UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

O Problema da Mochila Compartimentada e Aplicações

Fabiano do Prado Marques
Marcos Nereu Arenales

Nº 64

NOTAS

São Carlos - SP

UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação
ISSN 0103-2577

O Problema da Mochila Compartimentada e Aplicações

Fabiano do Prado Marques
Marcos Nereu Arenales

Nº 64

NOTAS

Série Computação

São Carlos – SP
Jan./2002

O Problema da Mochila Compartimentada e Aplicações*

Fabiano do Prado Marques
ICMC - USP
araxa@icmc.sc.usp.br

Marcos Nereu Arenales
ICMC - USP
arenales@icmc.sc.usp.br

Laboratório de Otimização – São Carlos
ICMC – USP
C.P. 668
13560-970 - São Carlos, SP

RESUMO

O *Problema da Mochila Compartimentada* estende o clássico problema da mochila e pode ser enunciado considerando-se a seguinte situação hipotética: um alpinista deve carregar sua mochila de capacidade limitada com possíveis itens de sua utilidade. A cada item atribui-se o seu *peso* e um *valor de utilidade* (até aqui, o problema coincide com o clássico Problema da Mochila). Entretanto, os itens são de *agrupamentos* distintos (alimentos, medicamentos, utensílios, etc.) e devem estar em compartimentos separados na mochila. Os compartimentos da mochila são flexíveis e têm capacidades limitadas. A inclusão de um compartimento tem um *custo* fixo que depende do agrupamento com que foi preenchido, além de introduzir uma perda da capacidade da mochila. O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total, descontado o custo de incluir compartimentos. Nesse trabalho, propomos um modelo de otimização não linear inteiro para o problema e algumas heurísticas para sua resolução. Uma aplicação prática de relevância deste problema aparece no corte de bobinas de aço, sujeito à laminação, detalhado em apêndice.

Palavras-chave: O Problema da Mochila Compartimentada, O Problema da Mochila, Otimização Inteira.

ABSTRACT

The Clustered Knapsack Problem extends the classical knapsack problem and can be stated as the following hypothetical situation: an alpinist should carry his knapsack of limited capacity with possible items of his utility. To each item it is associated a weight and a utility value (so far, the problem coincides with the classical integer Knapsack Problem). However, the items are of different classes (foods, medicaments, utensils, etc.) and they should be packed in separated clusters in the knapsack. The clusters are flexible and have limited capacities. Each cluster has a cost that depends on the class of items it has been filled and, in addition, each new cluster introduces a loss of capacity of the knapsack. The Clustered Knapsack Problem consists of determining the suitable capacities of each cluster and how these clusters should be filled. The objective is to maximize a total utility value paid off the cost of the clusters. In this work, we model the problem as an integer non-linear optimization problem and design some heuristic methods for its solution. A practical application of this problem arises in cutting steel coils subject to the lamination, discussed in details in appendix.

Keywords: The Clustered Knapsack Problem, The Knapsack Problem, Integer Optimization.

* Projeto financiado pela FAPESP.

O Problema da Mochila Compartimentada e Aplicações

1. Introdução

Cortar unidades maiores em unidades menores ou empacotar unidades menores dentro de unidades maiores são problemas idênticos, considerando que um item cortado de uma certa posição pode ser pensado como ocupando aquela posição. O *Problema de Empacotamento* consiste, genericamente, em empacotar unidades pequenas dentro de uma unidade grande, de forma que certo objetivo seja otimizado. Um exemplo desse problema consiste em arranjar o maior volume possível de caixas dentro de um contêiner. Por outro lado, o *Problema de Corte*, de forma genérica, consiste em cortar uma unidade grande (*objeto*), que esteja disponível, para a produção de um conjunto de unidades pequenas (*itens*) que estão sendo requisitadas. As formas e medidas do objeto e dos itens são bem especificadas. Conforme os itens solicitados, podemos combiná-los dentro de um objeto de inúmeras maneiras, respeitando-se um conjunto de restrições do processo de cortagem. A estas combinações denominamos *planos de corte*. O plano de corte ótimo é aquele que produz, por exemplo, a menor perda. O número de planos de corte é, na prática, muito elevado, exigindo que técnicas bem elaboradas sejam desenvolvidas para determinar o plano ótimo. Dentre essas técnicas podemos citar: enumeração implícita, programação dinâmica, relaxação lagrangiana, busca em grafos e heurísticas.

Quando uma quantidade elevada de itens deve ser produzida, temos um problema em que a solução exige a cortagem de vários objetos em estoque e a repetição de vários planos de corte. Este problema é conhecido na literatura como *Problema de Corte de Estoque* e o objetivo pode ser, entre outros, o menor número de objetos cortados, ou o menor custo total dos objetos cortados, considerando diferentes custos para os objetos em estoque.

Os problemas de cortes podem ser classificados de acordo com várias características [Dyckhoff e Finke,1992]. Em particular, uma primeira consiste nas dimensões relevantes do processo de cortagem. Assim, podemos ter os problemas de cortes unidimensionais, com apenas uma dimensão relevante ao processo de corte, como por exemplo, o corte de bobinas de papel, aço, tecidos, plásticos, entre outros. Já os problemas bidimensionais, nos quais duas dimensões são relevantes, podemos citar alguns já bastante estudados, como o corte de placas de madeira na indústria de móveis, chapas de aço e placas de vidro. Entre os problemas tridimensionais podemos citar o empacotamento de unidades em contêineres, cortes em indústrias de espuma, entre outros.

Além destes, temos ainda problemas do tipo 1.5-dimensional, onde uma das duas dimensões consideradas é variável. Este caso tem um exemplo no corte de peças de vestuário. Outros problemas são do tipo 2.5-dimensional, onde uma das três dimensões é variável. Um exemplo a ser citado é o problema de se efetuar o carregamento de unidades dentro de caixas, onde as bases estão definidas, mas a altura deverá ser determinada, por exemplo, a menor possível. Existem, também, os problemas multidimensionais, que surgem, por exemplo, associados ao Problema de Alocação de Tarefas [Dyckhoff e Finke,1992]. Porém, vale salientar que estas não são as únicas características relevantes na modelagem de um problema de corte.

Neste trabalho consideramos o problema de corte unidimensional: suponha que um objeto (barra, bobina, etc) deva ser cortado ao longo de seu comprimento em itens de comprimentos especificados. Cada item tem um valor associado que chamamos de valor de utilidade. Itens cujos comprimentos não foram especificados são considerados perdas e têm valores de utilidade nulos. O problema a ser tratado é de otimização combinatória e consiste em maximizar a soma dos valores de utilidades dos itens cortados. Observe que o clássico problema da mochila se enquadra nesta formulação genérica, mas muitas outras restrições podem ser necessárias.

Neste projeto estudamos uma variação do clássico problema da mochila, denominado o *Problema da Mochila Compartimentada*, que pode ser enunciado da seguinte forma: um alpinista deve carregar sua mochila de capacidade limitada com possíveis itens de sua utilidade. A cada item atribui-se o seu *peso* e um *valor de utilidade* (até aqui, o problema coincide com o clássico Problema da Mochila). Entretanto, os itens são de *agrupamentos* distintos (alimentos, medicamentos, utensílios, etc.) e devem estar em compartimentos separados na mochila. Os compartimentos podem ser construídos com capacidades flexíveis, porém limitadas. Na inclusão de um compartimento surge um *custo* fixo que depende do agrupamento com que foi preenchido, além de introduzir uma perda da capacidade da mochila. O problema consiste em determinar as capacidades adequadas de cada compartimento e como esses devem ser carregados, maximizando o valor de utilidade total, descontado o custo da inclusão dos compartimentos.

Na seção 2, é feita uma revisão dos nove tipos de problemas da mochila encontrados na literatura. Na seção 3 definimos e modelamos o problema da mochila compartimentada. Nas seções 4 e 5 propomos e apresentamos alguns resultados computacionais. Por fim, mostramos uma aplicação prática de relevância que aparece no corte de bobinas de aço, sujeito à laminação.

2. Problemas da Mochila

Nesta seção faremos uma breve revisão dos vários tipos de Problemas da Mochila estudados na literatura. O problema enfocado neste artigo, embora possa ser catalogado como um Problema da Mochila, não foi relacionado na literatura, como nos trabalhos [Martello e Toth,1990] e [Lin,1998].

Os Problemas da Mochila caracterizam uma classe de problemas da programação linear inteira e pertencem à classe de problemas *NP-difíceis* [Garey e Johnson,1979].

Genericamente, o Problema da Mochila é definido do seguinte modo: Suponha que um alpinista deva carregar sua mochila selecionando itens, dentre vários disponíveis e considerando a capacidade da mochila. A cada item é atribuído um valor de utilidade e o alpinista deve selecioná-los buscando maximizar o valor de utilidade total.

Descrevemos a seguir os problemas que têm sido catalogados como Problemas da Mochila, segundo os trabalhos [Martello e Toth,1990] e [Lin,1998] e apresentamos a modelagem matemática correspondente.

Para o problema do alpinista enunciado anteriormente, considere os seguintes dados:

- m: número de tipos de itens;
- v_i: valor de utilidade do item tipo i, $i=1,...,m$;
- l_i: peso do item tipo i, $i=1,...,m$;
- L: capacidade da mochila.

2.1 Problema da Mochila Inteiro

O problema modelado a seguir é chamado na literatura de *Problema da Mochila Inteiro* ou simplesmente *Problema da Mochila*. Não há limitação nas quantidades de itens selecionados. Este problema representa o corte de uma barra que deve ser cortada ao longo de seu comprimento, sem que uma quantidade máxima de itens seja especificada. Temos, então:

Variável de decisão:

- x_i: quantidade de itens do tipo i selecionados, $i=1,...,m$.

