
$\bar{\varepsilon}$	9	2150	331	4	36	2765	483	8	82	1668	618	10	25	2582	771	14
ε^{max}	1	1897	329	5	0	2603	474	10	6	1439	600	13	1	2563	763	18

Os resultados mostram que o Problema (14)-(21) possui objetivos conflitantes, de fato. Conforme o *setup* de máquina aumenta, o desperdício, número de padrões de cortes utilizados e excesso de produção diminuem consideravelmente. Vemos que quando o *setup* é máximo, praticamente o excesso de produção se anula bem como o desperdício com as perdas fica minimizado. Isso porque há vários padrões de cortes que se “ajustam” com sua respectiva frequência, ocasionando pouco excesso.

O modelo linearizado tem a capacidade de ir gerando os padrões de corte implicitamente conforme o objetivo do problema. A Tabela (2) ilustra os padrões de cortes e suas respectivas frequências, para a instância $m = 5$, sendo modificados conforme a variação de ε . Note que os padrões $(0, 0, 1, 1, 0)^T$ e $(1, 1, 1, 0, 0)^T$ estão presentes em todas as soluções eficientes.

Finalmente, apresentamos na Tabela (3) os tempos CPU (em segundos) necessários para otimizar cada Problema (22) e o número de variáveis e restrições da formulação linearizada. Até a instância com $m = 15$ itens, no máximo 30 minutos foram utilizados pelo CPLEX para otimizá-los. A instância com $m = 20$ itens com $\varepsilon = 14$ e 18 não foi otimizada com 1 hora de execução, sendo os *gaps* de integralidade iguais a 11% e 19%. É interessante notar que o objetivo z_2 é muito mais fácil de ser otimizado em relação ao objetivo z_1 . Uma estratégia em reduzir estes tempos computacionais seria em considerar no máximo m padrões de corte na formulação, em vez de $2m$. Isso faria com que o número de variáveis e restrições da formulação (14)-(21) reduzisse à metade. Porém, o agravante é que não garantiríamos o mínimo global de z_1 , pois este mínimo global poderia usar mais que m padrões de cortes em sua confecção.

Tabela 2: Padrões de cortes determinados na instância com $m = 5$

ε	j	Padrão	x_j
ε^{min}	1	$(0, 0, 1, 1, 0)^T$	103
	2	$(1, 1, 0, 0, 1)^T$	163
	3	$(1, 1, 1, 0, 0)^T$	89
$\bar{\varepsilon}$	1	$(0, 0, 2, 0, 0)^T$	25
	2	$(0, 0, 0, 0, 1)^T$	163
	3	$(0, 0, 1, 1, 0)^T$	103
	4	$(1, 1, 1, 0, 0)^T$	40
ε^{max}	1	$(2, 0, 1, 0, 0)^T$	4
	2	$(0, 1, 0, 0, 1)^T$	163
	3	$(0, 0, 1, 1, 0)^T$	103
	4	$(0, 0, 2, 0, 0)^T$	27
	5	$(1, 1, 1, 0, 0)^T$	32

Tabela 3: Número de variáveis, restrições e tempo de CPU

	ε	# var.	# restr.	CPU
$m = 5$	ε^{min}			23,32
	$\bar{\varepsilon}$	220	355	186,45
	ε^{max}			489,23
$m = 10$	ε^{min}			87,36
	$\bar{\varepsilon}$	880	1370	322,14
	ε^{max}			987,21
$m = 15$	ε^{min}			103,22
	$\bar{\varepsilon}$	2160	3315	655,21
	ε^{max}			1879,17
$m = 20$	ε^{min}			154,41
	$\bar{\varepsilon}$	3440	5260	3600,00
	ε^{max}			3600,00

5 Conclusões

O trabalho mostrou que o problema de corte não-linear unidimensional bi-objetivo pode ser resolvido usando-se procedimentos exatos de programação matemática, bastando linearizar a formulação e usando estratégias básicas de modelagem em programação inteira. Soluções globalmente eficientes para instâncias de até $m = 15$ itens podem ser determinadas em um tempo computacional razoável (≈ 30 minutos/solução eficiente). Além disso, esta abordagem permite que possamos garantir a otimalidade global das soluções eficientes.

Referências

- [1] A. F. Aliano, A. C. Moretti, and M. V. Pato, A comparative study of exact methods for the bi-objective integer one-dimensional cutting stock problem, *Journal of the Operational Research Society*, 69(1), 91-107, 2018.
- [2] R. R. Golfeto, A. C. Moretti, and L. L. D. Salles Neto, A genetic symbiotic algorithm applied to the one-dimensional cutting stock problem, *Pesquisa Operacional*, 29(2), 365-382, 2009.
- [3] R. W. Haessler, Controlling cutting pattern changes in one-dimensional trim problems, *Operations Research*, 23(3), 483-493, 1975.
- [4] I. Harjunkoski, T. Westerlund, R. Porn, and H. Skrifvars, Different transformations for solving non-convex trim-loss problems by MINLP, *European Journal of Operational Research*, 105(3), 594-603, 1996.
- [5] D. S. Hochbaum, approximation algorithms for NP-hard problems, *PWS Publishing Co*, 1996.
- [6] A. C. Moretti, L. L. D. Salles Neto, Nonlinear cutting stock problem model to minimize the number of different patterns and objects, *Computational & Applied Mathematics*, 27(1), 61-78, 2008.