

Um Algoritmo Enumerativo para Determinação de Padrões Tabuleiros Exatos e Restritos

Daniel Massaru Katsurayama

Instituto Nacional de Pesquisas Espaciais
massaru@lac.inpe.br

Horacio Hideki Yanasse

Instituto Nacional de Pesquisas Espaciais
horacio@lac.inpe.br

Resumo

Padrões tabuleiros pertencem a uma classe especial de padrões 2-estágios guilhotinados que consomem menos tempo de máquina para serem cortados, portanto, de interesse particular em ambientes de grande demanda. Propomos um algoritmo enumerativo para determinação de padrões tabuleiros exatos, onde o número de itens de cada tipo a ser produzido no padrão é limitado. Testes computacionais são apresentados para avaliar o desempenho do algoritmo proposto.

Palavras-chave: padrão tabuleiro, algoritmo enumerativo, padrões 2-estágios guilhotinado, problema de corte.

1. Introdução

Em algumas indústrias o corte de material é parte importante do processo de produção. Devido às características da máquina, alguns padrões de corte são cortados mais rapidamente que outros. Por exemplo, no processo de produção de corte de chapas retangulares de fibras de madeira reconstruída (vide, por exemplo, Belluzzo, 2002), pode acontecer das chapas serem giradas pela máquina de corte para que cortes transversais e longitudinais (cortes guilhotinados) ao longo do objeto sejam aplicados. O corte pode ser efetuado em vários estágios, onde cada estágio implica num giro de 90 graus do objeto ou parte dele, a ser cortado.

Padrões mais complexos podem ser cortados utilizando-se um maior número de estágios para se tentar reduzir ainda mais as perdas de material. No entanto, estes padrões exigem um maior número de giros da chapa aumentando o tempo de corte dos objetos. Belluzzo (2002), por exemplo, estima em cerca de 2 minutos o tempo necessário para que a máquina considerada em seus estudos efetue cada giro. É desejável, portanto, utilizar padrões de corte mais simples para se reduzir o tempo de corte dos objetos.

Padrões tabuleiros, também conhecidos como padrões 1-grupo (Gilmore & Gomory, 1965) pertencem a uma classe especial de padrões 2-estágios e podem ser

produzidos sem a necessidade de cortar separadamente cada uma das faixas obtidas no primeiro estágio de modo que apenas a serra (ou o objeto, vide Belluzzo, 2002) é girada em 90 graus após o corte do primeiro estágio. Na Figura 1, ilustra-se um exemplo de corte de padrão tabuleiro, onde as faixas obtidas no primeiro estágio são cortadas simultaneamente no segundo estágio, produzindo os itens solicitados.

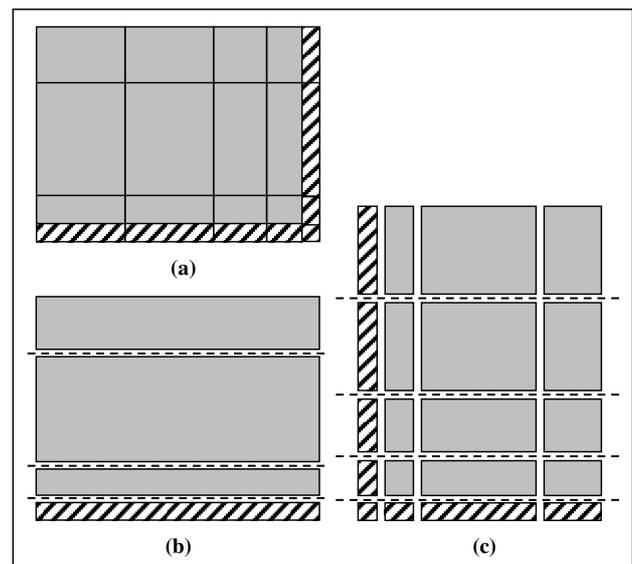


Figura 1. Corte de um padrão tabuleiro: (a) padrão tabuleiro final, (b) primeiro corte (primeiro estágio), (c) giro em 90 graus (segundo corte, segundo estágio)

Padrões tabuleiros, como o ilustrado na Figura 1, demandam menos tempo de máquina e são de particular interesse em ambientes de grande demanda, particularmente, quando a máquina de corte constitui-se num gargalo da produção. Os padrões tabuleiros são exatos quando todos os itens produzidos são obtidos imediatamente após o corte do segundo estágio, como ilustrado na Figura 1. Padrões tabuleiros podem também ser do tipo não exato (vide Figura 2), caso haja necessidade de se recortar algumas das peças obtidas no segundo estágio para produzir os itens finais solicitados. Neste caso, muitas vezes, o recorte é feito em outra

máquina, possivelmente menos carregada e mais simples, mas certamente incorrendo-se em custos adicionais.