Modelagem Matemática:

$$Maximizar\ \Phi = \sum_{i=1}^{m} v_i x_i \qquad (1.1)$$

$$Sujeito\ a: \sum_{i=1}^{m} l_i x_i \leq L \qquad (1.2)$$

$$x_i \geq 0 \text{ e inteiro}, \ i = 1,...m. \qquad (1.3)$$

Vale observar que este modelo é uma relaxação do Problema da Mochila Compartimentada uma vez que sua solução pode não ser possível acomodar os itens em compartimentos.

2.2 Problema da Mochila 0-1

O Problema da Mochila 0-1 é, talvez, o mais importante Problema da Mochila e um dos mais estudados problemas de otimização discreta. A razão para tal interesse está, basicamente, ligada a três fatores:

a) Pode ser visto como o mais simples problema de otimização linear inteira;
b) Aparece como um subproblema em muitos outros problemas mais complexos;
c) Pode representar uma gama muito grande de situações práticas.

No Problema da Mochila 0-1 temos a situação em que um único exemplar de cada item pode ser selecionado. Neste caso, as variáveis de decisão são:

- $x_i = \begin{cases} 1, \textit{se o item i for selecionado;} \\ 0, \textit{caso contrário.} \end{cases} \qquad i = 1,\ldots,m.$

Modelagem Matemática:

$$\text{Maximizar } \Phi = \sum_{i=1}^{m} v_i x_i \qquad (2.1)$$

$$\text{Sujeito a: } \sum_{i=1}^{m} l_i x_i \leq L \qquad (2.2)$$

$$x_i = 0 \text{ ou } 1,\ i = 1,\ldots m. \qquad (2.3)$$

2.3 Problema da Mochila Restrito

Alguns problemas podem apresentar condições adicionais, como por exemplo, a limitação da quantidade de itens a serem selecionados. Neste caso, o problema passa a ser chamado de *Problema da Mochila Restrito ou Limitado*. Para o Problema da Mochila Restrito, além dos dados necessários para os modelos anteriores, considere ainda:

- d_i: quantidade máxima de itens tipo i que pode ser selecionada, $i=1,\ldots,m$;

Modelagem Matemática:

$$\text{Maximizar } \Phi = \sum_{i=1}^{m} v_i x_i \qquad (3.1)$$

$$\text{Sujeito a: } \sum_{i=1}^{m} l_i x_i \leq L \qquad (3.2)$$

$$0 \leq x_i \leq d_i \text{ e inteiro},\ i = 1,\ldots m. \qquad (3.3)$$

Alguns problemas de cortes são formulados como um Problema da Mochila Restrito onde a produção excedente de itens é considerada perda. Tal problema pode ser visto como uma generalização do Problema da Mochila 0-1, onde $d_i = 1$, $i = 1,\ldots,m$.

2.4 Problema da Soma de Subconjuntos

O Problema da Soma de Subconjuntos consiste em selecionar um subconjunto de itens cuja soma total dos pesos dos itens escolhidos se aproxime ao máximo de L, sem excedê-lo.

Modelagem Matemática:

$$Maximizar\ \Phi = \sum_{i=1}^{m} l_i x_i \qquad (4.1)$$

$$Sujeito\ a\text{:}\ \sum_{i=1}^{m} l_i x_i \leq L \qquad (4.2)$$

$$x_i = 0\ ou\ 1,\ i = 1,...m. \qquad (4.3)$$

O Problema da Soma de Subconjuntos é um caso particular do Problema da Mochila 0-1, onde $v_i = l_i$ para $i = 1,...,m$. Tal problema é conhecido, também, como Problema da Mochila de Valor Independente.

2.5 Problema do Troco

O Problema do Troco consiste em selecionar um número x_i $(i = 1,...,m)$ de itens de cada tipo i tal que o peso total seja L e o número total de itens selecionados seja mínimo.

Modelagem Matemática:

$$Minimizar\ \Phi = \sum_{i=1}^{m} x_i \qquad (5.1)$$

$$Sujeito\ a\text{:}\ \sum_{i=1}^{m} l_i x_i = L \qquad (5.2)$$

$$x_i \geq 0\ e\ inteiro,\ i = 1,...m. \qquad (5.3)$$

O problema é chamado Problema do Troco, pois pode ser interpretado como o problema de um caixa (supermercado, banco, etc.) que deve devolver uma determinada quantia, L, usando, para isto, um número mínimo de moedas (ou notas) de valores específicos l_i $(i = 1,...,m)$. Nesse caso, para cada valor, um número ilimitado de moedas está disponível. É importante notar que a condição de igualdade imposta pode fazer com que inexista uma solução para o problema.

2.6 Problema da Mochila Quadrático

O Problema da Mochila Quadrático é um dos problemas mais estudados de Mochila Não Linear. Um Problema da Mochila Não Linear, em geral, é o Problema da Mochila com função objetivo não linear ou que envolve restrições não lineares. Um Problema da Mochila Quadrático geral pode ser formulado da seguinte forma:

Modelagem Matemática:

$$Maximizar\ \Phi = x^T Q x + c^T x \qquad (6.1)$$

$$Sujeito\ a\text{:}\ \sum_{i=1}^{m} l_i x_i \leq L \qquad (6.2)$$

$$x_i \geq 0\ \ e\ inteiros,\ i = 1,...,m \qquad (6.3),$$

com $x^T = (x_1, x_2, ..., x_m)$, $c^T = (c_1, c_2, ..., c_m)$ e Q uma matriz m x m, simétrica e não nula.

Até aqui, os Problemas da Mochila consistiram em carregar apenas uma mochila. Convém observar que o problema da Mochila Compartimentada pode ser modelado como um problema quadrático, onde a função objetivo e algumas restrições são quadráticas (seção 3.2). Os próximos problemas, também classificados por [Martello e Toth,1990] e [Lin,1998] como Problemas da Mochila, consistem em carregar várias mochilas.

2.7 Problema de Múltiplas Mochilas 0-1

O Problema de Múltiplas Mochilas 0-1 consiste em carregar um conjunto de n mochilas ($n \leq m$) cujas capacidades são dadas por: $L_j, j = 1,...,n$. Os demais dados do problema são os mesmos do problema para uma única mochila.

O problema consiste em selecionar subconjuntos disjuntos de itens e associa-los a diferentes mochilas, cujas capacidades não sejam excedidas, tal que o valor de utilidade dos itens selecionados seja máximo.

Variável de decisão:

- $x_{ij} = \begin{cases} 1, \textit{se o item i está associado à mochila j;} \\ 0, \textit{caso contrário.} \end{cases}$

Modelagem Matemática:

$$Maximizar\ \Phi = \sum_{j=1}^{n}\sum_{i=1}^{m} v_i x_{ij} \qquad (7.1)$$

$$Sujeito\ a: \sum_{i=1}^{m} l_i x_{ij} \leq L_j, \quad j = 1,...,n \qquad (7.2)$$

$$\sum_{j=1}^{n} x_{ij} \leq 1, \quad i = 1,...,m \qquad (7.3)$$

$$x_{ij} = 0\ ou\ 1,\ i = 1,...,m\ e\ j = 1,...,n \qquad (7.4)$$

Quando $n = 1$, o Problema de Múltiplas Mochilas 0-1 se reduz a um Problema da Mochila 0-1 simples, pois a restrição (7.3) torna-se redundante.

Note que este problema preserva a característica dos problemas anteriores (onde apenas uma mochila deveria ser preenchida) de que uma seleção de itens deve ser feita, ou seja, as mochilas são insuficientes para carregar todos os itens e a seleção é tal que o valor de utilidade seja maximizado (exceto o problema do troco, mas que também é de seleção de itens). Nos próximos dois problemas, todos os itens serão alocados, de modo que há uma seleção de mochilas.

Em [Dyckhoff e Finke,1992] é apresentado uma classificação dos problemas de corte e empacotamento (problemas da mochila são casos particulares) destacando esta característica (seleção de itens ou seleção de mochilas), além da dimensão do problema, como descrito na introdução deste trabalho.

Note ainda que, embora cada uma das mochilas possa ser vista como um compartimento a ser alocado numa mochila maior, este problema não modela o caso da mochila compartimentada, pois além das capacidades ocupadas pelos compartimentos serem incógnitas, somente itens de mesma característica podem ser alocados ao mesmo compartimento, o que não é considerado em (7.1)-(7.4).

2.8 Problema de Designação Generalizada

O Problema de Designação Generalizada pode ser descrito usando a terminologia aplicada para os Problemas da Mochila e é semelhante ao problema de múltiplas mochilas, porém o valor de utilidade e o peso do item i podem variar se associados a mochilas diferentes. Considere os seguintes dados:

- v_{ij}: ganho fornecido pela associação do item i à mochila j, $i=1,...,m$ e $j=1,...,n$;
- l_{ij}: peso do item i se associado à mochila j, $i=1,...,m$ e $j=1,...,n$;

Modelagem Matemática:

$$Maximizar\ \Phi = \sum_{j=1}^{n}\sum_{i=1}^{m} v_{ij}x_{ij} \qquad (8.1)$$

$$Sujeito\ a: \sum_{i=1}^{m} l_{ij}x_{ij} \le L_j \quad j=1,...,n \qquad (8.2)$$

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i=1,...,m \qquad (8.3)$$

$$x_{ij} = 0\ ou\ 1,\ i = 1,...,m\ e\ j = 1,...,n \qquad (8.4)$$

O problema é, freqüentemente, descrito na literatura como o problema de designação ótima de m tarefas a n processadores, dados o ganho v_{ij} e a quantidade de recursos l_{ij} correspondente à associação da tarefa i ao processador j e a quantidade total de recursos L_j suportados por cada processador j.

Note que este problema difere dos anteriores em dois aspectos: (i) todos os itens devem ser alocados (restrições (8.3)) e, (ii) os itens sofrem "deformações" se alocados em mochilas distintas.

