Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 8, n. 1, 2021.

Trabalho apresentado no XL CNMAC, Evento Virtual - Co-organizado pela Universidade do Mato Grosso do Sul (UFMS).

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Uma modificação na técnica de geração colunas para o problema de corte unidimensional

Carla T. L. S. Ghidini[1]
FCA - UNICAMP, Limeira, SP
Aurelio R. L. Oliveira[2]
IMECC/UNICAMP, Campinas, SP
Antonio C. Moretti[3]
IMECC/UNICAMP, Campinas, SP

Resumo. O problema corte unidimensional já foi e continua sendo bastante estudado devido à sua ampla aplicação, principalmente, no setor industrial. Este problema, classificado como NP-difícil, pode ser representado matematicamente por um modelo de programação linear inteira, o qual, geralmente, é resolvido por meio de métodos heurísticos. Um método heurístico muito conhecido e utilizado para a sua resolução é o método simplex com geração de colunas. Este método considera, inicialmente, um conjunto limitado de colunas, que representam os padrões de corte e a cada iteração um novo padrão de corte é gerado resolvendo um problema da mochila, cujos coeficientes de função objetivo são as variáveis duais do problema de corte relaxado (sem a restrição de integralidade das variáveis). Neste trabalho, propomos utilizar o centro analítico da face ótima do politopo dual ao invés das variáveis duais do método simplex para construir o problema da mochila e com isso acelerar a convergência do método. Os experimentos computacionais realizados mostraram bons resultados para essa abordagem, especialmente quando o problema primal é altamente degenerado.

Palavras-chave. Problema de Corte Unidimensional, Geração de Colunas, Método de Pontos Interiores

1 Introdução

O problema de corte consiste em determinar a melhor forma de cortar a matéria-prima (objetos) em itens de dimensões menores, a fim de satisfazer as demandas em carteira de clientes otimizando perdas e/ou quantidade de objetos cortados. Neste trabalho, tratamos dos problemas de corte unidimensionais que surgem em diversos tipos de indústrias de manufatura e, em particular, na indústria de papel, onde as bobinas jumbos (objetos) devem ser cortadas em bobinas menores que serão enviadas para outra etapa do processo ou estocadas até a data de entrega.

Um padrão de corte é um arranjo de itens no objeto. Há várias maneiras de se obter estes arranjos e para se ter uma ideia do número de padrões de corte que podem ser gerados, a partir de itens de tamanho l_i e um objeto de tamanho L, basta gerar todas as possíveis combinações dos itens de diferentes larguras que respeitam a inequação, $\sum_{i=1}^{m} a_i l_i \leq L$, em que a_i é o número de vezes que o item i aparece no padrão de corte que será usado para cortar o objeto e m é a quantidade de tipos de itens. Dependendo dos tamanhos do objeto e dos itens a serem cortados

[1]cghidini@unicamp.br.
[2]aurelio@po.ime.unicamp.br.
[3]amoretti@unicamp.br.

este número pode chegar à casa dos milhões, mesmo para uma carteira de pedidos com um número pequeno de tipos de itens (veja [2] para mais detalhes).

O problema de corte é um problema de programação linear inteira e classificado como do tipo NP-Difícil [8]. Geralmente, ele é resolvido por um procedimento heurístico e, por isso, não há garantia de que a solução ótima seja encontrada, mas soluções de boa qualidade para o problema a um tempo computacional aceitável são produzidas.

Na literatura, existem inúmeros trabalhos que tratam do problema de corte, em particular do caso unidimensional, e suas principais dificuldades na resolução (integralidade e grande número de variáveis). Foram desenvolvidos tantos métodos exatos [3] quanto heurísticos [9], de forma a obter uma boa solução para o problema em um tempo computacional aceitável. Dentre estes métodos destacamos o método simplex com geração de colunas [5], as heurísticas do tipo construtivas ([9],[6]), as quais, de forma geral, consistem em construir um padrão de corte com a menor perda possível e, posteriormente, calcular o número máximo de vezes que esse padrão pode ser usado, sem exceder a demanda dos itens e as heurísticas do tipo residuais ([7],[1]), as quais iniciam o procedimento resolvendo o problema relaxado (sem a restrição de integralidade das variáveis) pelo método simplex com geração de colunas, obtendo uma solução que pode ser fracionária e, posteriormente, alguma técnica é usada para tornar a solução inteira.

