

**Análise de produtividade  
de  
padrões de corte  
na  
indústria de móveis**

Altamir Figueiredo

Dissertação de Mestrado  
Pós-Graduação em Matemática Aplicada

# **Livros Grátis**

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Figueiredo, Altamir Gomes.

Análise de Produtividade de Padrões de Corte na Indústria de Móveis /  
Altamir Gomes de Figueiredo – São José do Rio Preto : [s.n.], 2005.  
113 f. ; 30 cm.

Orientadora: Maria do Socorro Nogueira Rangel  
Dissertação (mestrado) – Universidade Estadual Paulista. Instituto de Bioci-  
ências, Letras e Ciências Exatas

1. Otimização Matemática. 2. Pesquisa Operacional. 3. Programação  
Inteira. 4. Problema do Corte e Estoque. 5. Indústria de Móveis – Corte de  
painéis. I. Rangel, Maria do Socorro Nogueira. II. Universidade Estadual  
Paulista. Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 519.8

# **Análise de produtividade de padrões de corte na indústria de móveis**

**Altamir Gomes Figueiredo**

Orientadora: Prof<sup>a</sup> Dr<sup>a</sup> Maria do Socorro N. Rangel

Dissertação apresentada ao Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Campus de São José do Rio Preto, São Paulo, para a obtenção do título de Mestre em Matemática Aplicada

São José do Rio Preto

05 de maio de 2006



## **Banca Examinadora**

### **Titulares**

Prof<sup>a</sup>. Dr<sup>a</sup>. MARIA DO SOCORRO N. RANGEL (Orientadora)  
Dep. C. Comp. e Estatística / Inst. de Biociências, Letras e C. Exatas – DCCE / IBILCE

Prof. Dr. HORÁCIO HIDEKI YANASSE  
Centro de Tec. Espacial / Instituto Nacional de Pesquisas Espaciais – CTE / INPE

Prof. Dr. SILVIO ALEXANDRE DE ARAUJO  
Dep. C. Comp. e Estatística / Inst. de Biociências, Letras e C. Exatas – DCCE / IBILCE

### **Suplentes**

Prof. Dr. MARCOS NEREU ARENALES  
Inst. de C. Matemáticas e de Computação / Univ. São Paulo – ICMC / USP

Prof. Dr. GERALDO NUNES SILVA  
Dep. C. Comp. e Estatística / Inst. de Biociências, Letras e C. Exatas – DCCE / IBILCE



*Não há sonhos, onde não há vida  
Não há vida, onde não há esperança  
Não há esperança, onde não há sonhos*

*A*

*Oswaldo e Otília, pela vida  
Antônio e Tereza, pelo direito de sonhar*

*Dedico*



*Aos nossos filhos*

*Perdoem a cara amarrada  
Perdoem a falta de abraço  
Perdoem a falta de espaço  
Os dias eram assim...  
Perdoem por tantos perigos  
Perdoem a falta de abrigo  
Perdoem a falta de amigos  
Os dias eram assim...  
Perdoem a falta de folhas  
Perdoem a falta de ar  
Perdoem a falta de escolha  
Os dias eram assim...  
E quando passarem a limpo  
E quando cortarem os laços  
E quando soltarem os cintos  
Façam a festa por mim!  
E quando lavarem a mágoa  
E quando lavarem a alma  
E quando lavarem a água  
Lavem os olhos por mim!  
E quando brotarem as flores  
E quando crescerem as matas  
E quando colherem os frutos  
Digam o gosto prá mim!  
Ah, digam o gosto prá mim...*

*Ivan Lins e Vitor Martins*

*A*

*Edilamar, esposa e companheira  
Tomaz e Pedro Victor, meus filhos amados  
Dedico*



## Agradecimentos

*"E como máquina, o combustível a rotina queimava"*

*Eduardo Lima*

À prof<sup>a</sup> Maria do Socorro Rangel, pela orientação e ensinamentos em meu processo de pesquisa, e no esforço de dar forma a este trabalho.

Ao amigo Rildo, o grande companheiro de estudos. Mostrou-me que, havendo respeito, duas pessoas tão diferentes podem construir uma forte relação de amizade.

Aos demais amigos do curso de mestrado, em especial à turma da “Casa Rosada” e “agregados” (Amabile, Adriana, Serginho, Eduardo, Cristiane, Fábio, Danilo, Reginaldo, Nilton, Orestes, Maricato, Fernando, Gabriela, Flávio, Pedrão, Chela, Dani-ele, Daniel e José Antônio) por evitarem que minha rotina queimasse o combustível.

Aos amigos Áurea e Seishin, pelo apoio nos momentos difíceis dessa trajetória.

Ao CNPq pelo apoio financeiro e à UNESP que, através dos professores do programa e dos funcionários do departamento, me possibilitaram realizar este projeto.

Aos loucos mais adoráveis (Francisco, Helena, Jorge, Leonardo e Maria) que não medem esforços para me apoiar.

À minha família, que suportou os dois anos de afastamento e os sacrifícios impostos, sem me negar o seu apoio.



## Resumo

Neste trabalho, analisamos os padrões de corte adotados por uma Indústria de Móveis, e identificamos suas características básicas. Definimos, a partir dessas características, os padrões tabuleiros compostos, que pertencem a classe dos padrões de corte  $n$ -grupos, apresentada por Gilmore e Gomory (1965). Os padrões tabuleiros compostos preservam as facilidades de corte dos padrões tabuleiros, apresentando melhores índices de sobra de matéria-prima. Propomos uma heurística para a geração de um *pool* de padrões tabuleiros compostos, usados para resolver o problema de corte de estoque na indústria de móveis.

**Palavras-chave:** Programação Inteira, Padrão Tabuleiro, Padrão Tabuleiro composto,  $n$ -grupos, Corte de estoque, Padrões de Corte Bidimensionais, indústria de móveis



## Abstract

In this work, we analyze the cutting patterns used by a furniture Industry, and we determine some of its basic characteristics. We defined a composed checkerboard pattern, that belongs to the class of  $n$ -groups cutting patterns, presented by Gilmore and Gomory (1965). The composed checkerboard patterns preserve the easiness of the cutting process and have better indexes of waste. We propose a heuristic to generate a pool of composed checkerboard patterns to solve the cutting stock problem in the furniture Industry.

**Keyword:** Integer Programming, Checkerboard Patterns, Composed Checkerboard Patterns,  $n$ -groups, Cutting Stock, Bidimensional Cutting Patterns, Furniture Industry