2.9 Problema de Múltiplas Mochilas Idênticas (*Bin-Packing*)

O Problema de Múltiplas Mochilas Idênticas, mais conhecido na literatura por problema *Bin-Packing* considera:

- l_i: peso do item i, $i=1,...,m$;
- L: capacidade de cada caixa.

Todo item deve ser associado a uma única caixa, tal que o peso total dos itens em cada caixa não exceda L e o número de caixas usadas seja mínimo.

Variáveis de decisão:

- $y_j = \begin{cases} 1, \textit{se a caixa j é usada;} \\ 0, \textit{caso contrário.} \end{cases}$
- $x_{ij} = \begin{cases} 1, \textit{se o item i está associado à caixa j;} \\ 0, \textit{caso contrário.} \end{cases}$

Modelagem Matemática:

$$Minimizar\ \Phi = \sum_{j=1}^{n} y_j \qquad (9.1)$$

$$Sujeito\ a: \sum_{i=1}^{m} l_i x_{ij} \le L_j . y_j \quad j = 1,...,n \qquad (9.2)$$

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i=1,...,m \qquad (9.3)$$

$$y_j = 0\ ou\ 1,\ x_{ij} = 0\ ou\ 1,\ i = 1,...,m\ e\ j = 1,...,n. \qquad (9.4)$$

Nesta seção revisamos os vários problemas da mochila assim catalogados na literatura segundo [Martello e Toth,1990] e [Lin,1998]. Estes problemas diferem do Problema da Mochila Compartimentada, a ser detalhado nas próximas seções.

3. *O Problema da Mochila Compartimentada*

O problema enfocado neste estudo, o *Problema da Mochila Compartimentada*, tem sido pouco estudado na literatura [Hoto,1996] e [Marques,2000], sendo implicitamente tratado em [Ferreira *et al.*,1990] e [Carvalho,1991].

3.1 Definição do Problema

Um alpinista deve carregar sua mochila com m possíveis itens de sua utilidade. A cada item i, o alpinista atribui um *valor de utilidade* v_i e seu peso l_i. O máximo peso que o alpinista suporta em sua jornada é de L. Até este ponto, o problema coincide com a formulação clássica do Problema da Mochila. Porém, os itens são de *agrupamentos* distintos (por exemplo: *agrupamento 1*: alimentos, *agrupamento 2*: utensílios, *agrupamento 3*: roupas, *agrupamento 4*: calçados, etc.) e devem estar em compartimentos separados dentro da mochila (isto é, cada compartimento contém somente itens de um mesmo agrupamento). Os compartimentos da mochila são flexíveis, permitindo que o alpinista carregue maior peso em alimentos do que em roupas, por exemplo. Entretanto, as capacidades dos compartimentos (incógnitas) são limitadas inferior e superiormente (de acordo com os agrupamentos), caso estes sejam criados, digamos por: L_{min} e L_{max} (isto é, a soma dos pesos l_i dos itens alocados no compartimento deve ser superior ou igual a L_{min} e inferior ou igual a L_{max}) . A cada compartimento é associado um custo c_k, caso este seja preenchido com itens do agrupamento k e, além disso, cada compartimento incluído na mochila produzirá uma perda da capacidade da mesma, digamos, S (isto é, se, por exemplo, três compartimentos forem utilizados, então a capacidade disponível na mochila será de L-$3S$, ao invés de L). É importante ressaltar que vários compartimentos, de capacidades diferentes, podem ser criados para carregar itens de um mesmo agrupamento ou, em outras palavras, itens de um mesmo agrupamento podem ser alocados em mais que um compartimento.

O *Problema da Mochila Compartimentada* consiste em determinar as capacidades adequadas de cada compartimento e como estes devem ser carregados de modo que o valor de utilidade total (soma dos valores de utilidade de todos os itens selecionados) seja máximo, descontando-se os custos dos compartimentos, os quais dependem dos agrupamentos com que foram preenchidos (c_k).

3.2 Um Modelo Matemático para o Problema da Mochila Compartimentada

Nesta seção, propomos uma modelagem matemática para *o Problema da Mochila Compartimentada*. Observe que a cada compartimento da mochila, tem-se associado um único agrupamento k de itens, ou seja, não se admite itens de agrupamentos distintos no mesmo compartimento.

<u>Dados do problema:</u>

- $M = \{1,...,m\}$: conjunto dos tipos de itens;
- l_i: peso (Kg) do item i ($l_i > 0$), $i=1,...,m$;
- v_i: valor de utilidade do item i ($v_i \geq 0$), $i=1,...,m$;
- d_i: limite máximo de itens i na mochila, $i=1,...,m$;
- K: número total de agrupamentos distintos;
- A_k: agrupamento k, $k=1,...,K$, ($M = A_1 \cup A_2 ... \cup A_K$, $A_i \cap A_j = \emptyset$, com $i \neq j$);

- N_k: número total de possíveis compartimentos para o agrupamento k (por exemplo: $A_1=\{1,2\}$, então $L_{11} = l_1+l_2$, $L_{21} = 2l_1+l_2$, etc., são duas possíveis capacidades de compartimentos para o agrupamento 1. O número N_k depende essencialmente dos itens do agrupamento k);
- c_k: custo de um compartimento de itens do agrupamento k, onde $c_k \geq 0$, $k=1,\ldots,K$;
- L: capacidade (Kg) da mochila;
- L^k_{min} : capacidade mínima (Kg) de cada compartimento associado ao agrupamento A_k;
- L^k_{max}: capacidade máxima (Kg) de cada compartimento associado ao agrupamento A_k ($L^k_{min} < L^k_{max} < L$);
- S: perda (Kg) decorrente da inclusão de um novo compartimento na mochila.

Variáveis:

- α_{ijk}: número de itens do tipo i, do agrupamento k, no compartimento do tipo j, $i=1,\ldots,m$, $k=1,\ldots,K$ e $j=1,\ldots,N_k$;
- β_{jk}: número de vezes que o compartimento (padrão) do tipo j alocado com o agrupamento k é repetido na mochila, $k=1,\ldots,K$ e $j=1,\ldots,N_k$;

Assim, o *j-ésimo* compartimento com itens do agrupamento k tem:

- A capacidade ocupada dada por:

$$L_{jk} = \sum_{i\in A_k} l_i\alpha_{ijk}, \qquad k=1,\ldots,K \text{ e } j=1,\ldots,N_k. \qquad (10)$$

- O valor de utilidade dado por:

$$V_{jk} = \sum_{i\in A_k} v_i\alpha_{ijk}, \quad k=1,\ldots,K \text{ e } j=1,\ldots,N_k. \qquad (11)$$

Um modelo matemático para o problema da mochila compartimentada pode ser escrito por:

$$\begin{aligned}
&\textit{Maximizar} && \sum_{k=1}^{K}\sum_{j=1}^{N_k} (V_{jk}-c_k)\beta_{jk} && (12.1)\\
&\textit{Sujeito a:} && V_{jk} = \sum_{i\in A_k} v_i\alpha_{ijk},\ k=1,\ldots,K \text{ e } j=1,\ldots,N_k && (12.2)\\
& && L_{jk} = \sum_{i\in A_k} l_i\alpha_{ijk},\ k=1,\ldots,K \text{ e } j=1,\ldots,N_k && (12.3)\\
& && L^k_{min} \leq L_{jk}+S \leq L^k_{max},\ k=1,\ldots,K \text{ e } j=1,\ldots,N_k && (12.4)\\
& && \sum_{j=1}^{N_k} \alpha_{ijk}\beta_{jk} \leq d_i,\ i=1,\ldots,m && (12.5)\\
& && \sum_{k=1}^{K}\sum_{j=1}^{N_k}(L_{jk}+S)\beta_{jk} \leq L && (12.6)\\
& && \alpha_{ijk} \geq 0, \textit{ inteiro},\ i=1,\ldots,m,\ k=1,\ldots,K \text{ e } j=1,\ldots,N_k && (12.7)\\
& && \beta_{jk} \geq 0, \textit{ inteiro},\ k=1,\ldots,K \text{ e } j=1,\ldots,N_k && (12.8)
\end{aligned}$$

A função objetivo *(12.1)*, a ser maximizada, fornece o valor total de utilidade, descontado o custo de alocar os compartimentos; as restrições *(12.2)* e *(12.3)* fornecem, respectivamente, o valor de utilidade e a capacidade de cada compartimento (obviamente, estas restrições podem ser eliminadas, substituindo-as nas demais restrições); as restrições *(12.4)* impõem limites físicos aos compartimentos, onde uma perda S deve ser considerada; as restrições *(12.5)* limitam o número de itens na mochila (esta restrição, embora não tenha sido explicitada no enunciado do *Problema da Mochila Compartimentada*, é importante quando um número excessivo de itens deve ser evitado) e, por fim, a restrição *(12.6)* correspondente aos limites físicos da mochila. Observe que o problema *(12.*)* é um problema de otimização não linear inteiro.

O modelo *(12.*)* permite considerar um agrupamento especial de itens, chamados *"itens livres"*, os quais podem ser incluídos na mochila sem necessidade de compartimentação. Nas heurísticas a seguir, denominamos o agrupamento K como o agrupamento de itens livres. Assim, para $k=K$, a restrição *(12.4)* é desnecessária (ou, $L_K^{min}=0$ e $L_K^{max}=L$ de modo que é redundante) e $c_K=0$. Na aplicação em interesse, isto é, no corte de bobinas de aço sujeito à laminação, os limitantes dos compartimentos são idênticos: $L_k^{min}=L^{min}$ e $L_k^{max}=L^{max}$, exceto para o compartimento dos itens livres. Nas heurísticas propostas na próxima seção usamos esta característica, embora seja facilmente implementável quando os limitantes para os compartimentos dependem dos agrupamentos.

