Plano de Estudos

MAC5701 - Tópicos em Ciência da Computação

Aplicação do Método de Geração de Colunas ao Problema de Corte de Estoque Bidimensional

Aluno: Glauber Ferreira Cintra

Orientadora: Profa. Dra. Yoshiko Wakabayashi

Área: Otimização Combinatória

— São Paulo, março de 2003 —

1 Introdução

Apresentamos neste documento o plano de estudos a ser seguido, o qual deve servir de apoio para a elaboração de nossa tese de doutorado. Inicialmente definimos o problema a ser pesquisado e apresentamos suas principais variantes. Em seguida, introduzimos a técnica de geração de colunas e discutimos como implementá-la para o problema em questão. Finalmente, detalhamos o plano de estudos indicando algumas das referências bibliográficas com as quais trabalharemos e os métodos de resolução que pretendemos implementar.

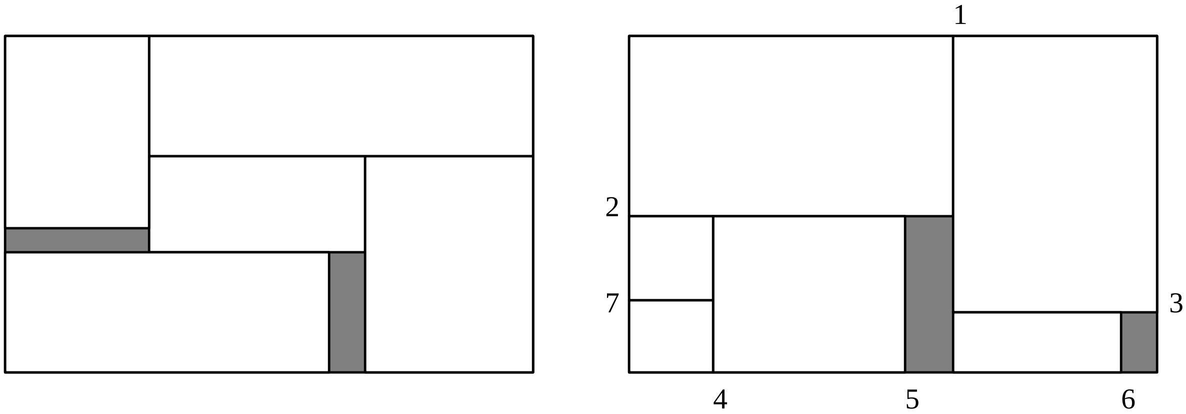
2 O Problema de Corte de Estoque Bidimensional

Nas últimas quatro décadas, os problemas de corte e empacotamento têm sido largamente estudados por um número crescente de pesquisadores, gerando e tendo se beneficiado de avanços significativos em diversas áreas, tais como *programação linear*, *programação dinâmica*, *teoria da complexidade computacional* e *algoritmos de aproximação*. O interesse por estes problemas é em parte explicado por sua aparente simplicidade e grande aplicabilidade prática. São porém, problemas de natureza complexa, $\mathcal{NP}$*-difíceis* [5].

Estes problemas surgem em diversas situações do mundo real. Podemos citar os processos de corte de barras de aço e alumínio, bobinas de papel, chapas de metal e madeira, lâminas de vidro, peças de carpete, blocos de isopor, etc. Já os processos de carregamento de contêineres de navio, carrocerias de caminhões e vagões de trem são alguns exemplos práticos de problemas de empacotamento. Pequenas melhorias nestes processos podem levar a ganhos substanciais, dependendo da escala de produção, e representar uma vantagem decisiva na competição com outras empresas do setor.

Nosso estudo concentra-se no problema de corte de estoque bidimensional (PCE_2). O PCE_2 consiste em: dado um objeto retangular, genericamente denominado de *placa*, de comprimento C e largura L, e uma lista de m itens retangulares, cada item i com comprimento c_i e largura $l_i \in I\!N$ $(i = 1, \ldots, m)$, determinar "a melhor maneira" de cortar o objeto de forma a produzir os itens da lista. Dependendo do contexto, "a melhor maneira" significa minimizar o desperdício (sobra) do objeto ou minimizar a quantidade de objetos utilizados, ou ainda maximizar o valor dos itens produzidos. Neste último caso um valor arbitrário está associado a cada item . Descrevemos a seguir as variantes do PCE_2 que temos interesse em estudar.