O objetivo desse trabalho é acelerar a convergência do método simplex com geração de colunas, principalmente para problemas que são altamente degenerados. Para isso, o problema da mochila é construído não utilizando as variáveis duais associadas à solução primal do método simplex, mas sim os multiplicadores de Lagrange vindos da solução do método de pontos interiores do tipo primal-dual. Essa proposta é justificada pelo fato de que se a solução do problema de programação linear não é única (primal ou dual), o algoritmo converge para um ponto interior do conjunto solução do problema. Com essa abordagem, o algoritmo convergirá para o centro analítico da face ótima do poliedro dual (pois existem soluções alternativas no dual). Dessa forma, o corte gerado pela inserção da coluna obtida pelo problema da mochila, cujos coeficientes da função-objetivo são os multiplicadores de Lagrange, provocará um bom corte na face ótima do poliedro dual, uma vez que eles representam o centro analítico da face e não um ponto extremo qualquer.

O restante do texto está organizado da seguinte forma: na Seção 2, descrevemos o problema de corte unidimensional, apresentamos um modelo matemático clássico que o representa e mostramos com qual modelo matemático trabalhamos, já que a melhoria obtida com a modificação proposta fica mais evidente em problemas degenerados. Na Seção 3, descrevemos brevemente o método simplex com geração de colunas para resolver problemas de corte unidimensional, bem como introduzimos a nossa ideia de modificar este método para reduzir o número de novos padrões de corte gerados. A Seção 4 traz a descrição de como as instâncias de problemas de corte usadas nos experimentos computacionais foram geradas, além de uma análise dos resultados obtidos ao comparar a abordagem clássica com a nossa proposta. Na Seção 5, estão as conclusões do trabalho e também algumas observações. Na última seção apresentamos as referências bibliográficas utilizadas.

2 Problema de Corte Unidimensional

O problema de corte unidimensional pode ser descrito da seguinte maneira: considere um conjunto de carteiras de pedidos com vários itens, as quais podem ter itens em comum. Juntando todas estas carteiras de pedidos tem-se m tipos de itens, em que o item tipo i, para $i = 1, ..., m$ tem comprimento l_i e demanda d_i, ou seja, d_i é a soma das demandas do item tipo i nas carteiras de pedidos. Os padrões de corte podem ser representados como colunas de uma matriz A, em que a_{ij} significa quantas vezes o item tipo i aparece no padrão de corte j. Usando os parâmetros descritos acima em conjunto com as variáveis de decisão, x_j para $j = 1, 2, ..., n$, que indicam quantos objetos serão cortados de acordo com o padrão de corte j, uma representação matemática deste problema é o modelo de programação linear inteira apresentado a seguir, o qual foi proposto em [5].

$$\begin{array}{ll} \text{Minimizar} & \displaystyle\sum_{j=1}^{n} x_j \\ \text{sujeito a} & \displaystyle\sum_{j=1}^{n} a_{ij}x_j \geq d_i \quad i = 1, 2, \ldots, m \\ & x_j \geq 0 \text{ e inteiro} \quad j = 1, 2, \ldots, n \end{array} \tag{1}$$

No Problema 1, o objetivo é minimizar o número de objetos cortados sujeito a um conjunto de restrições de atendimento a demanda dos itens.

Para garantir que as soluções primais sejam degeneradas e assim a abordagem proposta nesse trabalho seja melhor testada, consideramos o seguinte problema:

$$\begin{array}{ll} \text{Minimizar} & p_1\displaystyle\sum_{j=1}^{n} x_j + p_2\sum_{j=1}^{n} y_j \\ \text{sujeito a} & \displaystyle\sum_{j=1}^{n} a_{ij}x_j = d_i \quad i = 1, 2, \ldots, m \\ & x_j \geq y_j \quad j = 1, 2, \ldots, n \\ & x_j \leq My_j \quad j = 1, 2, \ldots, n \\ & x_j \geq 0 \text{ e inteiro} \quad j = 1, 2, \ldots, n \\ & y_j \in \{0, 1\} \quad j = 1, 2, \ldots, n \end{array}, \tag{2}$$

em que $y_j = \begin{cases} 1 & \text{se o padrão } j \text{ for usado;} \\ 0 & \text{caso contrário.} \end{cases}$

Este problema representa a minimização do desperdício e do *setup* de máquina. Os pesos p_1 e p_2 são constantes fornecidas pelo usuário para indicar a importância que cada objetivo tem para ele. Se o *setup* e o desperdício tem a mesma importância para o usuário, ele fixará os pesos $p_1 = p_2 = 0,5$. Caso o usuário considere o desperdício mais importante então ele fixará $p_1 > p_2$.