4. Heurísticas para o Problema da Mochila Compartimentada

Três heurísticas *"gulosas"* são propostas para a resolução *do Problema da Mochila Compartimentada,* considerando uma quantidade máxima de itens a serem alocados na mochila (d_i, para $i=1,\dots,m$). As heurísticas apostam na direção de que o custo de incluir um compartimento é alto, de modo que os compartimentos devem ser construídos nos seus limites máximos (no caso do corte de bobinas de aço, apêndice A, este custo corresponde à laminação que é um processo caro e demorado, justificando a aposta das heurísticas). A seguir, fazemos uma exposição das heurísticas propostas e na seção 5 analisamos seus desempenhos computacionais.

4.1 Heurística de Decomposição

Esta heurística consiste de duas fases: Na primeira fase são resolvidos *(K-1)* Problemas da Mochila de capacidade L_{max}^k, um para cada agrupamento (exceto o agrupamento K dos itens livres), resultando nos melhores compartimentos associados aos agrupamentos. Na segunda fase, um problema da mochila clássico é resolvido, considerando os compartimentos obtidos na primeira fase como itens, juntamente com os itens livres, para o carregamento da mochila. Baseado no modelo *(12.*)*, buscaremos uma solução com as seguintes características:

- A função objetivo *(12.1)* pode ser beneficiada com a construção de V_{jk} grandes e, portanto, para cada agrupamento k, construímos apenas o melhor compartimento ($N_k = 1$, portanto, j pode ser suprimido) maximizando *(12.2)* restrito a *(12.3)*, *(12.4)* e *(12.7)* (veja problema *(13.*)*). O valor de α_{ik} (j suprimido) é determinado, isto é, para cada agrupamento k, temos um compartimento;
- Com $N_k = 1$ (apenas um compartimento por agrupamento) e α_{ik} já determinado: determina-se L_k (tamanho do compartimento k, veja *(12.3)*, com o índice j suprimido). O valor de β_k é determinado de modo a otimizar *(12.1)*, sujeito às demais restrições do problema *(12.*)* (veja problema *(14.*)*).

O algoritmo é apresentado a seguir.

Algoritmo:
Passo 1: Selecionar o melhor compartimento para o agrupamento k resolvendo o seguinte Problema da Mochila, de capacidade L_{max}^k, $k=1, \dots, K-1$:

$$V_k = Maximizar \sum_{i \in A_k} v_i \alpha_{ik} \qquad (13.1)$$

$$Sujeito\ a: \sum_{i \in A_k} l_i \alpha_{ik} + S \leq L^k_{max} \qquad (13.2)$$

$$0 \leq \alpha_{ik} \leq d_i \ e\ inteiro,\ para\ i \in A_k, k = 1,...,K-1. \qquad (13.3)$$

$$Seja\ L_k = \sum_{i \in A_k} l_i \alpha_{ik} : tamanho\ do\ compartimento\ k.$$

(note que α_{ik} são fixados neste passo).

Passo 2: Resolver o seguinte problema da mochila envolvendo os compartimentos determinados no passo 1 e os itens livres:

- β_k é o número de vezes que o compartimento associado ao agrupamento k é repetido na mochila,
- γ_k é o número de vezes que o item livre k é repetido na mochila.

$$Maximizar \sum_{k=1}^{K-1} (V_k - c_k)\beta_k + \sum_{k \in A_K} v_k \gamma_k \qquad (14.1)$$

$$Sujeito\ a: \sum_{k=1}^{K-1} (L_k + S)\beta_k + \sum_{k \in A_K} l_k \gamma_k \leq L \qquad (14.2)$$

$$\alpha_{ik}\beta_k \leq d_i, i \in A_k, k = 1,...,K-1 \qquad (14.3)$$

$$\gamma_k \leq d_k, \quad k \in A_K, \qquad (14.4)$$

$$\alpha_{ik}, \gamma_k \geq 0\ e\ inteiros,\ i \in A_k, k = 1,...,K-1, k \in A_K. \qquad (14.5)$$

4.2 Heurística do Melhor Compartimento

Nesta heurística, como na anterior, o melhor compartimento para cada um dos agrupamentos de itens é determinado. Temos definido, então:

$$L'_k = \begin{cases} (L_k + S),\ se\ k \leq K-1 \\ l_k,\ se\ K-1 \leq k \leq m \end{cases},$$

$$V'_k = \begin{cases} V_k - c_k,\ se\ k \leq K-1 \\ v_k,\ se\ K-1 \leq k \leq m \end{cases}.$$

Em seguida, escolhe-se, o compartimento de maior valor por unidade (V'_k / L'_k), sendo então, alocado na mochila. Atualizam-se os dados e repete-se o processo até que um plano seja gerado. Com isto é evitado o problema *(14.*)* do algoritmo anterior. Porém, apesar de apenas um compartimento ser gerado por agrupamento, a cada passo, é possível ter diferentes compartimentos com o mesmo agrupamento, algo excluído na heurística anterior, onde era permitida apenas a repetição do melhor compartimento. O algoritmo é apresentado passo a passo a seguir.

Algoritmo:

Passo 1: *lutil* $\leftarrow L$; "*lutil* guarda a capacidade disponível na mochila a cada iteração"

Passo 2: Para $k = 1$ a K-1, faça:

2.1 Se ($lutil \geq L^k_{max}$)

então $L\_compartimento = L^k_{max}$

senão Se ($lutil < L^k_{min}$)

então $L\_compartimento = lutil$ e considera-se a observação em (*15.**), senão ($L^k_{min} \leq lutil < L^k_{max}$) $L\_compartimento = lutil$

2.2 Selecionar o melhor compartimento para o agrupamento k resolvendo o seguinte Problema da Mochila de capacidade $L\_compartimento$:

$$V_k = Maximizar \quad \sum_{i \in A_k} v_i \alpha_{ik} \qquad (15.1)$$

$$Sujeito\ a: \quad \sum_{i \in A_k} l_i \alpha_{ik} + S \leq L\_compartimento \qquad (15.2)$$

$$0 \leq \alpha_{ik} \leq d_i \ e\ inteiro,\ para\ i \in A_k, k = 1,...,K-1. \qquad (15.3)$$

Observação: *(se $L\_compartimento<Lmin$ ou se $d_i = 0$, $\forall i \in A_k$, então $V_k = 0$, $k=1,...,K-1$, restando apenas os itens livres para serem alocados.)*

Passo 3:

3.1 Escolha k^* tal que: $Max\{\frac{V_k - c_k}{L_k},\ k = 1,\cdots,K-1;\ \frac{v_k}{l_k},\ k \in A_K\}$

3.2 Atualização:

Se k^* em (3.1) corresponder a um compartimento

então $lutil \leftarrow lutil - L_k$. "alocar um compartimento"

$d_i \leftarrow d_i - \alpha_{ik^*}$

senão $lutil \leftarrow lutil - l_k$. "alocar um item livre"

$d_i \leftarrow d_i - 1$

3.3 Se $lutil \geq Min\{Min\{L_k, k=1,...,K-1\}, Min\{l_i, \forall i \in A_K\}\}$, volte ao Passo 2.

4.3 Heurística dos "z" Melhores Compartimentos

Esta heurística consiste basicamente na determinação dos *"z"* melhores compartimentos associados a cada agrupamento, ou seja, nas *"z"* melhores soluções dos Problemas da Mochila para cada agrupamento. A resolução de um problema de programação linear inteira considerando estes compartimentos como itens, juntamente com os itens livres, determina um plano de corte para a mochila compartimentada. O algoritmo é apresentado a seguir.

Algoritmo:

Passo 1: Selecionar os *"z"* melhores compartimentos para o agrupamento k determinando as "z" melhores soluções para o seguinte Problema da Mochila de capacidade L_{max}, sendo $V_{1k} \geq V_{2k} \geq ... \geq V_{zk}$ os "z" melhores valores de (16.1) e suas respectivas soluções α_{ijk}. Para $k=1,...,K-1$ e $j=1,...,z$, resolva:

$$V_{jk} = Maximizar \quad \sum_{i \in A_k} v_i \alpha_{ijk} \qquad (16.1)$$

$$Sujeito\ a: \quad \sum_{i \in A_k} l_i \alpha_{ijk} + S \leq Lmax \qquad (16.2)$$

$$0 \leq \alpha_{ijk} \leq d_i \ e\ inteiro,\ para\ i \in A_k. \qquad (16.3)$$

Passo 2: Resolver o seguinte problema de programação inteira envolvendo os $((K-1)*z)$ compartimentos selecionados e os $|A_K|$ itens livres:

- β_{jk} é o número de vezes que o compartimento j para o agrupamento k é repetido na mochila,

- γ_k é o número de vezes que o item livre k é repetido na mochila.

$$\text{Maximizar} \quad \sum_{k=1}^{K-1}\sum_{j=1}^{z}(V_{jk}-c_k)\beta_{jk}+\sum_{k\in A_K}v_k\gamma_k \qquad (17.1)$$

$$\text{Sujeito a:} \quad \sum_{k=1}^{K-1}\sum_{j=1}^{z}L_{jk}\beta_{jk}+\sum_{k\in A_K}l_k\beta_k \le L-S \qquad (17.2)$$

$$\sum_{j=1}^{z}\alpha_{ijk}\beta_{jk}\le d_i, \quad i\in A_k \; e \; k=1,\dots,K-1. \qquad (17.3)$$

$$0\le\gamma_k\le d_k, \quad k\in A_K. \qquad (17.4)$$

$$\beta_{jk}\ge 0, \text{ inteiro}, \quad k=1,\dots,K-1 \; e \; j=1,\dots,z. \qquad (17.5)$$

$$\text{onde } L_{jk}=\sum_{i\in A_k}l_i\alpha_{ijk}+S, \quad k=1,\dots,K-1.$$

Como podemos notar, o passo 2 explora uma resolução heurística para o problema (*17.**), não obtendo assim a solução ótima.