Introdução .....	1
1. Problemas do Corte e Empacotamento.....	5
1.1. Estrutura dos Problemas do Corte e Empacotamento .....	7
1.2. Tipologia de Dyckhoff .....	8
1.3. Tipologia proposta por Wäscher, Haußner e Schumann .....	10
1.4. Conclusão .....	15
2. O Problema do Corte de Estoque Bidimensional (PCE <sub>2</sub> ).....	17
2.1. Padrões de Corte Bidimensionais.....	18
2.2. Geração de padrões de corte bidimensionais .....	21
2.3. Modelo de Gilmore e Gomory para PCE <sub>2</sub> .....	27
2.4. Heurísticas para solução do PCE <sub>2</sub> usando padrões tabuleiros compostos.....	28
3. Problema do corte na indústria de móveis.....	33
3.1. Classificação dos móveis.....	34
3.2. A indústria moveleira no Brasil.....	35
3.3. O pólo moveleiro de Votuporanga .....	36
3.4. Uma indústria característica .....	36
3.5. Confeção de um móvel .....	37
3.6. Processo de corte do objeto .....	38
3.7. Análise dos padrões da indústria .....	45
3.8. Composição do preço do produto.....	46
3.9. Conclusão .....	48
4. Estudos computacionais .....	51
4.1. Estudo computacional da geração de padrões tabuleiros .....	52
4.1.1. Análise da diversidade nas dimensões dos itens .....	54
4.1.2. Análise do impacto de itens de dimensões pequenas .....	57
4.1.3. Conclusão .....	64
4.2. Estudo computacional das heurísticas propostas.....	64
4.2.1. Fase 1: Heurística 1 x Heurística 2 .....	66
4.2.2. Fase 2: Heurística 2 x CorteBi x Indústria .....	68
4.2.3. Fase 3: Programação de lotes duplos.....	72
4.2.4. Conclusão .....	73
Considerações Finais .....	77
Apêndice	
A.1. Conjuntos de dados para os testes da seção 4.2.....	81
A.2. A geração de padrões tabuleiros na sintaxe do Xpress-Mosel .....	82
A.3. A Heurística 2 na sintaxe do Xpress-Mosel .....	91
Referências Bibliográficas.....	107



O problema do corte de estoque está presente em diversos segmentos da indústria onde existe necessidade de se cortar peças maiores em peças menores, seja no corte de bobinas (de papel, tecido, aço) ou de painéis (de aço, madeira, espuma, vidro), para atender a uma demanda por peças menores pre-especificada. Quando no processo de corte duas dimensões forem relevantes, teremos um problema de corte bidimensional. Em muitas ocasiões a decisão de como cortar pode envolver uma estratégia de mercado, seja ela de redução de custo ou de rapidez no atendimento à demanda. Nesta dissertação nos concentraremos no problema do corte de estoque bidimensional, mais precisamente no problema do corte de estoque aplicado à indústria moveleira.

A indústria moveleira vive hoje, como os demais segmentos da indústria, a necessidade de se tornar mais competitiva comercialmente. Para isso ela precisa inovar seus processos de produção, de forma a produzir mais com um menor custo. Para a indústria moveleira, um dos principais quesitos, quando se aborda a questão do custo, é o corte da matéria-prima, ou seja, é como cortar melhor a matéria-prima.

Para definirmos se um corte da matéria-prima, na confecção de um móvel, é bom ou ruim, necessitamos analisar os padrões de corte envolvidos buscando observar, principalmente, dois aspectos: o tempo de corte e a sobra de matéria-prima. Assim, faremos um estudo dos problemas do corte de estoque bidimensional, classificando os padrões de corte quanto ao tipo de corte e o número de estágios. Abordaremos uma classe especial de padrões de corte, os padrões de corte  $n$ -grupos (Gilmore e Gomory, 1965), em especial o padrão tabuleiro (Yanasse e Katsurayama, 2001 e 2004) e o padrão tabuleiro composto (Figueiredo e Rangel, 2005). Discutiremos alguns modelos de otimização para a geração de padrões de corte e para o problema do corte de estoque bidimensional.

Discutiremos, a partir dos padrões de corte adotados por uma Indústria característica do pólo moveleiro de Votuporanga/SP, algumas das características que julgamos serem necessárias aos padrões de corte para que atendam às necessidades da Indústria. Veremos que os padrões de corte adotados pela Indústria tendem a ser uma composição de padrões tabuleiros que foram denominados padrões tabuleiros compostos (Figueiredo e Rangel, 2005). A partir desta discussão, proporemos uma heurística para a criação de um *pool* de padrões tabuleiros compostos, que servirão de base para solução do problema do corte de estoque bidimensional. A idéia, com a criação deste *pool*, é encontrar uma solução para o problema usando padrões que atendam as principais características dos padrões adotados pela Indústria.

Consideraremos, no decorrer desta dissertação, que o leitor esteja familiarizado com os conceitos de programação linear (Bazaraa *et al*, 1990) e programação inteira, *branch-and-bound*, planos de corte, geração de colunas, *branch-and-cut*, *branch-and-price* e *branch-and-cut-and-price* (Nemhauser e Wolsey, 1988; Barnhart *et al*, 1996; Wolsey, 1998; Vance, 1998; Cacceta e Hill, 2001; Belov e Sheithauer, 2006).

Estruturamos esta dissertação em 5 partes: na primeira, discutimos a estrutura e classificação dos problemas de corte e empacotamento; na segunda, a discussão é cen-

trada na formulação dos problemas do corte de estoque bidimensional e na geração de planos de corte para estes; na terceira, apresentamos um panorama geral do setor moveleiro no Brasil, e discutimos os padrões de corte adotados pela Indústria; na quarta, apresentamos e discutimos os resultados dos testes realizados; por fim, apresentamos as considerações feitas ao longo do trabalho e identificamos alguns pontos que, mesmo não tendo sido cobertos neste trabalho, consideramos relevantes e merecedores de uma maior investigação. Anexamos, como Apêndice, os códigos utilizados nos testes, desenvolvidos no Xpress-Mosel



## *Capítulo 1*

# **Problemas do Corte e Empacotamento**

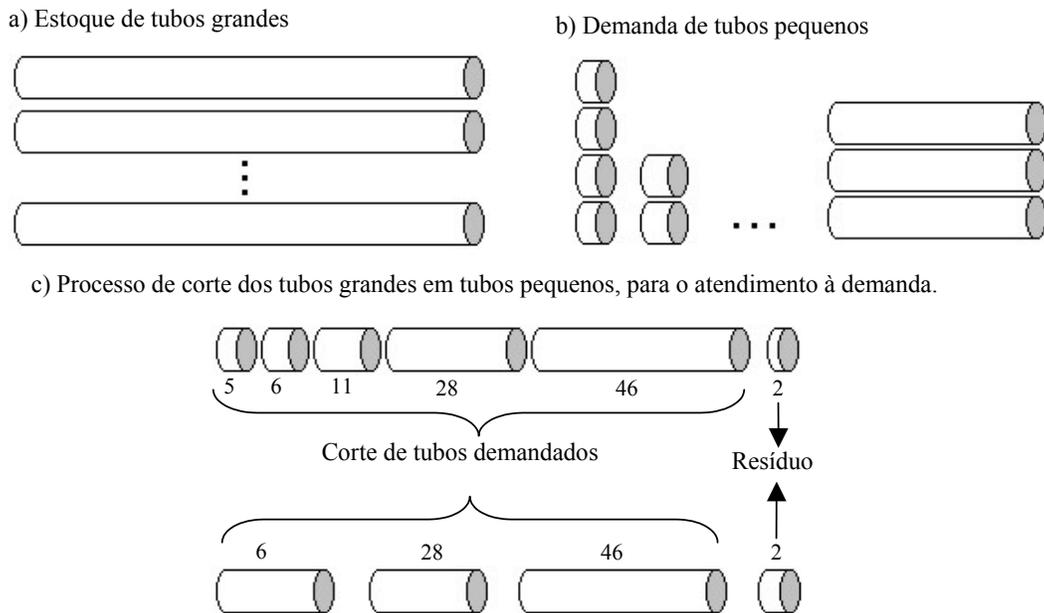
---

Inúmeras indústrias possuem seus processos de produção atrelados ao corte de unidades maiores (objetos) em unidades menores (itens). Inúmeras outras possuem seus processos de produção atrelados ao empacotamento de unidades menores (itens) dentro de unidades maiores (objetos). Cortar objetos em itens ou empacotar itens em objetos consistem, matematicamente falando, em problemas idênticos, pois apresentam essencialmente a mesma estrutura lógica, podendo ser abordados com as mesmas formulações e estratégias de solução. Por isso, problemas desta classe são referenciados na literatura como Problemas de Corte e Empacotamento.