Vale ressaltar que o *setup* e o desperdício são objetivos conflitantes, pois se o objetivo é minimizar o desperdício então a tendência é gerar vários itens em vários padrões de corte e a solução ótima poderá conter m padrões de corte. Neste caso, o número de *setups* também será m. Porém, ao incorporar o termo que representa o *setup*, a solução ótima tende a ter variáveis básicas no nível zero (padrões não usados) e, consequentemente, o número de *setups* tende a ser menor do que m.

Um procedimento heurístico que pode ser usado para resolver problemas dos tipos (1) e (2) é o método simplex com geração de colunas descrito brevemente na Seção 3.

3 Método Simplex com Geração de Colunas

O método simplex com geração de colunas trabalha com o problema relaxado (sem as restrições de integralidade das variáveis). Logo, a solução ótima conterá no máximo m variáveis com valores maiores que zero. Dessa forma, como apenas um número pequeno do total de possíveis padrões pertencerá à solução final com valores diferentes de zero, Gilmore e Gomory [5] propõe trabalhar com a relaxação do problema (1) usando a ideia do método simplex revisado. O método começa com o problema (1) relaxado formado com apenas m padrões de corte homogêneos (padrões que contém apenas um tipo de item em sua composição) e após resolvê-lo até a otimalidade, as variáveis duais ótimas são usadas para obter uma nova coluna (padrão de corte) que, se for lucrativa será adicionada ao problema. O processo de geração de colunas e re-otimização do problema ampliado continua até que não haja mais colunas lucrativas ou seja identificado que o problema não tenha uma solução finita. Diferentemente do método simplex revisado, que usa as variáveis duais da base atual para calcular o custo reduzido das variáveis não-básicas, a técnica proposta por Gilmore e Gomory resolve um problema da mochila para obter o custo reduzido da nova coluna (variável).

O grande problema do método simplex com geração de colunas ocorre quando existem soluções degeneradas, pois apesar do problema gerar uma coluna lucrativa, esta pode não melhorar o valor da função-objetivo, pelo fato de que ao entrar a nova coluna, a variável que sai pode ser uma variável degenerada e, portanto, o teste da razão dá um incremento zero para o valor da função-objetivo. Uma outra maneira de visualizar esta situação é que quando uma nova coluna é adicionada ao problema primal, isso equivale a adicionar uma nova restrição no dual. Mas, o dual é alternativo, logo, a solução dual associada ao ponto extremo degenerado do primal é um ponto extremo de uma face alternativa ótima no dual e a restrição adicionada ao dual não corta toda esta face ótima. Com isso, ainda existem soluções alternativas no dual e, consequentemente, o primal fica degenerado. O primal só deixará de ser degenerado quando a nova coluna adicionada gerar uma nova restrição no dual que corte todos os pontos da face alternativa ótima.

Portanto, este é o intuito deste trabalho, visualizar uma maneira de gerar um ponto que quando adicionado ao problema primal, corte a maior área possível da face alternativa ótima. Para tanto, usamos como variáveis duais do problema da mochila, não a solução dual associada à solução primal degenerada, mas sim o centro analítico da face alternativa ótima. Para isso, basta resolver o problema primal por um método de pontos interiores do tipo primal-dual [10], o qual usa um processo de centragem e converge para o centro analítico da face alternativa ótima no caso do dual ser alternativo (e o primal associado ser degenerado).

As perguntas que surgem naturalmente são: como construir o problema da mochila para o problema (2) relaxado? A função-objetivo do problema da mochila contém as variáveis duais referentes a todas as restrições do problema (2) relaxado ou apenas as variáveis duais referentes às restrições de demanda dos m itens? O problema da mochila será construído usando apenas as variáveis duais referentes às restrições de demanda e é possível mostrar que o custo reduzido gerado pela mochila é o mesmo custo reduzido obtido ao calculá-lo pela fórmula $c_{n+1} - c_B^t \overline{B}^{-1} a^{n+1}$, onde $\overline{B}$ é a matriz básica obtida após a inserção de mais duas colunas que se referem às variáveis de folga das restrições $x_{n+1} \geq y_{n+1}$ e $x_{n+1} \leq M y_{n+1}$.

4 Experimentos Computacionais

Para os testes computacionais foram geradas 18 classes de problemas com 20 exemplares em cada, utilizando o gerador CUTGEN1 desenvolvido em [4], considerando os seguintes parâmetros:

- m: número de tipos de itens em cada exemplar da classe;
- L: comprimento do objeto;
- v_1: limitante inferior para o tamanho relativo dos itens com relação a L. Isto é, $l_i \geq v_1 L$ para $i = 1, 2, \ldots, m$;
- v_2: limitante superior para o tamanho relativo dos itens com relação a L. Isto é, $l_i \leq v_2 L$ para $i = 1, 2, \ldots, m$;
- $\overline{d}$: demanda média dos itens que compõe o exemplar.