5. Resultados Computacionais

As heurísticas apresentadas na seção 4 foram testadas para uma série de exemplos gerados aleatoriamente, classificados de acordo com as seguintes 4 características: i) número de itens livres; ii) comprimentos dos itens; iii) limites sobre o número de itens dos agrupamentos e, iv) limites sobre o número de itens livres. Tais características tentam simular situações observadas com dados reais. A seguir apresentaremos as características que foram consideradas para cada conjunto de dados. Alguns parâmetros foram fixados para todos os exemplos, representando dados reais. Adotaremos o comprimento (e não o peso, como feito até agora) como característica de ocupação dos itens na mochila.

- **Parâmetros fixados:**
  1. Capacidade da Mochila: $L = 1200$ mm;
  2. Capacidades mínima e máxima para os compartimentos: $Lmin = 154$ mm e $Lmax = 456$ m.
  3. Perda devido à inclusão de compartimento (imperfeições nas bordas): $S = 12$ mm.
  4. Número de agrupamentos gerados aleatoriamente: $K\in[3, 8]$.
  5. Número de itens por agrupamento gerado aleatoriamente: $|A_k|\in[2, 6]$.
  6. $v_i = l_i + \sigma$, $\sigma\in[1,100]$, (Obs: Na geração de colunas, cada mochila compartimentada, quando a função objetivo é minimizar a perda, terá os valores de utilidade neste formato, onde σ é o multiplicador simplex).
  7. $c_k\in[1,100]$.

- **Parâmetros variados: (para cada parâmetro, consideramos duas possibilidades)**

i) Número de Itens Livres:

1: número pequeno de itens livres: $|A_K|\in[1, 3]$

2: número grande de itens livres: $|A_K|\in[4, 10]$

ii) Comprimentos dos Itens:

3: Itens pequenos: $l_i\in[36 \text{ mm}, 105 \text{ mm}]$.

4: Itens grandes e pequenos: $l_i\in[36 \text{ mm}, 444 \text{ mm}]$.

(Obs: na indústria, os comprimentos dos itens variam bastante de modo que a situação 4 melhor representa a realidade; a situação 3 explora o pior caso).

iii) Limites sobre o Número de Itens dos Agrupamentos:

5: limitante baixo: $d_i \in [1, 3]$, $i \in A_k$, $k = 1, \ldots, K\text{-}1$.

6: limitante alto: $d_i \in [4, 15]$, $i \in A_k$, $k = 1, \ldots, K\text{-}1$.

iv) Limites sobre o Número de Itens Livres:

7: limitante baixo: $d_i \in [1, 3]$, $i \in A_K$.

8: limitante alto: $d_i \in [4, 15]$, $i \in A_K$.

Na indústria metalúrgica, onde surge o *Problema da Mochila Compartimentada*, o número de itens associados aos agrupamentos corresponde a 40% do total dos itens a serem alocados na mochila. Porém, embora as heurísticas descritas na seção 4 sejam projetadas para uma única mochila, elas devem ser executadas em outras ocasiões onde as várias situações acima podem ocorrer (por exemplo, na técnica de geração de colunas, uma mochila compartimentada deve ser resolvida a cada iteração do método simplex ou como geradora de padrões eficientes na heurística de exaustão [Hinxman,1980]).

A seguir, é feito um resumo dos resultados obtidos para vários exemplos gerados. Para cada situação combinada 20 exemplos foram gerados aleatoriamente e executados para as 3 heurísticas (Decomposição, Melhor Compartimento e "z" Melhores Compartimentos com $z = 2$).

5.1 Resultados Obtidos

A seguir será apresentada uma tabela que mostra os valores médios obtidos para a função objetivo nos 20 exemplos executados para cada conjunto de dados.

	Parâmetros dos Conjuntos	Decomposição	Melhor Compartimento	"z" Melhores Compartimentos	Todas as Heurísticas
Conjunto A	1,3,5,7	2212,50	2143,00	2298,40	2218,60
Conjunto B	1,3,5,8	2172,00	2140,30	2206,70	2173,65
Conjunto C	1,3,6,7	2400,90	2647,70	2439,70	2496,79
Conjunto D	1,3,6,8	2850,40	2895,50	2875,20	2874,37
Conjunto E	1,4,5,7	1549,30	1578,90	1679,30	1603,12
Conjunto F	1,4,5,8	2032,40	1983,60	2044,50	2020,82
Conjunto G	1,4,6,7	1920,70	1939,20	2075,50	1979,09
Conjunto H	1,4,6,8	2138,30	2193,10	2180,40	2171,26
Conjunto I	2,3,5,7	2376,40	2364,90	2412,50	2385,26
Conjunto J	2,3,5,8	2594,10	2465,80	2602,10	2554,65
Conjunto K	2,3,6,7	2779,50	2702,20	2806,20	2763,28
Conjunto L	2,3,6,8	2988,20	2937,60	3001,90	2976,56
Conjunto M	2,4,5,7	2007,10	1920,90	2038,20	1989,38
Conjunto N	2,4,5,8	2474,20	2480,80	2474,20	2477,07
Conjunto O	2,4,6,7	1949,00	1914,40	2055,80	1973,69
Conjunto P	2,4,6,8	2476,00	2319,20	2537,10	2444,73
Média Total	**Todas as Combinações**	**2307,56**	**2289,19**	**2357,98**	**2318,90**

Tabela 1: Valores Médios da Função Objetivo Obtidos Executando os Conjuntos de Dados para as Heurísticas.

Baseado nesta tabela, um gráfico (*gráfico 1* - na seqüência) foi construído. Neste gráfico, traçamos um comparativo entre os valores médios obtidos para a função objetivo (vistos na tabela 1) considerando as heurísticas executadas e, também, um comparativo com os valores considerando todas as heurísticas. Para tal, consideramos como base os valores médios obtidos para a heurística dos "z" Melhores Compartimentos (correspondendo a 100%) e de acordo com os valores obtidos nas outras heurísticas, verificamos se eles estão acima ou abaixo dos valores base. O gráfico 1 ilustra esta comparação.

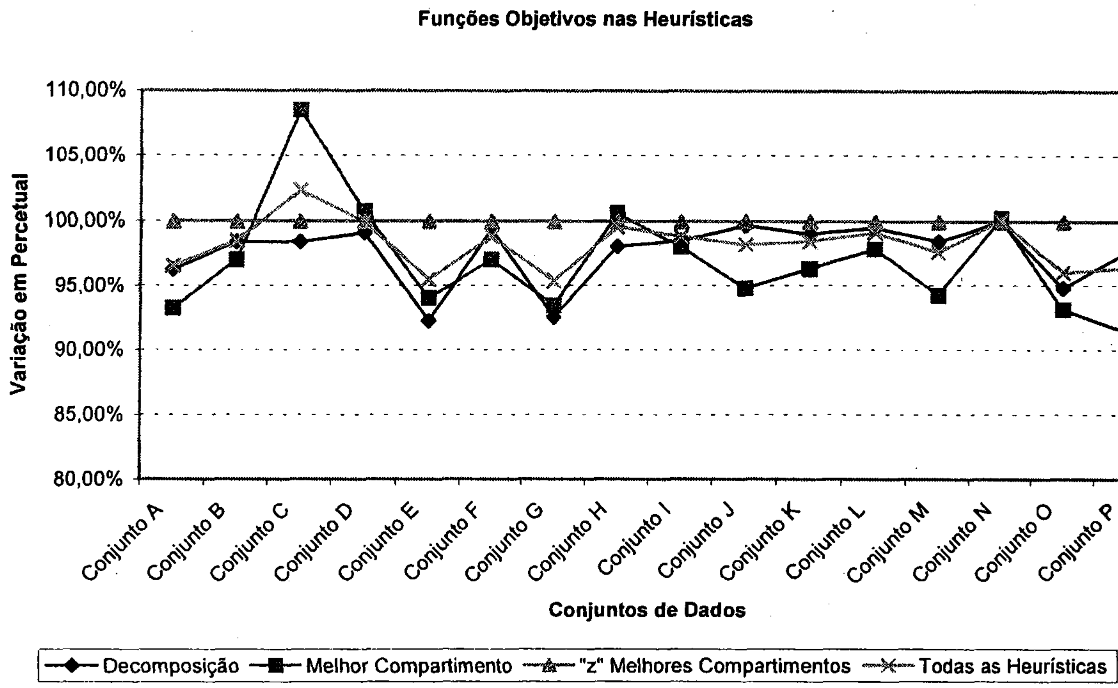

Gráfico 1: Comparativo entre os valores obtidos para a função objetivo nas diversas heurísticas.

5.2 Análise dos Resultados Obtidos

A execução de vários exemplos permitiu-nos concluir que as heurísticas possuem, de modo geral, desempenhos bastante semelhantes. A heurística dos "z" Melhores Compartimentos obteve um desempenho ligeiramente destacado em relação às demais, embora em alguns casos a Heurística do Melhor Compartimento tenha obtido resultados um pouco melhores.