A estrutura lógica básica dos Problemas de Corte e Empacotamento pode ser facilmente percebida se olharmos os problemas 1, 2 e 3 a seguir.

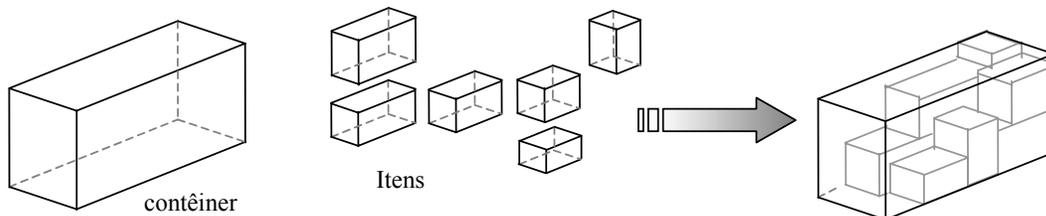
- Problema 1: O corte de tubos para equipamentos de calefação (Figura 1.01). Considere que exista um estoque ilimitado de tubos grandes, de tamanho 98cm, usados para a produção de tubos menores, e uma lista de tubos menores, com tamanhos entre 5 e 46cm, que devem ser produzidos para atender a uma demanda semanal (Heicken e König, 1980, *apud* Dyckhoff, 1990).

Figura 1.01: Corte de tubos em estoque para atendimento à demanda



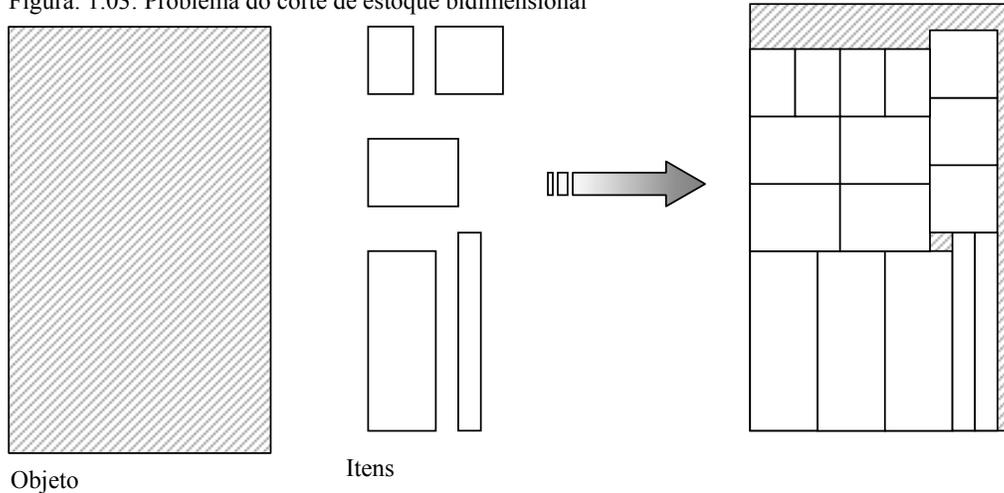
- Problema 2: O carregamento de contêineres (Figura 1.02). Este tipo de problema também apresenta dois grupos de dados: um estoque de objetos, consistindo de um ou mais contêineres e uma lista de itens que deverão ser alocados dentro do(s) contêiner(es) (Gehring *et al*, 1990; Haessler e Talbot, 1990; *apud* Dyckhoff, 1990).

Figura 1.02: Problema do carregamento de contêiner



- Problema 3: O corte de placas de madeira para a confecção de moveis (Figura 1.03). Consiste em encontrar a melhor forma de cortar um objeto retangular, para a produção de itens, também retangulares (Dyckhoff, 1990).

Figura: 1.03: Problema do corte de estoque bidimensional



### 1.1. Estrutura dos Problemas de Corte e Empacotamento

Nos anos 60, Gilmore e Gomory (1962, 1963, 1965) publicaram uma seqüência de trabalhos de modelagens e métodos de solução para os problemas de Corte e Empacotamento. Nos anos que se seguiram, diversos autores realizaram as mais variadas pesquisas e publicaram uma grande gama de problemas com a mesma estrutura lógica, definida da seguinte forma(Wäscher et al, 2004):

- a) dados dois conjuntos de elementos, definidos em 1, 2, 3 ou n dimensões:
  - Um de peças grandes (objetos), normalmente compondo a entrada ou o estoque do problema;
  - Um de peças pequenas (itens), normalmente compondo a saída ou a demanda a ser atendida;
- b) uma seleção de itens agrupados em um ou mais subconjuntos, com cada subconjunto associado a um objeto, de modo que sejam asseguradas as seguintes condições geométricas:
  - Todos os itens de um subconjunto estejam totalmente contidos em um objeto;
  - Os itens não se sobrepõem;
- c) e, uma função objetivo.

A solução final de um problema do Corte e Empacotamento pode resultar na utilização de alguns ou todos os objetos e de alguns ou todos os itens. Nela podemos distinguir cinco subproblemas que devem ser resolvidos, simultaneamente, de forma a conseguir um “ótimo global”:

- O problema da seleção de itens;
- O problema da seleção de objetos;
- O problema do agrupamento dos itens selecionados;
- O problema da designação dos grupos de itens para os objetos;
- O problema do arranjo dos itens em cada um dos objetos, respeitando as restrições geométricas mencionadas.

Tipos especiais de Problemas de Corte e Empacotamento são caracterizados por propriedades adicionais. Em particular eles podem ser degenerados, no sentido de não possuírem todos os subproblemas mencionados.

## 1.2. Tipologia de Dyckhoff

Em 1990, Dyckhoff propôs uma classificação tomando por base a estrutura lógica dos Problemas de Corte e Empacotamento e suas principais características: dimensionalidade, mensuração das quantidades, forma e sortimento dos objetos e itens, disponibilidade, restrições dos padrões, restrições de alocação e sua viabilidade. Estas características foram agrupadas em quatro critérios básicos, usados para classificar os problemas e descritos a seguir:

1. Dimensionalidade
  - (1) Unidimensional;
  - (2) Bidimensional;
  - (3) Tridimensional;
  - ( $n$ )  $n$ -dimensional.
2. Tipo de alocação
  - (B) Todos os objetos e uma parte dos itens;
  - (V) Uma parte dos Objetos e Todos os itens.
3. Sortimento dos Objetos
  - (O) Um objeto;
  - (I) Objetos idênticos;
  - (D) Objetos diferentes.
4. Sortimento dos itens
  - (F) Poucos itens (diferentes);
  - (M) Muitos itens de muitos tamanhos diferentes;
  - (R) Muitos itens com relativamente poucos tamanhos diferentes;
  - (C) Tamanhos congruentes.

Com esses critérios ele classificou, de forma consistente e sistemática, os diversos tipos de Problemas de Corte e Empacotamento, agrupando-os em classes, definidas através de uma quádrupla ( $\alpha / \beta / \gamma / \delta$ ), onde  $\alpha$ ,  $\beta$ ,  $\gamma$  e  $\delta$  correspondem, respectivamente, aos critérios 1, 2, 3 e 4 definidos anteriormente.