Uma restrição bastante comuns nos processos que envolvem corte de placas é exigir que todos os cortes sejam ortogonais ao um dos lados da placa e se estendam em linha reta desde um lado até o lado oposto da placa. Tais cortes são chamados de *corte de guilhotina*. Chamamos de padrão de corte (ou simplesmente *padrão*) cada possível forma de cortar uma placa. Dizemos que um padrão é guilhotinável se pode ser obtido com uma sequência de cortes de guilhotina. Na figura a seguir temos, à esquerda, um padrão não guilhotinável e, à direita, um padrão guilhotinável. O problema de corte de guilhotina é uma variante do PCE_2 onde todos os padrões têm que ser guilhotináveis.



(a) Padrão não guilhotinável (b) Padrão guilhotinável

Christofides e Whitlock [2] propuseram associar a cada item i um inteiro b_i ($i = 1, \ldots, m$), que representa o número máximo de vezes que o item i pode aparecer em um padrão qualquer. Tal restrição, puramente arbitrária, visa diminuir a quantidade de padrões viáveis, numa tentativa de facilitar a resolução do problema. Este variante é conhecida como PCE_2 *restrito*.

Finalmente, podemos associar a cada item i uma demanda inteira d_i ($i = 1, \ldots, m$). neste caso, poderemos vir a cortar muitas placas de modo a atender a demanda de todos os itens. Combinando estas três características, obtemos as seguintes variantes do PCE_2:

- Problema de guilhotina sem demanda
- Problema de guilhotina com demanda
- Problema de guilhotina restrito sem demanda
- Problema de guilhotina restrito com demanda
- PCE_2 sem demanda
- PCE_2 com demanda
- PCE_2 restrito sem demanda
- PCE_2 restrito com demanda

Na seção seguinte, descreveremos como aplicar o método de geração de colunas para as variantes do PCE_2 onde os itens possuem demanda. Para o leitor interessado em conhecer estratégias de resoluções para estas e outras variantes do PCE_2, recomendamos a revisão bibliográfica de Lodi, Martello e Monacci [10].

3 Geração de Colunas

O método de geração de colunas foi pela primeira vez proposto para o problema de corte de estoque unidimensional (PCE_1) por Gilmore e Gomory [6, 7]. Para aplicar o método de geração de colunas, precisamos primeiramente formular o PCE_1 como um problema de programação linear inteira.

Para fazer isso, representamos cada padrão j por um vetor-coluna a_j, cujo i-ésimo elemento indica o número de vezes que o item i ocorre nesse padrão. O problema agora consiste em considerar os padrões viáveis e decidir quantas vezes cada padrão deve ser utilizado de modo a atender a demanda, minimizando o número total de objetos utilizados. Supondo existir n padrões viáveis, introduzimos uma variável x cujos elementos são inteiros $x_j\,(j = 1, \ldots, n)$, que indicam quantas vezes o padrão j é selecionado. Assim, denotando por A a matriz $m \times n$ cujas colunas são os vetores $a_1, \ldots, a_n$, e representando por d o vetor das demandas, o problema pode ser assim formulado:

$$
\begin{aligned}
&\min \textstyle\sum_{j=1}^{n} x_j \\
&\text{sujeito a } Ax = d \\
&\qquad\quad x_j \geq 0 \text{ e inteiro} \qquad j = 1, \ldots, n.
\end{aligned}
\tag{1}
$$

A formulação acima traz consigo duas dificuldades em termos computacionais. A primeira é determinar a matriz A (que pode ter um número exponencial de colunas); a segunda é resolver um problema de programação linear inteira (que é sabido ser $\mathcal{NP}$-difícil).