Tomando como exemplo o conjunto de dados C, cujos parâmetros para a execução envolviam algumas características interessantes como: poucos itens livres (parâmetro 1), itens pequenos (parâmetro3), grande quantidade permitida de itens associados aos agrupamentos (parâmetro 6) e pequena quantidade permitida de itens livres, podemos observar que a Heurística do Melhor Compartimento obteve, para esse conjunto, os resultados melhores. Tal combinação, de uma forma geral, simula a situação onde os itens que devem ser alocados na mochila são relativamente pequenos e a quantidade de itens pertencentes aos agrupamentos é bem maior que a quantidade de itens livres. O melhor desempenho da heurística do Melhor Compartimento para esse caso (e a melhora do desempenho para alguns conjuntos semelhantes) pode ser explicado pelo fato desta heurística apostar, a cada iteração, na alocação (uma única vez) do melhor compartimento ou item livre. Nestas circunstâncias, a alocação de um compartimento favorece o fato de que, na iteração seguinte, um novo compartimento diferenciado (do mesmo agrupamento) possa ser gerado de forma que tal compartimento tenha, ainda assim, valor de utilidade maior que os demais que estão concorrendo na mochila. Tal possibilidade não é considerada nas demais heurísticas e, devido à situação imposta, pode ser um fator decisivo no desempenho desta heurística. O fato é que itens livres pequenos e em baixa quantidade e limitação não são capazes de preencher uma mochila, obrigando a heurística a alocar compartimentos e forçando-a a gerar novos compartimentos diferenciados. Vale salientar, ainda, que pequenas variações nessas características podem gerar resultados semelhantes (por exemplo, o conjunto de dados D).

Apesar de alguns contra-exemplos, a Heurística dos "z" Melhores Compartimentos teve um desempenho superior na maioria dos casos. Isto já era de se esperar, pois tal heurística trabalha com um poder combinatório maior que as demais para o preenchimento da mochila (maior número de compartimentos são gerados para cada agrupamento).

Quanto a Heurística de Decomposição, por se tratar de um caso particular da Heurística dos "z" Melhores Compartimentos (onde $z = 1$), já era esperado que não obtivesse resultados tão destacados. Porém, ainda assim, esta heurística não perdeu tanto em desempenho para as demais, sendo, inclusive, melhor que a Heurística do Melhor Compartimento quando consideramos as todos os exemplos executados para todas as combinações.

Em relação ao tempo de execução, os resultados têm mostrado que as heurísticas rodam, na maioria dos casos, muito bem, com tempos semelhantes e bastante reduzidos. Tal característica torna-se importante se desejarmos usar estas heurísticas na geração de colunas para um problema de corte de estoque (onde demandas devem ser satisfeitas).

6. Conclusões e Perspectivas Futuras

Este trabalho abordou o *Problema da Mochila Compartimentada*, um problema de otimização combinatória, que é uma variação do clássico Problema da Mochila (porém, sua formulação matemática sofre profundas mudanças) e de importantes aplicações práticas como na produção de tubos de aço na indústria metalúrgica. Um modelo matemático de otimização não linear inteiro foi proposto e nos permitiu definir um conjunto de métodos heurísticos para a sua resolução. Porém, vale salientar que novas propostas de resolução do problema vêm sendo analisadas. Dentre as novas propostas, destaca-se a [Hoto *et al.*,2001] onde todos os possíveis compartimentos para os agrupamentos envolvidos são obtidos e, a partir daí, a mochila compartimentada é preenchida.

Uma nova heurística está em desenvolvimento considerando-se a possibilidade de gerar para cada agrupamento, "z" compartimentos distintos (i.e., distintas capacidades para os compartimentos, ao invés de fixar em L_{max}), o que melhoraria o poder combinatório para o preenchimento da mochila. Além disso, um método de geração de colunas para o problema de corte de estoque, onde demandas dos itens devem ser satisfeitas e os padrões de corte são mochilas compartimentadas vem sendo desenvolvido.

Uma outra continuação deste trabalho consiste em estender o problema aqui estudado para o caso de diversos tipos de mochilas com características diferentes (como, por exemplo, capacidades variadas). Isto levaria a considerar custos (c_k) diferenciados de cada agrupamento a cada tipo de mochila.

Outra proposta de pesquisa futura a ser considerada é a abordagem por grafos E/OU, já que em [Vianna,2000] foi mostrado que diferentes problemas de corte (i.e., diferentes formas geométricas, processos, etc.) podem ser representados como um caminho completo num grafo E/OU.

Por fim, poderíamos estudar o *Problema da Mochila Compartimentada* para casos bidimensionais envolvendo situações encontradas na prática como, por exemplo, o problema de corte de placas de circuitos impressos (os compartimentos são pedaços menores da placa original, sobre os quais incidem diferentes processos); problemas tridimensionais, como no caso de empacotamento de caixas que devem estar agrupadas (compartimentadas) por cliente.

Apêndice A: Aplicações no Corte de Aço

Descrevemos aqui o corte em bobinas de aço, sujeito à laminação, o que foi a motivação inicial deste estudo, procurando enfocar os principais aspectos envolvidos no processo. Além disso, fazemos uma descrição dos custos envolvidos no processo de produção.

Este problema foi observado em uma empresa que produz tubos de aço para diversas aplicações. A linha de produção consiste em produzir fitas, a partir de bobinas de aço em estoque. Essas fitas, por sua vez, serão utilizadas na confecção dos tubos soldados, que terão finalidades específicas.

1. Considerações Iniciais

Antes de prosseguir, vamos consideremos as seguintes definições encontradas em [Ferreira *et al.*,1990], [Carvalho,1991], [Pereira,1993] [Hoto,1996] e [Hoto,2001].

Definições:

- *Bobinas mestres* são os *objetos* a serem cortados (mochilas a serem preenchidas). Tais bobinas são identificadas pelos seus pesos (entre 12000 e 13500 Kg), larguras (variando de 1100 a 1200 mm), espessura do aço e pelo teor de carbono do aço segundo os critérios do *SAE* (*Society of Automotive Engineers*). Os mais utilizados são: SAE 1008, SAE 1010, SAE 1012, SAE 1017 e SAE 1021.
- *Bobinas intermediárias* são as bobinas obtidas durante a primeira etapa de corte (os compartimentos). As bobinas intermediárias herdam algumas de suas características das bobinas mestres, como a espessura e o teor de carbono do aço. Porém, por sua vez, as bobinas intermediárias terão seus próprios pesos e larguras.
- *Fitas* são os *itens* obtidos durante a segunda etapa de corte, a partir das bobinas intermediárias. As fitas possuem características bem definidas como a largura (de acordo com o diâmetro dos tubos a serem produzidos), a espessura e o tipo de aço.

Observe que uma bobina com o tipo de aço SAE 1008 somente poderá produzir fitas com esta característica. No que se segue, supomos que o conjunto de fitas selecionado para ser produzido tenha as características necessárias de uma mesma bobina mestre a ser cortada.

2. O Processo de Corte

Uma peculiaridade do problema ao se efetuar cortes em bobinas de aço sujeitas a um processo técnico de laminação, reside no fato de que as bobinas mestres não podem ser laminadas devido a sua largura, pois existem restrições físicas, impostas pelo laminador, estabelecendo limites nas larguras das bobinas para que essas possam ser laminadas (veja seção 4 deste apêndice). Portanto, as bobinas mestres devem ser cortadas em bobinas intermediárias, cujas larguras respeitem as restrições do laminador e, posteriormente, recortadas em "fitas". Além disso, vale enfatizar que o processo de laminação é executado visando ajustar a espessura das bobinas intermediárias de acordo com as fitas demandadas (na prática, cerca de 40% da produção necessita de laminação).

Sendo assim, temos duas etapas de corte bem caracterizadas, definindo um compartimento de fitas (bobinas intermediárias deverão produzir fitas de mesma espessura) na construção de um plano de corte. A figura 1 dá uma idéia mais clara do processo de corte das bobinas.

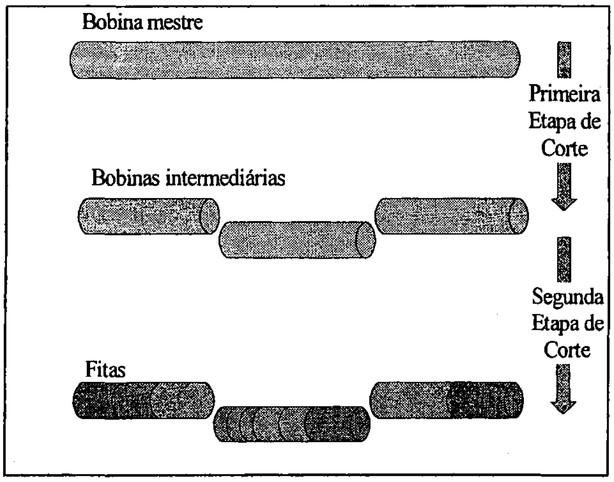

Figura 1: O processo de corte em bobinas de aço.

Denomina-se *"Refugo"* todo tipo de *perda inevitável* de material durante o processo de obtenção das fitas, e *"Sobra"* todas as outras categorias de perdas de material. Temos dois refugos intrínsecos: um durante a primeira etapa de corte e outro durante a segunda etapa de corte [Ferreira *et al.*,1990]. A figura 2 ilustra melhor este fato:

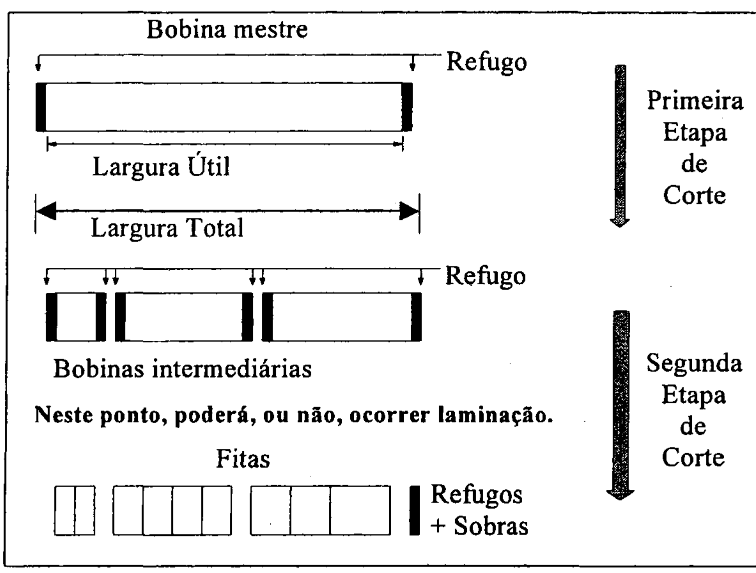

Figura 2: Corte em bobinas de aço com refugos embutidos.