### **Limitações da Tipologia de Dyckhoff**

Incontestavelmente o trabalho de Dyckhoff foi um marco na pesquisa dos Problemas de Corte e Empacotamento, ele destacou a estrutura básica comum entre os problemas de corte e os problemas de empacotamento e, com isso, promoveu a integração dessas duas áreas de pesquisa, até então vistas separadamente. No entanto, sua tipologia não obteve uma unanimidade internacional. Algumas desvantagens e controvérsias foram apontadas por diversos pesquisadores (Wäscher *et al*, 2004), tais como:

- A classificação pode não ser necessariamente única.  
O problema do carregamento de veículos foi codificada por Dyckhoff (1990) como 1/V/I/F e 1/V/I/M. onde “F” caracteriza a situação em que poucos itens de diferentes formas são designados. “M” caracteriza a situação em que muitos itens de muitas diferentes formas são designados.
- A tipologia é parcialmente inconsistente, podendo apresentar resultados confusos.  
Por exemplo, o problema do empacotamento por faixas (*strip-packing problem*), um caso particular do problema de empacotamento bi-dimensional, onde um conjunto de itens de diferentes tamanhos é empacotado em retângulos individuais de largura fixa e comprimento mínimo (Jacob, 1996; Martello *et al*, 2003; *apud* Wäscher *et al*, 2004), foi codificado (classificado) por pesquisadores e profissionais da área como 2/V/O/M. No entanto, na tipologia de Dyckhoff ele recebe a notação 2/V/D/M. Dyckhoff (1990) justifica sua notação dizendo que este problema “... é equivalente ao problema da seleção de objetos, onde o estoque é dado por uma infinidade de objetos de mesma largura e com todas as possibilidades de comprimento e somente um objeto será escolhido do estoque, i.é, o de menor comprimento”. Dyckhoff (1990) denomina, ainda, o problema como um problema 2-D *Bin Packing*. Denominação esta que, no entender de diversos pesquisadores (Lodi *et al*, 2002; Miyazawa e Wakabayashi, 2003; Faroe *et al*, 2003; *apud* Wäscher *et al*, 2004), deveria ser reservada para a extensão natural do problema *Bin Packing* clássico (unidimensional).

- A aplicação da tipologia de Dyckhoff não resulta, necessariamente, em categorias de problemas homogêneos.

No caso do corte unidimensional, onde um grande número de itens com poucas formas diferentes são produzidos de uma quantidade ilimitada de objetos, podemos incorrer em duas situações diferentes se os objetos possuírem diferentes tamanhos: na primeira, os objetos são separados em pequenos grupos de elementos idênticos; na segunda, todos os objetos são diferentes. Ambas as situações são classificadas por Dyckhoff como 1/V/D/R. Gradisar *et al* (2002) consideraram que utilizar a mesma classificação para as duas situações não é adequado, pois elas requerem soluções diferentes (Wäscher *et al*, 2004).

### 1.3. Tipologia proposta por Wäscher, Haußner e Schumann

Em geral os Problemas de Corte e Empacotamento podem ser definidos em 2 tipos, elementares ou combinados, Wäscher *et al* (2004) apresentaram o esboço de uma nova tipologia. Nela eles utilizam esses dois tipos para o desenvolvimento de modelos, algoritmos, geradores de problemas e para a categorização literária.

As definições dos tipos dos problemas são baseadas em 5 critérios, dimensionalidade, tipo de designação, sortimento de itens, sortimento de objetos e forma dos itens, descritos a seguir:

- Dimensionalidade

Tal como na tipologia de Dyckhoff, são consideradas 1, 2 ou 3 dimensões, eventualmente são consideradas dimensões superiores (Lins e Morábito, 2002; *apud* Wäscher *et al*, 2004).

- Tipo de designação

Neste critério, que toma por base o critério análogo de Dyckhoff, os autores se referem aos problemas como:

- Minimização do consumo (*input minimization*)

Um dado conjunto de itens deve ser designado para um conjunto de objetos. Aqui o conjunto de objetos é suficiente para acomodar todos os itens, ou seja, todos os itens serão designados para uma seleção de objetos de “valor mínimo”, não existindo o problema de seleção com respeito aos itens

- Maximização da produção (*output maximization*)

Um dado conjunto de itens deve ser designado para um conjunto de objetos. O conjunto de objetos não é suficiente para acomodar todos os itens. Com isso, todos os objetos serão usados, para que a produção de uma seleção de itens seja maximizada.

- Sortimento de itens

Com respeito ao sortimento de itens, há uma distinção em 3 casos:

- Itens idênticos

Todos os itens são da mesma forma e tamanho. Esta categoria de problemas é idêntica a  $\delta$ ="C" na classificação de Dyckhoff.

- Sortimento fracamente heterogêneo

Os itens podem ser agrupados em classes de itens idênticos. Por definição, itens de mesma forma e tamanho que possuem orientações diferentes são tratados como itens de tipos diferentes. A demanda de cada item é relativamente grande, podendo ser limitada. Esta categoria corresponde a  $\delta$ ="R" na classificação de Dyckhoff.

- Sortimento fortemente heterogêneo

O conjunto de itens é caracterizado pelo fato de que poucos itens são idênticos. Se isto ocorre os itens são tratados como elementos individuais e, conseqüentemente, suas demandas serão iguais a 1. Esta categoria corresponde a  $\delta$ ="M" na classificação de Dyckhoff.

- Sortimento de objetos

- Um objeto

Neste caso o conjunto de objetos consiste de apenas um elemento, que poderá ter todas as dimensões fixas, ou ter uma ou mais dimensões variáveis. O primeiro caso é idêntico a  $\gamma$ ="O" na classificação de Dyckhoff, enquanto que o segundo representa uma extensão do conjunto de tipos elementares de Dyckhoff.

- Vários objetos

Neste caso não é necessário distinguir entre dimensões fixas e variáveis, com respeito aos problemas encontrados na literatura. Somente serão considerados objetos de dimensões fixas. De forma análoga às categorias introduzidas para o

sortimento de itens, os objetos serão distinguidos entre idênticos, fracamente e fortemente heterogêneos. Com isso, é feita uma extensão da tipologia de Dyckhoff, que somente identifica os objetos em formas idênticas e formas diferentes ( $\gamma$ ="I" e  $\gamma$ ="D", respectivamente).

- Forma dos itens

No caso de problemas bi e tridimensionais, para a definição de problemas do tipo refinado precisaremos distinguir também entre itens regulares e itens irregulares.

### **Tipos de problemas**

Wäscher *et al* (2004) classificam os Problemas do Corte e Empacotamento em três tipos: Básico, Intermediário e Refinado, conforme descrito a seguir.

#### **Tipo Básico**

Tipos básicos de Problemas do Corte e Empacotamento são obtidos pela combinação de dois critérios: "Tipo de designação" e "Sortimento de itens".

- Maximização da produção ( "*output maximization types*")

Problemas deste tipo têm em comum o fato de que o estoque de objetos é limitado, não sendo possível produzir todos os itens. Como queremos maximizar a produção dos itens, segundo uma dada função objetivo, todos os objetos serão utilizados. Logo, em geral, há um problema de seleção relativo aos itens a serem produzidos, não em relação aos objetos. Os seguintes problemas são considerados do tipo básico:

- Problema do empacotamento de itens idênticos (*IIPP – Identical Item Packing Problem*)

Esta categoria de problemas consiste na designação do maior número possível de itens idênticos para um dado conjunto (limitado) de objetos. Como os itens são idênticos, não há um problema de seleção, agrupamento ou de distribuição de itens. O problema é reduzido a um arranjo dos itens (idênticos) em cada um dos objetos.