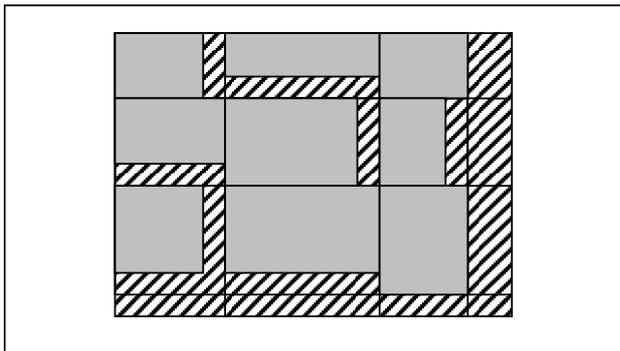


Figura 2. Padrão tabuleiro não exato

Embora a literatura sobre problemas de corte seja extensa (vide, por exemplo, Gilmore & Gomory, 1961, 1963, 1965; Hinxman, 1980; Dyckhoff, 1990; Dyckhoff & Waescher, 1990; Sweeney & Patternoster, 1992; Arenales & Morabito, 1997; entre outros), existem poucos trabalhos que tratam especificamente sobre determinação de padrões tabuleiros.

Morabito & Arenales (2000) formularam o problema da determinação de padrões tabuleiros exatos como um problema quadrático inteiro e propuseram uma heurística para sua obtenção. Scheithauer (2002) sugere que o problema da determinação de padrões tabuleiros exatos e não exatos seja formulado como um problema de programação linear inteira mista e resolvido diretamente através de um pacote de otimização. Em Yanasse & Morabito (2003) são apresentados modelos de programação linear inteira e não linear para o problema da determinação de padrões tabuleiros e 2-estágios exatos e não exatos. É também apresentada uma proposta de linearização da formulação de Morabito & Arenales (2000) para o problema da determinação de padrões tabuleiros exatos. Testes computacionais limitados com o pacote CPLEX também foram apresentados para avaliar alguns dos modelos propostos.

O primeiro algoritmo enumerativo exato para determinação de padrões tabuleiros exatos parece ter sido o proposto em Katsurayama & Yanasse (1999, 2000). Eles desenvolveram um algoritmo baseado no algoritmo de Yanasse, Soma & Maculan (2000) para determinação das K-melhores soluções para o problema da mochila unidimensional. O algoritmo proposto por estes autores se aplica a padrões tabuleiros exatos irrestritos e funciona em duas fases, primeiro, buscando uma melhor combinação nas faixas e, depois, buscando uma boa combinação das faixas. O algoritmo segue, portanto, o procedimento sugerido por Gilmore & Gomory (1965) para geração de padrões 2-estágios. Segundo os autores, a escolha pelo algoritmo de Yanasse, Soma & Maculan se deve ao fato de que a enumeração proposta por estes autores permite considerar restrições adicionais ao problema básico da mochila sem grande esforço computacional adicional. Isto

é particularmente interessante em ambientes de corte porque no processo de recuperação de soluções pode-se levar em conta se elas atendem a eventuais restrições adicionais impostas pelo problema, e cuidar para que apenas as soluções que atendam a tais restrições sejam recuperadas. No entanto, na prática, este procedimento mostra-se pouco eficiente para a determinação de padrões tabuleiros restritos, uma vez que apesar de ser fácil controlar-se a restrição dos itens em cada faixa, este controle é perdido quando se tenta combiná-las.