Na prática, por uma restrição no número de facas, o número de bobinas intermediárias na primeira etapa é limitado.

Uma dificuldade no processo de cortagem decorre da existência de demandas baixas, de maneira que a utilização de uma bobina mestre com peso consideravelmente alto poderá proporcionar excesso de produção, mesmo que apenas uma fita de um determinado tipo apareça no plano de corte. Para evitar este excesso, [Hoto,2001] considera uma formulação 1.5-dimensional, introduzindo uma variável no comprimento da bobina mestre. Embora essa abordagem pareça interessante, há uma resistência operacional que a impede de ser implementada na prática, portanto, não vamos considerá-la neste trabalho.

3. A Cortadeira

A grosso modo, uma bobina mestre é desenrolada e o processo de corte para obtenção das bobinas intermediárias é feito longitudinalmente por "discos de corte" (não há cortes transversais e por isso podemos entender que o problema é unidimensional). Nesta etapa, existem perdas intrínsecas (que podem ser visualizadas na figura 3 como refugos) nas laterais da bobina, para eliminar as irregularidades das bordas, variando entre 6 a 8 mm. Posteriormente, as bobinas intermediárias são enroladas e passarão, ou não, para o processo de laminação se as espessuras das fitas forem idênticas à espessura da bobina mestre, não será necessário o processo de laminação.

O tempo de preparação da máquina de corte é cerca de 40 minutos e o tempo médio para o corte de uma bobina gira em torno de 15 minutos. A figura 3 ilustra o processo de corte.

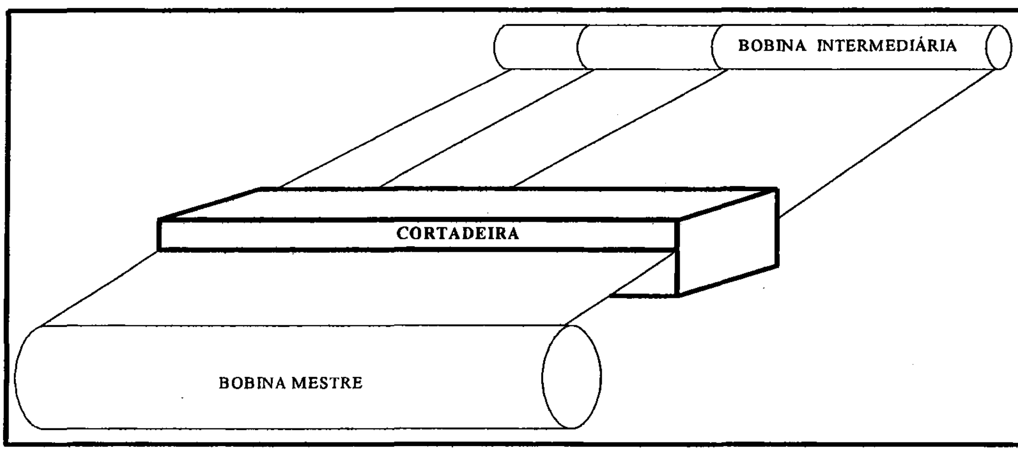

Figura 3: Perfil do processo de cortagem em bobinas de aço.

4. Laminação, Encruador, Recortadeira e Recozimento

O processo de laminação tem por objetivo diminuir (quando necessário) a espessura do material de uma bobina intermediária. Esse processo se dá à temperatura ambiente (laminação a frio) por meio de cilindros de laminação que efetuam pressão sobre a lâmina de aço (figura 4). Cabe notar que, eventualmente, uma mesma bobina intermediária poderá sofrer mais de uma laminação. Isto é necessário para que se possa obter a espessura desejada do material (a redução é cerca de 15 a 20% da espessura em cada passo, e de 50 a 60% no total). Como todas as outras, a máquina que executará a laminação (o laminador) possui restrições do tipo: Não é possível laminar uma bobina intermediária com menos de 154 mm e mais de 456 mm de largura (isto limita os tamanhos dos compartimentos, conforme definição do *Problema da Mochila Compartimentada*, seção 3).

Após a última laminação, o material da bobina adquiriu algumas imperfeições superficiais que são corrigidas num aparelho denominado encruador. Este aparelho nada mais é que um laminador que, ao contrário do convencional, apenas corrige, através de uma laminação suave, as imperfeições da superfície da bobina. A figura 4 ilustra o processo de laminação em perfil.

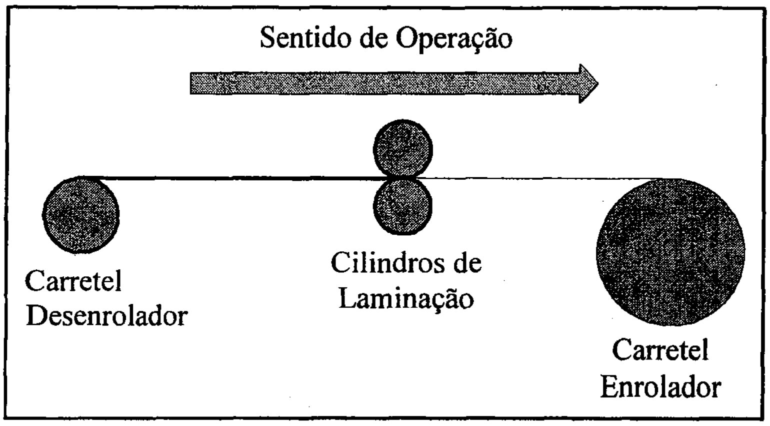

Figura 4: Perfil do processo de laminação.

Em seguida, a bobina intermediária é submetida a recortadeira, uma máquina especial, semelhante à cortadeira, porém preparada para que se obtenha as fitas. Nessa máquina, os cortes são limitados num total de 8 e, como na cortadeira, existe uma perda intrínseca nas laterais das bobinas intermediárias variando de 3 a 5 mm. O limite de largura imposto pelas máquinas que farão o recorte é de 600mm.

Devemos salientar que, quando submetemos uma mesma bobina intermediária ao processo de laminação, o material adquiri alguns fatores físicos indesejáveis. Em face desse inconveniente é necessário, após o recorte, um tratamento térmico ao material denominado recozimento. No recozimento, as fitas são acondicionadas em fornos especiais onde permanecem por um certo período e, em seguida, são lentamente resfriadas até atingirem a temperatura ambiente. O processo todo leva cerca de 24 horas e constitui um gargalo na produção. Daí a importância de se laminar o mínimo possível.

Basicamente, [Hoto,2001] resume o fluxo da produção pelo esquema representado na figura 5. Entretanto vale salientar a possibilidade de haver pequenas alterações.

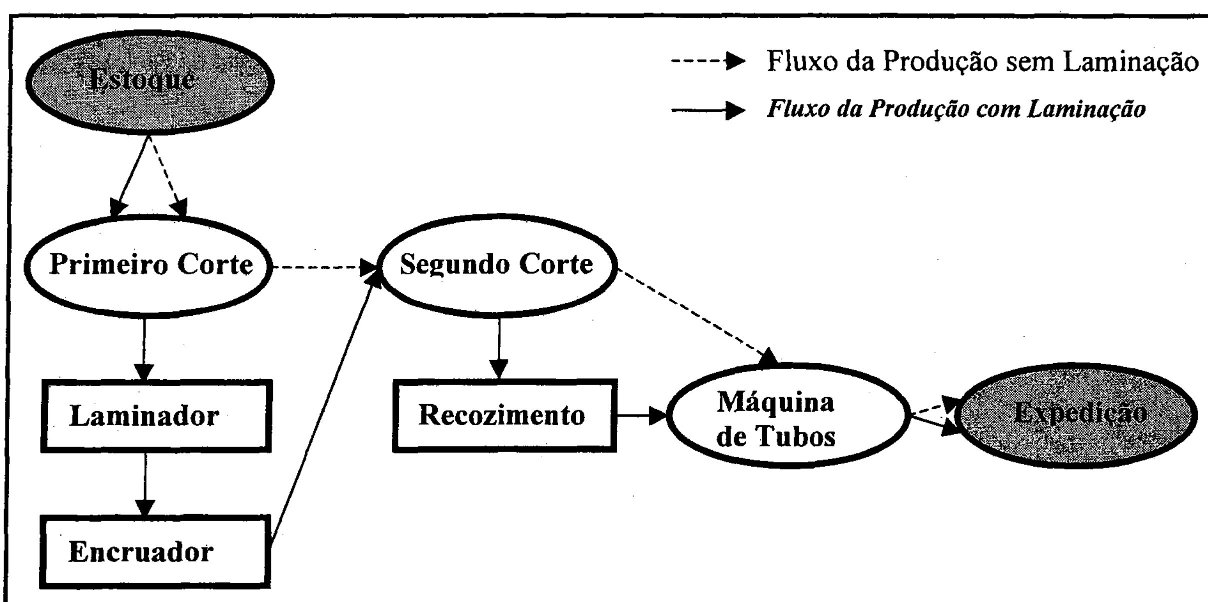

Figura 5: Fluxo da produção de tubos de aço.

5. Custos dos Processos Envolvidos no Problema

Como visto, custos inerentes ao processo e relativos às perdas de material são típicos do problema. Dentre os custos inerentes, podemos destacar: o custo de laminação; custos devido a refugos que, de uma maneira geral, estão intimamente ligados aos tipos de equipamentos disponíveis e a fatores tecnológicos; e custos devido aos acertos de corte (troca de padrões).

O custo de laminação é uma preocupação fundamental, e nesse sentido, tenta-se evitar o máximo possível este tipo de operação, pois laminar, como já ressaltamos, é um processo caro. Naturalmente, o processo de laminação nem sempre poderá ser evitado, portanto, nesse caso é interessante buscar uma minimização deste processo [Hoto,1996].