- Problema da alocação (*PP – Placement Problem*)

Esta é uma categoria na qual um sortimento fracamente heterogêneo de itens precisa ser designado a um dado conjunto (limitado) de objetos. A produção dos

itens tem que ser maximizada ou, alternativamente, o desperdício correspondente minimizado.

- Problema da mochila (*KP – Knapsack Problem*)

Esta é uma categoria com sortimento fortemente heterogêneo de itens, que precisa ser designado a um dado conjunto (limitado) de objetos. Busca-se maximizar os itens produzidos, conforme a função objetivo adotada.

- Minimização do consumo (“*input minimization types*”)

Estes problemas são caracterizados pelo fato da quantidade de objetos ser grande o bastante para acomodar todos os itens. Neste caso, busca-se minimizar o valor dos objetos necessários para o atendimento à demanda, conforme a função objetivo adotada.

- Problema da dimensão aberta (*ODP – Open Dimension Problem*)

Neste caso, ao menos uma das dimensões dos objetos é variável.

- Problema do corte de estoque (*CSP – Cutting Stock Problem*)

Nesta categoria uma pequena diversidade de itens deve ser completamente alocada em uma seleção de objetos, a menor possível.

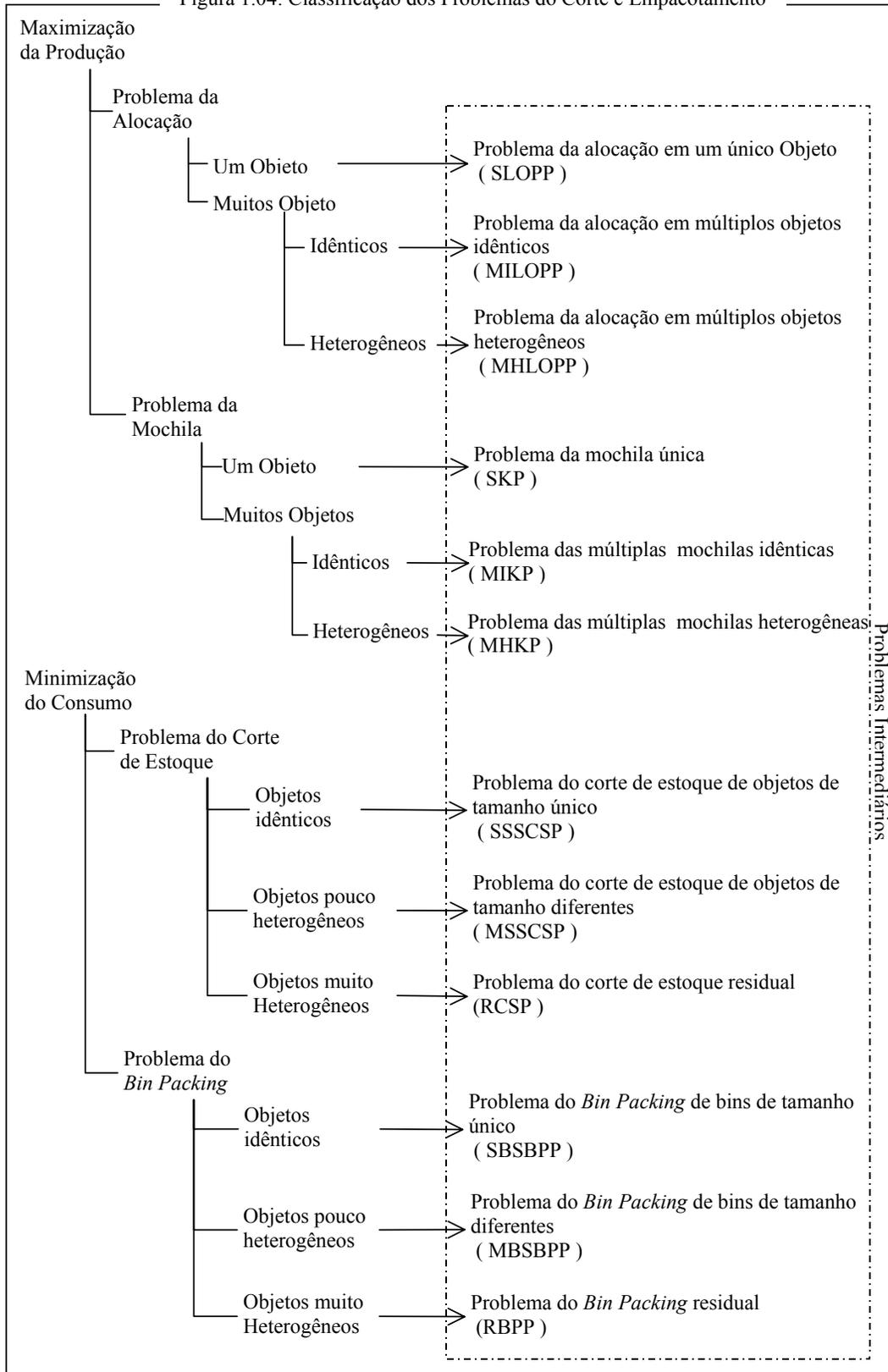
- Problema do Bin Packing (*BPP – Bin Packing Problem*)

Este tipo de problema possui uma grande diversidade de itens, que devem ser alocados em um conjunto de objetos, onde o valor (número ou tamanho total) dos objetos necessários tem que ser minimizado.

### **Tipo Intermediário**

A fim de definir tipos de problemas mais homogêneos, foi adicionado um novo critério aos tipos básicos: o sortimento de objetos. Com isso, os tipos básicos foram reestruturados gerando uma quantidade maior de tipos, denominados tipos intermediários. Conforme mostrado no esquema da figura 1.04, a seguir.

Figura 1.04: Classificação dos Problemas do Corte e Empacotamento



## Tipos refinados

Por último, Wäscher *et al* (2004) consideram a aplicação dos critérios de dimensionalidade e de forma dos itens, resultando em subcategorias dos problemas do tipo intermediário, que são identificadas pelos adjetivos adicionados aos seus nomes. Assim a estrutura do nome de um problema do tipo refinado será:

$$\{\text{dimensão}\} + \{\text{forma}\} + \{\text{IPT}\}$$

onde: dimensão → 1, 2 ou 3-dimensional;

forma →  $\emptyset$ , retangular, circular, cilíndrica, ..., irregular

IPT → Nome do problema do tipo intermediário.

### 1.4. Conclusão

Enquanto a tipologia de Dyckhoff (1990) nos apresenta um código de estrutura fixa para classificação dos problemas, tal qual ocorre em outras áreas (por exemplo, problemas de sequenciamento de tarefas), a tipologia proposta por Wäscher *et al* (2004) rompe bruscamente esta idéia e nos apresenta uma classificação sem uma estrutura comum em seus códigos, ou seja, seus códigos se apresentam mais como siglas para abreviação dos diversos problemas, do que códigos de classificação propriamente dito. Possuem uma considerável variação de tamanho (de 3 a 6 caracteres), o que dificulta a identificação de uma estrutura lógica, por alguém recém chegado a área. Por outro lado, é incontestável que a tipologia de Wäscher *et al* é mais abrangente e precisa que a de Dyckhoff. Não podemos fechar os olhos às diversas críticas apresentadas por Wäscher *et al* à tipologia de Dyckhoff.