Neste trabalho, propomos um algoritmo para a determinação de padrões tabuleiros exatos, onde o número de itens de cada tipo a ser produzido no padrão é limitado. O algoritmo proposto também se baseia no algoritmo enumerativo de Yanasse, Soma & Maculan (2000) para determinação das K-melhores soluções para o problema da mochila unidimensional. Inicialmente, obtêm-se possíveis combinações de itens restritos para a resolução do problema de corte. Em seguida, verifica-se a possibilidade de produzir padrões tabuleiros com as combinações obtidas. Nosso objetivo é gerar padrões tabuleiros exatos visando otimizar uma função objetivo que reduza a perda de material, ou o custo de produção e que satisfaça a restrição imposta sobre os itens.

Testes computacionais são apresentados para avaliar o desempenho do algoritmo proposto e comparar os resultados obtidos com a literatura.

2. O Algoritmo Proposto

Considere um objeto retangular de dimensão $W \times L$ a ser cortado de modo a produzir diferentes tipos de itens retangulares menores de dimensão $w_i \times l_i$ e lucro (valor de utilidade) p_i , $i = 1, \dots, m$. O número de itens do tipo i no padrão é limitado por u_i . A área ocupada por cada um destes itens no padrão é dada por $a_i = w_i l_i$. Sabe-se que uma condição necessária que deve ser satisfeita por quaisquer combinações de itens é que a soma total das áreas de todos os itens não deve exceder a área do objeto, o que corresponde a resolver o seguinte problema relaxado:

$$\begin{aligned} &\text{Maximizar} && \sum_{i=1}^m p_i \lambda_i \\ &\text{Sujeito a:} && \sum_{i=1}^m a_i \lambda_i \leq WL, \\ &&& \lambda_i \leq u_i, \quad i = 1, \dots, m, \\ &&& \lambda_i \geq 0, \quad i = 1, \dots, m \text{ e inteiro.} \end{aligned}$$

No algoritmo proposto, encontra-se a melhor solução para o problema relaxado e verifica-se a possibilidade de gerar um padrão tabuleiro exato para a combinação $\lambda = (\lambda_1, \dots, \lambda_m)$ obtida. Se for possível a geração do

padrão, então a solução foi encontrada e o algoritmo é encerrado. Caso contrário, prossegue-se verificando a possibilidade de geração da segunda, da terceira, ..., da K-melhor combinação λ , até que o padrão tabuleiro exato seja obtido. Um resumo deste procedimento é apresentado na Figura 3:

Passo 1:	Construa uma lista com as K-melhores soluções (combinações) para o problema relaxado;
Passo 2:	Pegue a melhor combinação da lista construída;
Passo 3:	Tente gerar um padrão tabuleiro exato para a dada combinação. Se o padrão tabuleiro foi gerado com sucesso, então PARE ; Caso contrário, retorne ao Passo 2 .

Figura 3. Algoritmo para determinação de padrões tabuleiros exatos e restritos

Propomos, a seguir, um algoritmo para gerar um padrão tabuleiro.

3. Algoritmo para Geração de um Padrão Tabuleiro Exato a Partir de uma Combinação Dada de Itens

O algoritmo a ser apresentado pressupõe uma combinação dada de itens que poderiam estar no padrão, e fornece um padrão tabuleiro exato com estes itens, caso um exista ou conclui que não existe tal padrão.

No algoritmo tenta-se identificar a cada passo, que faixas horizontais e/ou verticais devem existir no padrão. Numa primeira fase, estas faixas são identificadas a partir dos itens que devem estar contidos no padrão, ou seja, se existe um padrão tabuleiro exato com a combinação $\lambda = (\lambda_1, \dots, \lambda_m)$, o padrão a ser gerado deve ser composto por faixas verticais e horizontais de comprimentos e larguras compatíveis com os comprimentos e larguras dos itens que fazem parte desta combinação, ou seja, faixas verticais e horizontais de comprimentos e larguras compatíveis com os itens correspondentes a $\lambda_k > 0$, $k = 1, \dots, m$. Seja Ω_w , o conjunto dos diferentes tipos de larguras w_k ($k = 1, \dots, m$) e, Ω_l , o conjunto dos diferentes tipos de comprimentos l_k ($k = 1, \dots, m$), dos itens existentes na combinação λ , com $\lambda_k > 0$, $k = 1, \dots, m$. O padrão tabuleiro, caso exista, deve ter no mínimo, uma faixa vertical de dimensão $W \times c$, para todo $c \in \Omega_l$, e uma faixa horizontal de dimensão $d \times L$, para todo $d \in \Omega_w$.