Em síntese, as diversas peculiaridades da linha de produção aqui citadas apontam para um problema complexo. Além disso, a necessidade do processo de laminação induz planos de corte pouco evidentes, pois nem todas as fitas têm a mesma espessura. Uma empresa na capital paulista que trabalha com este problema indica uma perda de material da ordem de 10%, o que corresponde a uma média mensal de 500 toneladas de aço [Hoto,2001].

Apêndice B: O Problema de Corte de Estoque

Na seção 4, apresentamos heurísticas para o problema de corte envolvendo um único objeto (por exemplo, uma mochila ou bobina). Entretanto, na prática, uma demanda de itens deve ser atendida obrigando o corte de muitas bobinas (ou preenchimento de muitas mochilas). Este problema é chamado na literatura de *problema de corte de estoque* [Dyckhoff e Finke,1992]. Para altas demandas, onde um plano de corte gerado para uma bobina deva ser repetido muitas vezes, [Gilmore e Gomory,1961,1963,1965] propuseram um modelo de programação linear com um número muito grande de colunas e a técnica de geração de colunas (método simplex) para resolvê-lo. A cada iteração do método simplex um problema de corte (por exemplo, um *Problema da Mochila Compartimentada*) deve ser resolvido para encontrar um plano de corte, onde os valores de utilidade (v_i) são funções dos multiplicadores simplex da iteração (veja [Arenales e Morabito,1997]).

Quando a demanda não é muito alta, de modo que poucas repetições (1 ou 2) dos planos de corte ocorrerão, é usual o emprego de uma heurística de repetição exaustiva [Hinxman,1980], que consiste numa heurística gulosa da seguinte forma:

Passo 1: Gere o melhor plano de corte que não exceda a demanda (*o Problema da Mochila Compartimentada*);
Passo 2: Utilize esse plano uma vez (ou tantas vezes quanto possível);
Passo 3: Atualize a demanda e repita o Passo 1, enquanto houver demanda a ser satisfeita.

Observe que no Passo 1, podemos usar uma das heurísticas apresentadas na seção 4. Deixaremos para trabalhos futuros a proposta do problema de corte de estoque anterior.

Bibliografia

[Arenales e Morabito,1997] Arenales, M.N. e Morabito, R. (editores) (1997) O Problema de Corte e Empacotamento e Aplicações Industriais, *XX Congresso Nacional de Matemática Aplicada e Computacional*, Gramado-RS.

[Beasley,1985] Beasley, J. (1985) Algorithms for Unconstrained Two-dimensional Guillotine Cutting, *Journal of Operational Research Society*, 36:297-306.

[Carvalho,1991] Carvalho, J.M.V. (1991) Um Problema de Corte em Duas Fases, Tese de Doutoramento, Universidade do Minho, Portugal.

[Carvalho e Rodrigues,1994] Carvalho, J.M.V. and Rodrigues, A.J.G. (1994) A Computer Based Interective Approach to a Two-stage Cutting Stock Problem, *INFOR*, 32(4):243-252.

[Chvàtal,1980] Chvàtal, V. (1980) *Linear Programming*, W.H.Freeman and Company, New York.

[Dowsland e Dowsland,1992] Dowsland, K.A. and Dowsland, W.B. (1992) Packing Problems, *European Journal of Operational Research*, 56:02-14.

[Dyckhoff e Finke,1992] Dyckhoff, H. and Finke, U. (1992) *Cutting and Packing in Production and Distribuition*. Springer-Verlag Co., Heidelberg.

[Ferreira *et al.*,1990] Ferreira, J.S., Neves, M.A. and Castro, P.F. (1990) A Two-phase Roll Cutting Problem, *European Journal of Operational Research*, 44:185-196.

[Garey e Johnson,1979] Garey, M.R. e Johson, D.S. (1979) *Computers and Intracrability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Co., San Francisco, 1979.

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

[Gilmore e Gomory,1963] Gilmore, P. and Gomory, R. (1963) A Linear Programming Approach to the Cutting Stock Problem, parth II, *Operations Research*, 14:94-120.

[Gilmore e Gomory,1965] Gilmore, P. and Gomory, R. (1965) MultiStage Cutting Stock Problems os Two and More Dimensions, *Operations Research*, 14:1045-1074.

[Herz,1972] Hertz, J. (1972) Recursive Computational Procedure for Two-dimensional Stock Cutting, *IBM Journal of Research and Development*, 16:462-469.

[Hinxman,1980] Hinxman, A.I. (1980) Trim-Loss and Assortment Problems: A Survey, *European Journal of Operational Research*, 5: 8-18.

[Hoto,1996] Hoto, R.S.V. (1996) Otimização no Corte de Peças Unidimensionais com Restrições de Agrupamento, Dissertação de Mestrado, ICMSC-USP, São Carlos, São Paulo.

[Hoto e Arenales,1996] Hoto, R.S.V. e Arenales, M.N. (1996) Um Problema de Corte Unidimensional com Restrições de Agrupamento e Aplicações Industriais, *I Oficina Nacional em Problemas de Corte e Empacotamento*, São Paulo - SP, 31-36.

[Hoto e Arenales,1997] Hoto, R.S.V. e Arenales, M.N. (1997) O Problema do Corte em Bobinas de Aço, *XX CNMAC*, Gramado - RS, 1997, pp 545,546.

[Hoto *et al.*,1997] Hoto, R.S.V., Maculan, N. e Arenales, M.N. (1998) O Problema do Corte em Bobinas de Aço Via Geração de Colunas, *XX CNMAC*, Gramado - RS, 1997, pp 545,546.

[Hoto,2001] Hoto, R.S.V. (2001) O Problema da Mochila Compartimentada aplicado no Corte de Bobinas de Aço. Tese de Doutoramento, COPPE-UFRJ, Rio de Janeiro, Rio de Janeiro.

[Hoto *et al.*,2001] Hoto, R.S.V., Maculan, N., Arenales, M.N. e Marques, F.P. (2001) Um Novo Procedimento para o Cálculo de Mochilas Compartimentadas, *XXXII Simpósio Brasileiro de Pesquisa Operacional*, Campos do Jordão-SP.

[Lin,1998] Lin, E.Y.H. (1998). A Bibliographical Survey on Some Well-Known Non-Standard Knapsack Problems. *INFOR*, 4, 36 (1998), 274-317.

[Marques,2000] Marques, F.P. (2000). O Problema da Mochila Compartimentada, Dissertação de Mestrado, ICMC-USP, São Carlos, São Paulo.

[Martello e Toth,1990] Martello, S. e Toth, P. (1990) *Knapsack Problems Algorithms and Computer Implementations*, John Wiley & Sons, Chichester.

[Morabito,1992] Morabito, R. (1992) Uma abordagem em grafo E/OU para o problema de empacotamento: Aplicações ao Carregamento de Paletes e Contêineres. Tese de Doutoramento, EESC-USP, São Carlos, São Paulo.

[Pereira,1993] Pereira, M.A. (1993) Uma Abordagem Matemática para o Problema de Corte e Laminação de Fitas de Aço, Dissertação de Mestrado, UNICAMP, Campinas, São Paulo.

[Soma *et al.*,1997] Soma, N.; Yanasse, H. e Maculan, N. (1997) O Problema de Corte e Empacotamento e Aplicações Industriais: O Problema da Mochila, *XX Congresso Nacional de Matemática Aplicada e Computacional*, Gramado-RS, 24-58.

[Sweeney e Paternoster,1992] Sweeney, P. and Paternoster, E. (1992) Cutting and Packing Problems : A Categorized, Aplication-Oriented Research Bibliography, *Journal of the Operational Research Society*, 43:691-706.

[Vianna,2000] Vianna, A.C.G. (2000). Problemas de corte e empacotamento:uma abordagem em grafo E/OU, Tese de Doutorado, ICMC-USP, São Carlos, São Paulo.

[Yanasse *et al.*,1997] Yanasse, H.H., Soma, N.Y. e Maculan, N. (1997) An Algorithm for Determining the K-best Solutions of the One-dimensional Knapsack Problem. Submetido para *Discrete Applied Mathematics*.

NOTAS DO ICMC

SÉRIE COMPUTAÇÃO

063/2001 TOMÉ, M F; MANGIAVACHI, N; CUMINATO, J A; CASTELO, A – A marker-and-cell technique for simulating unsteady viscoelastic free surface flows.

062/2001 VARGAS, A J C; NONATO, L G. – β-conexão: uma família de objetos tridimensionais reconstruídos a partir de seções planares.

061/2001 OLIVEIRA JR., O N; MARTINS, R T; RINO, L H M ; NUNES, M G V – O uso de interlínguas para comunicação via internet: o projeto UNL/Brasil.

060/2001 SILVA, E Q ; MOREIRA, D A – Use of software agents to the management of distance education courses over the internet.

059/2001 OLIVEIRA, M C F; LEVKOWITZ, H – Visual data exploration and mining: a survey.

058/2001 SOARES, M D; FORTES, R P M; MOREIRA, D A – Version–web : a tool for helping web pages version control.

057/2001 LIANG, Z; MACAU, E E N; OMAR, N - Scene Segmentation of the Chaotic Oscillator Network.

056/2000 BATISTA, G E A P A; CARVALHO, A C P L F; MONARD, M C – Applying one-sided selection to unbalanced datasets.

055/2000 NONATO, L G; MINGHIM, R.; OLIVEIRA, M C F; TAVARES, G – A novel approach for delaunay 3D reconstruction with a comparative analysis in the light of applications.

054/2000 MORSELLI JR, J C M; SANTANA, R H C; SANTANA, M J; ULSON, R S – An approach for dynamic swapping of distributed simulation synchronisation protocols.