### **Problemas do Corte de Estoque Bidimensional**

---

O problema do corte de estoque bidimensional surge quando, no processo de produção, necessitamos cortar peças do estoque (objetos) em peças menores (itens) levando em conta duas dimensões. Este problema é de grande aplicação em indústrias que utilizam como matéria prima placas de vidro, aço, papel, plástico, couro, tecidos, madeira, entre outros. Neste trabalho denotaremos o Problema do Corte de Estoque Bidimensional por  $PCE_2$  (Cintra, 2004). Na tipologia de Dyckhoff (1990) o  $PCE_2$  é classificado como 2/V/I/R, enquanto na tipologia proposta por Wäscher *et al* (2004) ele é classificado como 2-SSSCSP.

O objetivo do  $PCE_2$  a ser tratado neste trabalho é encontrar o número mínimo de objetos retangulares em estoque, de dimensões  $L \times W$ , necessários ao atendimento à demanda  $b_i$  de itens retangulares menores de dimensões  $\ell_i \times w_i$ , onde  $L$ ,  $W$ ,  $\ell_i$ ,  $w_i$  e  $b_i$  são inteiros com  $i=1, \dots, n$ . A solução do problema nos dá não só o número mínimo de objetos como também especifica como os objetos necessários para o atendimento à demanda dos itens serão cortados, isto é, nos dá também o padrão de corte.

## 2.1 Padrões de corte bidimensionais

Um PCE<sub>2</sub>, com grande aplicação prática, é o que considera o comprimento ( $L$ ) e a largura ( $W$ ) como dimensões relevantes. Nele, busca-se determinar a melhor forma de cortar objetos retangulares de tamanho padrão ( $L \times W$ ) em itens também retangulares ( $\ell_i \times w_i$ ), considerando suas demandas ( $b_i$ ).

Tais formas de corte, doravante denominados padrões de corte, determinam a disposição, sem sobreposição, dos itens em um objeto, determinando os cortes necessários no objeto para a obtenção dos itens.

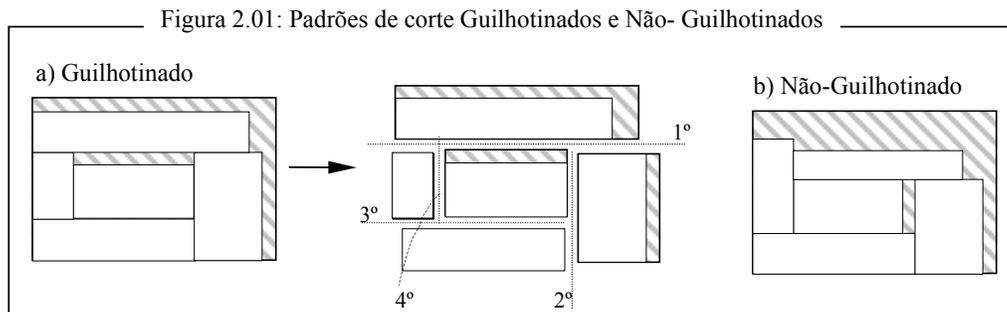
A um padrão de corte pode ser associado um vetor ( $A_j \in Z_+^n$ ), que registra a quantidade de cada item  $i$  obtido pela aplicação do padrão  $j$ .

$$A_j = (a_{1j} \ a_{2j} \ \dots \ a_{ij} \ \dots \ a_{nj})^T$$

Os retângulos, gerados após a aplicação de um padrão de corte, que não possuem representação em seu vetor associado são denominados sobras. Um padrão de corte apresenta diversas características que podem ser determinantes na avaliação de sua qualidade. Dentre elas destacamos: o tipo de corte, a rotação de itens, o índice de sobra de material e número de estágios.

### Tipo de corte: Guilhotinado x Não-Guilhotinado

Quando o corte é feito de uma extremidade a outra do objeto, dizemos que o corte é guilhotinado. Caso contrário diremos que o corte é não-guilhotinado. Se um padrão de corte só possuir cortes guilhotinados ele é denominado padrão de corte guilhotinado. Se isto não ocorrer ele será denominado padrão de corte não-guilhotinado (Figura 2.01). Neste trabalho consideraremos apenas os cortes guilhotinados paralelos a um dos lados do objeto, dividindo-o em dois retângulos menores.



### Rotação dos itens

Em certas indústrias, os objetos possuem uma característica que os orienta. Por exemplo: pranchas de madeira nativa, placas de compensado laminado, tecidos estam-

pados, etc. Nestes casos a rotação dos itens terá reflexo direto na qualidade do item cortado. Por outro lado, em outras indústrias os objetos a serem cortados são lisos, ou seja, sem veios ou estampas que os oriente. Neste caso os itens a serem cortados podem sofrer uma rotação ortogonal, sem que isso interfira na qualidade do item cortado.

Na primeira situação, quando um padrão de corte não admitir a rotação de itens, diremos que os itens possuem orientação fixa. Na segunda, quando admitir a rotação de itens, diremos que os itens têm rotação permitida. Neste segundo caso, podemos considerar cada item rotacionado como um novo item e, assim, tomar o problema como um problema de itens de orientação fixa. Neste caso, devemos estar atentos, na implementação, ao controle do atendimento à demanda dos itens, visto que um dado item e o novo item gerado por sua rotação responderão pela mesma demanda.

### **Perda: Desgaste da serra, Excedente de produção e Sobra de material**

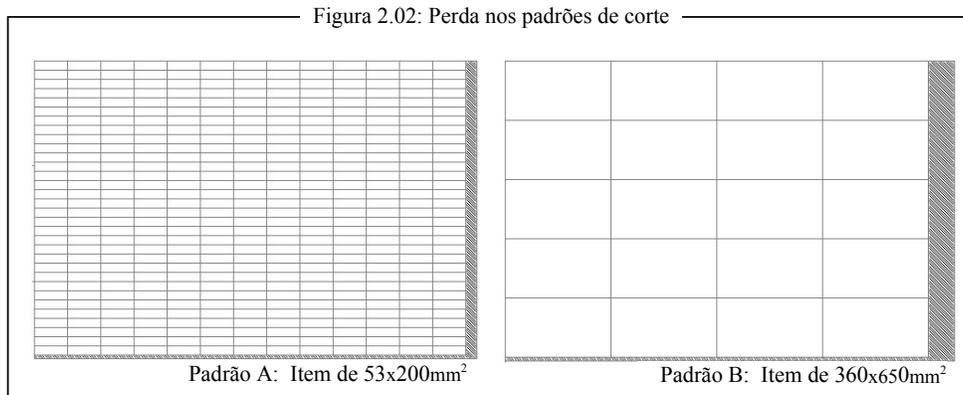
Um dos fatores primordiais na avaliação de um padrão de corte é, sem dúvida, o seu percentual de perda, que pode ser calculado pela fórmula:

$$P_j = L \cdot W - \sum_{i=1}^n (a_{ij} \cdot \ell_i \cdot w_i) \quad , \quad j = 1, 2, \dots, m \quad (2.01)$$

No entanto, precisamos estar atentos a forma como se dá esta perda. Uma parte dela é inerente ao processo de corte propriamente dito, o que nós chamamos de desgaste da serra. Uma outra parte, que compõe esta perda, são peças cortadas além da demanda, ou seja, por itens produzidos em excesso e por retalhos quaisquer do objeto, de dimensões não predefinidas, denominado sobra de material. A perda devido ao desgaste da serra, em certos tipos de matéria prima, é inevitável e precisa ser levada em conta quando queremos avaliar o custo de produção. Porém, quando da avaliação da eficiência do padrão de corte, esta perda pode gerar uma distorção, pois padrões compostos por muitos itens pequenos irão sempre apresentar um alto percentual de perda devido ao desgaste da serra, sem que isso implique que sejam perdulários.