Cada 5-upla $(m, L, v_1, v_2, \overline{d})$ descreve uma classe homogênea de exemplares. A Tabela 1 mostra os valores usados para criar cada uma das classes.

As classes foram geradas combinando diferentes valores de v_1, v_2, m e $\overline{d}$. Para $v_1 = 0,01$ e $v_2 = 0,2$ são gerados itens pequenos em relação a L. Para $v_1 = 0,01$ e $v_2 = 0,8$ itens com

Tabela 1: Dados das classes de exemplares geradas.

Classe	v_1	v_2	m	$\overline{d}$	Classe	v_1	v_2	m	$\overline{d}$
1	0,01	0,20	10	10	10	0,01	0,80	20	100
2	0,01	0,20	10	100	11	0,01	0,80	40	10
3	0,01	0,20	20	10	12	0,01	0,80	40	100
4	0,01	0,20	20	100	13	0,20	0,80	10	10
5	0,01	0,20	40	10	14	0,20	0,80	10	100
6	0,01	0,20	40	100	15	0,20	0,80	20	10
7	0,01	0,80	10	10	16	0,20	0,80	40	10
8	0,01	0,80	10	100	17	0,20	0,80	40	10
9	0,01	0,80	20	10	18	0,20	0,80	40	100

tamanhos variados (pequenos, médios e grandes) e para $v_1 = 0,2$ e $v_2 = 0,8$ itens com tamanhos grandes em relação a L.

Com relação a m, para $m = 10$ temos um problema pequeno, $m = 20$ um problema médio e $m = 40$ consideramos um problema grande.

Já se $\overline{d} = 10$ temos um problema com itens de demanda baixa e $\overline{d} = 100$ temos um problema com itens de demanda alta.

Foi implementada em linguagem C, a técnica de geração de colunas com e sem aceleração. Na versão com aceleração, para resolver, a cada iteração do método o problema (2) relaxado restrito (composto por um número limitado de variáveis) foi usado um algoritmo do método de pontos interiores do tipo primal-dual, o qual que faz parte da biblioteca *Callable* do Cplex. Além disso, a função objetivo do problema da mochila foi composta pelos multiplicadores de Lagrange advindos da solução obtida pelo método de pontos interiores. Para o caso sem aceleração, o problema (2) relaxado restrito foi resolvido usando um algoritmo primal simplex, que também faz parte da biblioteca do Cplex e os coeficientes da função-objetivo do problema da mochila usados foram as variáveis duais associadas com a base ótima.

Nos testes, comparamos o desempenho das duas versões (com e sem aceleração) com relação ao número de novas colunas geradas. A Tabela 2 contém: o número máximo de colunas geradas pela versão que usa o método de pontos interiores (MPI) e pela versão que usa o método simplex (MS), isto é, dentro de cada classe é informado qual foi o maior número de colunas novas geradas (Máximo) dentre os 20 exemplares daquela classe; o menor número de colunas geradas (Mínimo); o número médio de colunas geradas (Média) em cada classe. Na última linha da tabela está a média geral de cada coluna. Além disso, comparamos também as duas versões com relação ao tempo total de resolução dos problemas. A Figura 1 mostra o perfil de desempenho feito a partir dos resultados obtidos.

Diante dos resultados apresentados na Tabela 2 e na Figura 1, vemos que a estratégia de usar o MPI se mostrou mais vantajosa, pois gerou menos colunas na média e teve um melhor desempenho com relação ao tempo total de resolução, resolvendo aproximadamente 72% dos problemas com menor tempo enquanto que a versão MS não chegou a resolver 30% dos problemas com menor tempo.

5 Conclusões

Neste trabalho, propomos uma nova abordagem para acelerar a convergência do método simplex com geração de colunas, principalmente para problemas que são altamente degenerados. Os experimentos computacionais indicaram que isto é verdade, apesar dos resultados obtidos não te-

Tabela 2: Número de colunas geradas usando MPI e Simplex.