A segunda fase que se segue é aplicada recursivamente, cada vez que novas faixas verticais e /ou horizontais são definidas. As novas faixas são arranjadas ao longo da área disponível no objeto. As faixas verticais e horizontais são

arranjadas a partir do canto inferior esquerdo, conforme suas especificações. Com a sua inclusão são definidos de imediato os itens resultantes com os cortes das mesmas. Se o comprimento e/ou largura do objeto não for suficiente para se acomodar todas as faixas, então se conclui imediatamente que não será possível gerar o padrão com aquela combinação de itens e o algoritmo é encerrado. Caso contrário, uma vez arranjadas as faixas, determinamos quais itens resultam do corte das faixas e que fazem parte da combinação e quais são perdas de material. As faixas já definidas definem um retângulo de dimensão $D \times C$, localizado na parte inferior esquerda do objeto, que já faz parte da solução final do padrão tabuleiro exato, caso ele exista, como ilustrado na Figura 4.

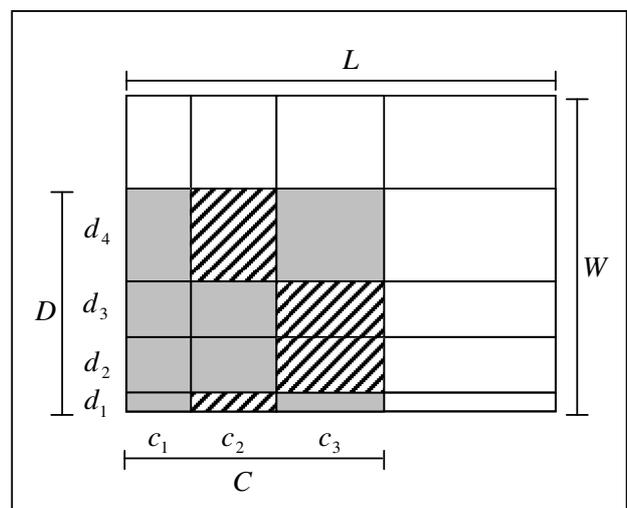


Figura 4. Construção da solução inicial

Sabemos que, caso seja possível gerar um padrão que atenda às restrições de maneira exata, então a perda estimada para o padrão final pode ser calculada através de

$$\gamma = WL - \sum_{k=1}^m \lambda_k w_k l_k .$$

Portanto, se a perda no retângulo $D \times C$ superar γ , concluímos também de imediato que não será possível gerar o padrão tabuleiro exato desejado e o algoritmo é encerrado. Caso contrário, a combinação λ dos itens ainda a serem cortados é atualizada subtraindo-se os itens já definidos do retângulo $D \times C$. Se não há mais itens na combinação, o padrão tabuleiro exato foi obtido e o algoritmo é encerrado, caso contrário, prosseguimos conforme o que se segue.

A proposta é tentar definir novas faixas verticais e horizontais necessárias para se completar a combinação λ restante. A definição destas novas faixas é feita resolvendo-se o problema do bin-packing unidimensional relativo aos "bins" localizados à direita e acima do retângulo $D \times C$. É importante destacar que os problemas de bin-packing não precisam ser resolvidos de maneira exata. Apenas um limitante simples é quase sempre

suficiente para estabelecer um número mínimo de “bins” a serem utilizados. Seja c , o comprimento do item r (para algum r), tal que $\lambda_r > 0$. Se, o número de “bins” já existentes de dimensão $(W - D) \times c$ não for suficiente para acomodar todos os itens restantes com comprimento c da combinação λ , então é necessário cortar pelo menos mais uma faixa vertical de dimensão $W \times c$.