Por exemplo, supondo que a espessura da serra seja de 4mm, as dimensões do objeto ( $L \times W$ ) de 2750x1830 mm<sup>2</sup> e que possuímos dois itens, um de 200x53 mm<sup>2</sup> e o outro de 650x360 mm<sup>2</sup>. Se cortarmos dois objetos, um para cada item (Figura 2.02), e efetuarmos os cálculos, de acordo com (2.01), veremos que a perda do padrão A é maior que a do padrão B ( $P_A=12,38\%$  e  $P_B=7,00\%$ ). Tomando por base estes valores somos levados a crer, inicialmente, que o padrão B é mais eficiente que o padrão A. No entanto, consi-

derando que cada item  $i$  tem atrelado a si dois cortes, um de tamanho  $\ell_i$  e outro  $w_i$ . Podemos afirmar que para cada um deles existe uma inevitável perda por desgaste da serra ( $Pd$ ).



No exemplo, é visível que a sobra de material no padrão B é maior que no padrão A. Assim, se incluirmos na fórmula do cálculo da perda o desgaste devido a espessura da serra (Morábito e Arenales, 2000), e refizermos os cálculos, o valor encontrado representará, exatamente a perda por sobra de material ( $Ps$ ):

$$Ps_j = (L + \delta) \cdot (W + \delta) - \sum_{i=1}^n (a_{ij} \cdot (\ell_i + \delta) \cdot (w_i + \delta)), \quad j = 1, 2, \dots, m \quad (2.02)$$

onde  $\delta$  é o desgaste provocado pela serra.

Perceba que neste caso teremos uma perda, por sobra de material, de 4,23% para o padrão A e de 5,74% para o padrão B. A perda relativa ao desgaste da serra ( $Pd_j$ ) poderá ser calculada pela diferença entre a perda total ( $P_j$ ) e a perda por sobra ( $Ps_j$ ):

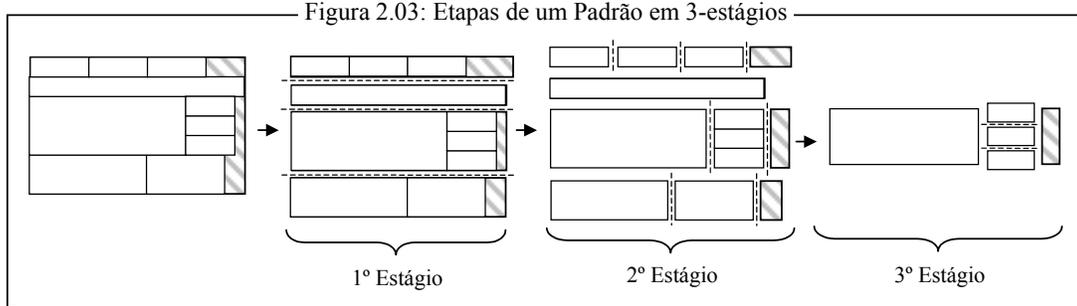
$$Pd_j = P_j - Ps_j .$$

Sem dúvida é importantíssimo levar em conta a perda por desgaste da serra no cálculo do custo de produção. Todavia considerá-la na avaliação da eficiência de um padrão de corte pode gerar distorções na análise. Na avaliação da eficiência de um conjunto de padrões de corte é necessário, ainda, levar em conta os excessos na produção dos itens. Estes excessos, em geral, acarretam um aumento do custo de estocagem.

### Número de estágios

O número de estágios de um padrão de corte mede a quantidade de mudanças na direção do corte. No primeiro estágio é realizada uma seqüência de cortes paralelos a um dos lados. A partir daí, a cada necessidade de mudança na direção do corte (rotação de 90°), para a realização do padrão de corte, incrementamos de uma unidade o número de estágios.

Assim, se forem necessários  $k-1$  mudanças de direção, teremos um padrão de corte em  $k$ -estágios. A figura 2.03 ilustra um padrão de corte onde são necessárias duas mudanças de direção no corte do padrão, caracterizando um padrão de corte em 3-estágios.



Os padrões em  $k$ -estágios podem, ainda, ser subdivididos em dois grupos: Exato, quando ao final do  $k$ -ésimo estágio os itens não necessitarem de um corte de ajuste (Figura 2.02); Não-Exato, quando ao menos um item necessitar de ajuste (Figura 2.01a).

## 2.2 Geração de padrões de corte bidimensionais

Uma importante subclasse dos padrões de corte guilhotinados é a dos padrões de corte em 2-estágios. Gilmore e Gomory, em 1964, apresentaram um método para elaboração de padrões de corte em 2-estágios que consiste em resolver a correspondente generalização do problema da mochila, conforme o modelo .

$$\begin{aligned} \text{Modelo 1: } \max \sum \pi \cdot y \\ \text{S.a : } y \in \{A_1, A_2, \dots, A_m\} \\ A_j \text{ é um padrão de corte em 2 - estágios, } \quad j = 1, 2, \dots, m. \end{aligned} \quad (2.03)$$

Tomando um objeto de dimensões  $L \times W$  e  $n$  itens de dimensões  $\ell_i \times w_i$ , o padrão de corte  $y$  pode ser obtido da seguinte forma (e.g. Cavali e Rangel, 2004):

1ª) Supondo que os itens estão ordenados, de forma que  $w_i \leq w_{i+1}$ ,  $i = 1, \dots, n$ . Para cada largura  $w_i$  calculamos  $\pi_i^*$ , com a alocação de itens  $\ell_k \times w_k$ , onde  $w_k \leq w_i$ , em uma faixa  $L \times w_i$ . Para cada  $i$ ,  $i \in [1, \dots, n]$  é resolvido o problema da mochila:

$$\begin{aligned} \pi_i^* = \max \sum_{k=1}^i \pi_k \cdot r_{ik} \\ \text{S.a : } \sum_{k=1}^i \ell_k \cdot r_{ik} \leq L \\ r_{ik} \in \mathbb{Z}_+ \quad , \quad k = 1, 2, \dots, i \quad , \quad i \in [1, \dots, n] \end{aligned} \quad (2.04)$$

onde  $r_{ik}$  é a quantidade de itens  $\ell_k \times w_k$  na faixa  $L \times w_i$ .

2ª) O valor da função objetivo no Modelo 1 (2.03) é obtido pela solução do problema:

$$\begin{aligned} \pi_j &= \max \sum_{i=1}^n \pi_i^* \cdot z_i \\ \text{S.a: } &\sum_{i=1}^n w_i \cdot z_i \leq W \\ &z_i \in Z_+ \quad , \quad i = 1, 2, \dots, n \end{aligned} \quad (2.05)$$

onde  $z_i$  é a quantidade da faixa  $L \times w_i$  no padrão  $j$ .