Classes	**Máximo**		**Mínimo**		**Média**	
	MPI	**MS**	**MPI**	**MS**	**MPI**	**MS**
1	28	27	14	14	20,55	20,70
2	30	33	10	11	17,80	18,30
3	44	43	26	27	33,50	34,65
4	51	53	23	23	29,70	31,10
5	78	81	54	55	62,20	65,05
6	60	64	38	42	49,25	52,10
7	15	16	3	3	8,90	9,40
8	15	19	2	2	6,80	7,60
9	32	41	11	14	22,45	26,35
10	39	42	9	9	23,15	27,50
11	100	123	33	48	62,00	75,60
12	83	92	33	42	56,70	68,95
13	11	15	1	1	6,15	6,75
14	9	10	1	1	4,80	4,90
15	23	31	4	4	16,00	20,40
16	26	34	4	4	14,90	18,65
17	54	78	17	24	34,85	55,85
18	53	85	23	27	37,95	58,05
Média	41,72	49,28	17	19,5	28,20	33,44

rem sidos muito enfáticos. Isto porque, apesar de termos trabalhado com um formato de modelo de programação linear que favoreceria a geração de soluções degeneradas, a degenerescência do politopo primal depende dos dados gerados. Na análise média dos casos, concluímos que o objetivo do trabalho foi alcançado uma vez que a abordagem que usa os métodos de pontos interiores se sobressaiu ao ter apresentado um melhor desempenho tanto em relação ao número de colunas geradas quanto em relação ao tempo total de resolução.

Agradecimentos

Agradecemos ao CNPq pelo apoio financeiro.

Referências

[1] Campello, B. C., Ayres, A. O. C., Oliveira, W. A. e Ghidini, C. T. L. S. Uma nova abordagem heurística para determinar os padrões de corte no problema de corte de estoque, *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, volume 5, 2017.

[2] Chen, C. S., Hart, S. M. and Tham, W. M. A simulated annealing heuristic for the one-dimensional cutting stock problem, *European Journal of Operations Research*, 93:522–535, 1996.

[3] Degraeve, Z. and Peeters, M. Optimal integer solutions to industrial cuttingstock problems: Part 2, benchmark results, *INFORMS Journal on Computing*, 15(1):58–81, 2003.

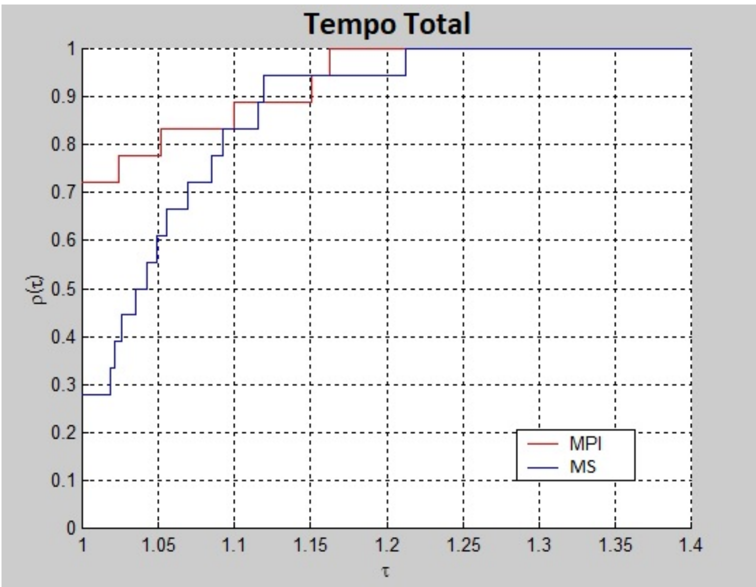

Figura 1: Perfil de desempenho – Tempo total de resolução.

[4] Gau, T. and Wäscher, G. CUTGEN1: A problem generator for the standard one-dimensional cutting stock problem, *European Journal of Operational Research*, 84:572–579, 1995.

[5] Gilmore, P. C. and Gomory, R. E. A Linear Programming Approach to the Cutting-Stock Problem, *Operations Research*, 9(6):849–859, 1961.

[6] Hoto, R., Maculan, N., Marques, F. e Arenales, M. Um problema de corte com padrões compartimentados, *Pesquisa Operacional*, 23(1):169–187, 2003.

[7] Poldi, K. C. e Arenales, M. N. Heurísticas para o problema de corte de estoque unidimensional inteiro, *Pesquisa Operacional*, 26(3):473–492, 2006.

[8] Suliman, S. M. A. Pattern generating procedure for the cutting stock problem, *International Journal of Production Economics*, 74:293–301, 2001.

[9] Wäscher, G. and Gau, T. Heuristics for the integer one-dimensional cutting stock problem: A computational study, *Operations-Research-Spektrum*, 18(3):131–144, 1996.

[10] S.J. Wright, *Primal-Dual Interior Point Methods.* SIAM Publications, Philadelphia, 1996.