Para se desvencilhar destas dificuldades, Gilmore e Gomory propuseram o método de geração de colunas que consiste, como o próprio nome sugere, em ir gerando gradativamente as colunas da matriz A. Iniciamos tomando como matriz A a matriz identidade de ordem m. Resolvemos a relaxação linear de (1):

$$
\begin{aligned}
&\min \textstyle\sum_{j=1}^{n} x_j \\
&\text{sujeito a } Ax = d \\
&\qquad\quad x_j \geq 0 \qquad j = 1, \ldots, n.
\end{aligned}
\tag{2}
$$

Para isso, usamos o algoritmo *simplex revisado* [3], onde a cada iteração uma nova coluna é gerada. Se (x, y) é a solução corrente de (2), onde $y = [y_1, \ldots, y_m]$ é a variável dual correspondente, uma nova coluna é gerada resolvendo-se o seguinte problema da mochila:

$$\begin{aligned} &\max \textstyle\sum_{i=1}^{m} y_i z_i \\ &\text{sujeito a } \textstyle\sum_{i=1}^{m} z_i l_i \leq L \\ &\qquad\qquad z_i \geq 0 \text{ e inteiro} \qquad\qquad i = 1, \ldots, m\ . \end{aligned}$$

A solução ótima de (2) não será necessariamente inteira. Para contornar este problema, Gilmore e Gomory propuseram, após achar a solução ótima, arredondar para cima o valor das variáveis (isto é, arredondar cada x_i para $\lceil x_i \rceil$), obtendo assim uma solução inteira. No entanto, este procedimento pode acarretar a produção de itens em quantidade superior à demanda, e conduz a uma solução inteira eventualmente longe da ótima. Diversas heurísticas têm sido propostas para obter uma solução inteira mais próxima da ótima [4, 12].

Nosso objetivo é aplicar o método de geração de colunas para o PCE_2. Para isto é necessário dispor de métodos para gerar novas colunas (não basta apenas resolver um problema da mochila pois é preciso certficar-se que os itens não se sobrepõem) e a partir da solução de (2) obter uma boa solução inteira.

4 Plano de Estudos

Pretendemos implementar algoritmos para resolver as variantes do PCE_2 alistadas no final da seção 2. Para isto, adaptaremos os algoritmos propostos por Herz [8], Beasley [1], Li e Cheng [9], Wang [11]. Estes algoritmos resolvem as tarefas necessárias (geração de novas colunas e obtenção de uma boa solução inteira) para a aplicação do método de geração de colunas para o PCE_2.

Referências

[1] J. E. Beasley, *Algorithms for unconstrained two-dimensional guillotine cutting*, Journal of the Operational Research Society, 36 (1985), pp. 297–306.

[2] N. Christofides and C. Whitlock, *An algorithm for two dimensional cutting prblems*, Operations Research, 25 (1977), pp. 30–44.

[3] V. Chvátal, *Linear Programming*, W. H. Freeman and Company, New York, 1980.

[4] G. F. Cintra, *Algoritmos híbridos para o problema de corte unidimensional*, in XXV Conferência Latinoamericana de Informática, Assunção, 1999.

[5] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$-Completeness*, W. H. Freeman and Co., San Fransisco, 1979.

[6] P. Gilmore and R. Gomory, *A linear programming approach to the cutting stock problem*, Operations Research, 9 (1961), pp. 849–859.

[7] ——, *A linear programming approach to the cutting stock problem - part II*, Operations Research, 11 (1963), pp. 863–888.

[8] J. C. Herz, *A recursive computational procedure for two-dimensional stock-cutting*, IBM Journal of Research Development, (1972), pp. 462–469.

[9] K. Li and K.-H. Cheng, *Generalized first-fit algorithms in two and three dimensions*, Int. J. Found. Comput Sci., 1 (1992), pp. 131–150.

[10] A. Lodi, S. Martello, and M. Monaci, *Two-dimensional packing problems: a survey*, European J. Oper. Res., 141 (2002), pp. 241–252. Cutting and packing.

[11] P. Y. Wang, *Two algorithms for constrained two dimensional cutting stock problems*, Operations Research, 31 (1983), pp. 573–586.

[12] G. Wäscher and T. Gau, *Heuristics for the integer one-dimensional cutting stock problem: a computational study*, OR Spektrum, 18 (1996), pp. 131–144.