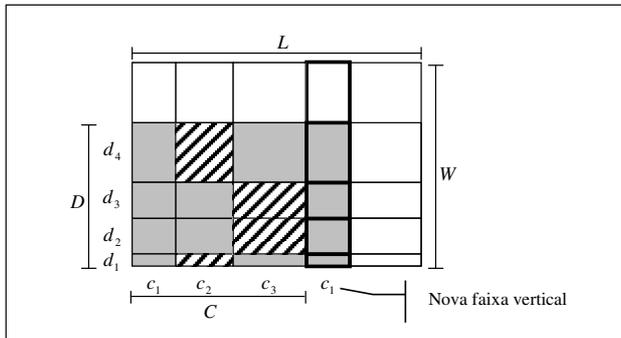


Figura 5. Corte de uma nova faixa vertical

De forma similar, seja d , a largura do item s (para algum s), tal que $\lambda_s > 0$, então uma faixa horizontal de dimensão $d \times L$ deverá ser cortada se o número de “bins” de dimensão $d \times (L - C)$ não for suficiente para acomodar todos os itens da combinação λ , tendo largura d .

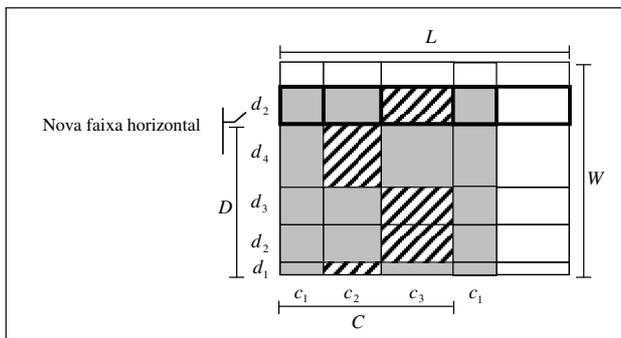


Figura 6. Corte de uma nova faixa horizontal

Se novas faixas verticais ou horizontais forem definidas, retorna-se a segunda fase do algoritmo. Novos itens serão cortados, novos espaços vazios de perdas serão possivelmente definidos, um novo retângulo $D \times C$ é definido, a combinação λ é atualizada e verificamos novamente se a geração do padrão tabuleiro ainda é possível.

Este procedimento é aplicado de forma iterativa até que se encontre o padrão tabuleiro, ou se conclua que não é possível obter-se um padrão tabuleiro com aquela combinação ou ainda, chega-se a conclusão de que todos os “bins” de dimensão $(W - D) \times c$ e de dimensão $d \times (L - C)$ são, cada um deles, isoladamente, suficientes para acomodar todos os itens restantes da combinação λ . Neste caso, é preciso verificar se eles também são

suficientes para cortar todos os itens restantes quando considerados de forma integrada. Entretanto, não se tem agora uma definição se os itens restantes deverão ser cortados de uma faixa vertical ou horizontal. Assim, inicia-se um procedimento de enumeração considerando as duas possibilidades possíveis, a inserção de algum item restante em um dos “bins” disponíveis de mesma largura ou em um “bin” de mesmo comprimento.

Por exemplo, suponha que inicialmente a inserção de um determinado item seja feita num “bin” acima do retângulo $D \times C$. Com isto, uma nova faixa horizontal, da largura do item inserido, será definida e podemos retornar à segunda fase do algoritmo. Com isso, o procedimento segue de forma recursiva, até que se encontra uma solução, e finalizamos o algoritmo ou, se não foi possível gerar o padrão através desta chamada recursiva, verificamos a outra possibilidade de inserção do item que é num “bin” à direita do retângulo $D \times C$. Neste caso, uma nova faixa vertical, do comprimento do item inserido, será definida e podemos retornar à segunda fase do algoritmo. O procedimento segue de forma recursiva até que se encontra uma solução, e finalizamos o algoritmo ou, se conclui que não é possível gerar um padrão tabuleiro com a combinação λ e o algoritmo é encerrado.

4. Testes Computacionais

Para realização dos testes computacionais sobre o algoritmo proposto, foram utilizados 10 exemplos numéricos extraídos de Yanasse & Morabito (2003). Os exemplos foram gerados aleatoriamente pelos autores para a determinação de padrões tabuleiros exatos. Nos exemplos, considera-se um objeto de dimensão 100×100 e $m = 5$ diferentes tipos de itens retangulares de dimensão $w_i \times l_i$, $i = 1, \dots, m$. Na Tabela 1, apresenta-se os valores das soluções e os tempos computacionais (em segundos) obtidos pelo algoritmo proposto para os exemplos testados. Os testes foram realizados numa estação de trabalho Sun UltraSPARC-II, 340Mhz, 256Mb RAM.

Tabela 1. Tempo computacional do algoritmo proposto para os exemplos extraídos de Yanasse & Morabito (2003)

Exemplo	Valor da Solução	Tempo (em segundos)
1	7800	8.80
2	8280	8.24
3	7000	5.01
4	7566	2.18
5	7936	8.68
6	9306	0.50
7	4770	9.94
8	7904	7.17
9	7800	1.37
10	5810	6.19

Para efeito de comparação dos resultados obtidos pelo algoritmo proposto, na Tabela 2, apresenta-se os tempos computacionais reportados por Yanasse & Morabito (2003) com a resolução dos modelos para determinação de padrões tabuleiros exatos propostos por estes autores. Estes autores utilizaram um microcomputador Pentium III, 1Ghz, 128Mb de RAM.

Tabela 2. Tempos computacionais obtidos resolvendo-se os Modelos 4 e 5 de Yanasse & Morabito (2003) (Fonte: Yanasse & Morabito, 2003)

Exemplo	Tempo (em segundos)	
	Yanasse & Morabito (2003)	Yanasse & Morabito (2003)
	Modelo 4	Modelo 5
1	0.32	12.57
2	0.35	1.35
3	0.29	1.96
4	0.29	0.65
5	0.35	1.26
6	0.29	0.45
7	0.55	7.07
8	0.35	1.55
9	0.26	0.40
10	0.35	1.65

Comparando as Tabelas 1 e 2, observa-se que o algoritmo proposto apresentou tempos computacionais elevados, se comparado com os tempos obtidos por Yanasse & Morabito (2003) mesmo levando em consideração que as máquinas são diferentes.

Refinamentos podem ser introduzidos no algoritmo visando melhorar seu tempo computacional. No algoritmo proposto, é construída uma matriz de dimensão $m \times LW$ para enumerar as possíveis combinações de itens que devem ser recuperadas a partir da própria tabela. O algoritmo, portanto, apresenta um requisito de memória na ordem de $O(LWm)$. Em problemas práticos, onde as dimensões dos objetos pode facilmente ultrapassar centenas de unidades e onde existem dezenas de tipos de itens de tamanhos relativamente pequenos a serem cortados, o problema de se manipular matrizes grandes pode tornar-se crítico. Uma solução encontrada para resolver este problema é realizar uma mudança de escala nos dados de entrada do problema. Neste caso, as áreas dos itens e objeto são divididas por um fator de escala, calculado pela menor área dentre os itens considerados. Na Tabela 3, apresenta-se os resultados obtidos com a utilização deste refinamento. Observa-se que a mudança de escala resultou numa melhoria significativa do tempo computacional do algoritmo proposto. Entretanto, devido ao truncamento resultante da divisão dos dados de entrada pelo fator de escala, podem ocorrer perdas de combinações no problema original. Para os exemplos testados, no entanto, não houve perda de qualidade da solução.

Tabela 3. Tempo computacional do algoritmo proposto (versão refinada, com mudança de escala nas dimensões dos itens e objetos) para os exemplos extraídos de Yanasse & Morabito (2003)

Exemplo	Fator de Escala	Tempo (em segundos)
1	195	5.00
2	299	2.04
3	280	1.45
4	384	0.93
5	221	0.63
6	697	0.38
7	390	0.22
8	304	0.42
9	504	0.16
10	392	0.16

5. Considerações Finais

Neste trabalho foi proposto um algoritmo para determinação de padrões tabuleiros exatos, onde o número de itens no padrão é limitado. O algoritmo utiliza um esquema enumerativo para gerar as possíveis combinações de itens restritos e, em seguida, determina o melhor padrão tabuleiro exato verificando-se a possibilidade de produzir padrões tabuleiros com as combinações obtidas. Para verificar isto, foi proposto um método para geração de um padrão tabuleiro exato dado uma combinação de itens que o padrão deve conter. O método gera um padrão tabuleiro exato, caso ele exista, ou indica que nenhum padrão exato contendo aqueles itens pode ser obtido. Testes computacionais realizados com algumas instâncias, baseadas em dados de problemas reais, mostraram que o método proposto fornece uma resposta em segundos, no máximo.

Apesar da determinação do padrão tabuleiro ser feita de maneira eficiente dada uma combinação, pode acontecer que o número de combinações viáveis possíveis a serem investigadas seja muito grande até que consiga assegurar a otimalidade do padrão gerado. Nestes casos, a introdução de refinamentos mostrou-se fundamental na melhoria do tempo computacional. Neste trabalho, foram utilizados fatores de escala. Testes realizados em exemplos extraídos da literatura mostraram que o uso do algoritmo proposto com a introdução deste refinamento, pode resultar num ganho significativo de tempo computacional, em alguns casos, sem perda de qualidade na solução. Outros refinamentos podem ser introduzidos, por exemplo, computar-se melhores limitantes inferiores para as perdas definidas durante a segunda fase do algoritmo. Isto está sendo investigado no momento.

5. Reconhecimento

Este trabalho foi parcialmente financiado pelo CNPq (Conselho Nacional de Pesquisa e Desenvolvimento

Científico e Tecnológico) e pela FAPESP (Fundação de Amparo a Pesquisa do Estado de São Paulo).

6. Referências

[1] Arenales, M. N. & Morabito, R. Modelos e métodos básicos. In: Arenales M. N. & Morabito, R. ed. Mini-curso: o problema de corte e empacotamento e aplicações industriais. XX Congresso Nacional de Matemática Aplicada e Computacional e II Oficina Nacional de Problemas de Corte e Empacotamento, Gramado (RS): ICMC-USP, 1997. Cap. 1, p.1-23.

[2] Belluzzo, L. Otimização nos planos de corte de chapas de fibra de madeira reconstituída: Um estudo de caso. Dissertação de Mestrado em Engenharia de Produção. Universidade Federal de São Carlos – São Carlos (SP), 2002.

[3] Dyckhoff, H. A typology of cutting and packing problems. *European Journal of Operations Research*, v.44, n.2, p.145-159, 1990.

[4] Dyckhoff, H. & Waescher, G. Special issue: Cutting and packing. *European Journal of Operations Research*, v.44, n.2, 1990.

[5] Gilmore, P. & Gomory, R. A linear programming approach to the cutting-stock problem. *Operations Research*, v.9, n.6, p.849-859, 1961.

[6] Gilmore, P. & Gomory, R. A linear programming approach to the cutting-stock problem II. *Operations Research*, v.11, n.6, p.863-888, 1963.

[7] Gilmore, P. & Gomory, R. Multistage cutting stock problems of two and more dimensions. *Operations Research*, v.13, n.1, p.94-120, 1965.

[8] Hinxman, A. The trim-loss and assortment problems: a survey. *European Journal of Operations Research*, v.5, n.1, p.8-18, 1980.

[9] Katsurayama, D. M. & Yanasse, H. H. Um algoritmo enumerativo para determinação de padrões tabuleiros. [CDROM]. In: Simpósio Brasileiro de Pesquisa Operacional, 31., Juiz de Fora (MG), 1999. Anais. Tec Art Editora, 1999. Seção de Corte e Empacotamento.

[10] Katsurayama, D. M. & Yanasse, H. H. Um algoritmo enumerativo para determinação de padrões tabuleiros: aspectos computacionais de implementação. [CDROM]. In: Simpósio Brasileiro de Pesquisa Operacional, 32., Viçosa (MG), 2000. Anais. Tec Art Editora, 2000. Seção de Corte e Empacotamento.

[11] Morabito, R. & Arenales, M. N. Optimizing the cutting of stock plates in a furniture company. *International Journal of Production Research*, v.38, n.12, p.2725-2742, 2000.

[12] Scheithauer, G. On a two-dimensional guillotine cutting problem. IFORS 2002. Edinburgh, Scotland, U. K., 2002. Cutting and Packing Workshop.

[13] Sweeney, P. & Patternoster, E. Cutting and packing problems: a categorised, application-oriented research

bibliography. *Journal of the Operational Research Society*, v.43, n.7, p.691-706, 1992.

[14] Yanasse, H. H.; Soma, N. Y. & Maculan N. An algorithm for determining the k-best solutions of the one-dimensional knapsack problem. *Pesquisa Operacional*, v.20, n.1, p.117-134, 2000.

[15] Yanasse, H. H. & Morabito, R. Linear Models for two-stage constrained two-dimensional guillotine cutting problems. Working paper, INPE/UFSCAR (submetido para publicação), 2003.