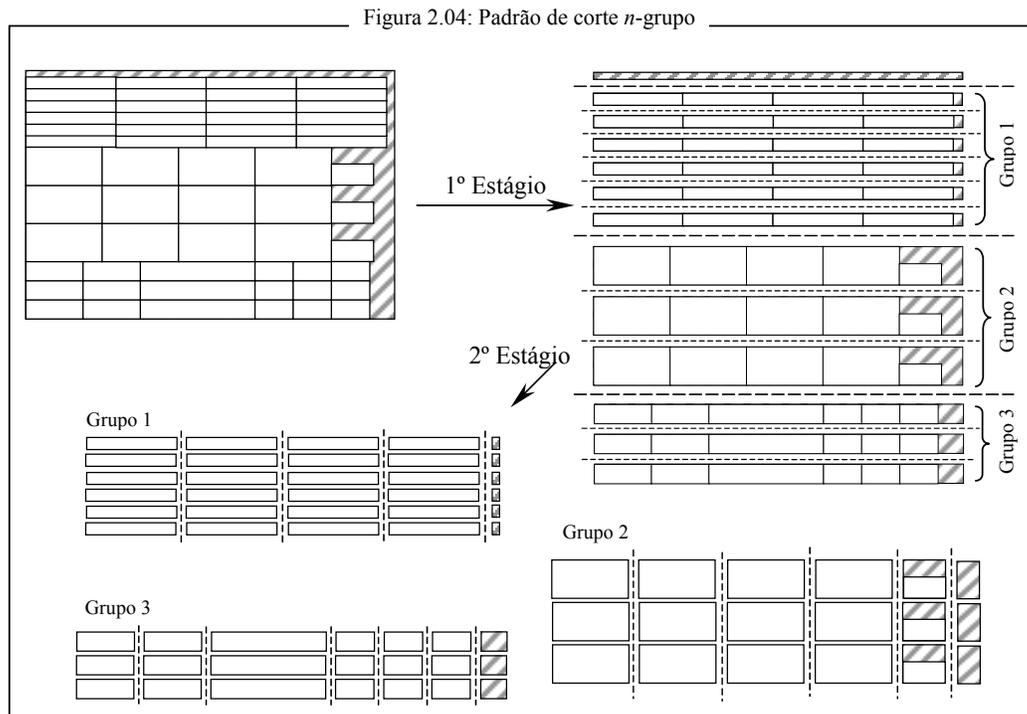
3ª) Finalmente, o vetor  $A_j$  é obtido por:

$$a_{ij} = \sum_{k=1}^n z_k \cdot r_{ik} \quad , \quad i = 1, \dots, n \quad (2.06)$$

A partir do Modelo 1, podemos obter um Modelo 2, para o caso exato, tomando  $w_k = w_i$  na primeira etapa.

### Padrão $n$ -grupo

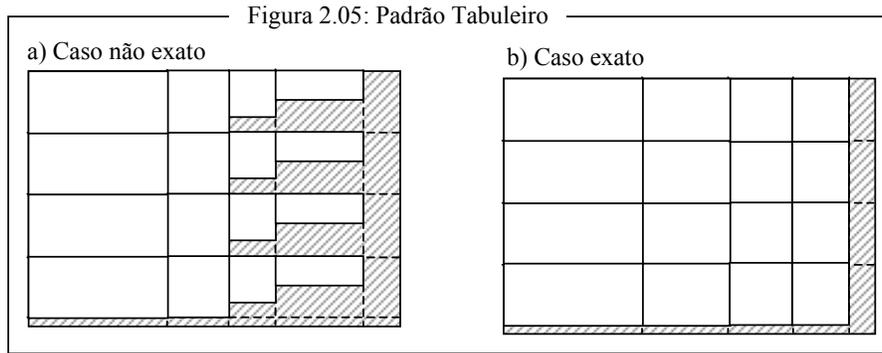
Gilmore e Gomory, em 1965, apresentam uma subclasse especial de padrões de corte guilhotinado em 2-estágios, no qual as faixas resultantes do primeiro estágio são divididas em  $n$  grupos, de modo que todas as faixas de um mesmo grupo são cortadas simultaneamente no segundo estágio (Figura 2.04). Esses padrões são denominados padrões  $n$ -grupo.



### Padrão Tabuleiro

O padrão tabuleiro, também referido na literatura como padrão 1-grupo, é uma classe particular dos padrões em 2-estágios que apresenta como característica o fato de que no segundo estágio as faixas, geradas no primeiro estágio, podem ser cortadas simultaneamente, gerando todos os itens do padrão de corte (Gilmore e Gomory, 1964).

Podemos obter um padrão tabuleiro tomando apenas um  $w_i$  em (2.05), conforme a Figura 2.05.



Um padrão de corte tabuleiro, que produz apenas um tipo de item é denominado padrão tabuleiro homogêneo, ou simplesmente padrão homogêneo. Por exemplo, os padrões A e B da Figura 2.02

Morábito e Arenales (2000) formularam o problema da geração de padrões de corte tabuleiro como um problema quadrático inteiro. Propuseram uma heurística, sugerindo o Modelo 3 (2.07 à 2.10), descrito a seguir:

$$\text{Modelo 3 : Max } \sum_{i=1}^n \sum_{k=1}^n \pi_{ik} \cdot r_i \cdot z_k \quad (2.07)$$

$$\text{S.a : } \begin{cases} \sum_{i=1}^n \ell_i \cdot r_i \leq L & (2.08) \\ \sum_{k=1}^n w_k \cdot z_k \leq W & (2.09) \\ r_i \geq 0, z_k \geq 0 & (i, k = 1, 2, \dots, n) & (2.10) \end{cases}$$

onde:  $r_i$  é o número de vezes que o comprimento  $\ell_i$  é cortado;

$z_k$  é o número de vezes que a largura  $w_k$  é cortada;

e, no caso não-exato:

$$\pi_{ik} = \begin{cases} \text{Max}_j \{ \ell_j \times w_j \text{ tal que } \ell_i \geq \ell_j \text{ e } w_i \geq w_j \} \\ 0 \text{ caso contrário} \end{cases}$$

no caso exato:

$$\pi_{ik} = \begin{cases} \pi_k \text{ se, para algum item } s, \ell_s = \ell_i \text{ e } w_s = w_k, \\ 0 \text{ caso contrário} \end{cases}$$

O Modelo 3, tanto para o caso exato quanto o não exato, é de difícil solução, o que levou Morábito e Arenales a não explorarem esta formulação.

Yanasse e Morábito (2006) fazem uma revisão de diversos modelos de programação linear inteira e não-linear para o problema do corte guilhotinado em dois estágios. Dentre os quais destacamos o modelo proposto por Scheithauer (2002), para ob-

tenção de um padrão tabuleiro não-exato, que aqui denominamos Modelo 4 (2.11 à 2.20) e descrevemos a seguir:

Seja :  $L, W$  → Comprimento e Largura do Objeto

$\ell_i, w_i$  → Comprimento e Largura do item  $i$

$J = \lfloor L/\ell_{\min} \rfloor$  → Nr. máx. de faixas ao longo de  $L$

$K = \lfloor W/w_{\min} \rfloor$  → Nr. máx. de faixas ao longo de  $W$

Variáveis:  $L_j$  → Comprimento da  $j$ -ésima faixa ao longo de  $L$

$W_k$  → Comprimento da  $k$ -ésima faixa ao longo de  $W$

$x_{ijk}$  →  $\begin{cases} 1 & \text{se o item } i \text{ esta no retângulo } L_j \times W_k \\ 0 & \text{caso contrário} \end{cases}$

$$\max \sum_{i=1}^n \sum_{j=1}^J \sum_{k=1}^K (\ell_i \cdot w_i \cdot x_{ijk}) \quad (2.11)$$

$$\text{S.a : } \sum_{j=1}^J L_j \leq L \quad (2.12)$$

$$\sum_{k=1}^K W_k \leq W \quad (2.13)$$

$$\sum_{i=1}^n x_{ijk} \leq 1, \quad \forall j, k \quad (2.14)$$

$$\sum_{i=1}^n \ell_i \cdot x_{ijk} \leq L_j, \quad \forall j, k \quad (2.15)$$

$$\sum_{i=1}^n w_i \cdot x_{ijk} \leq W_k, \quad \forall j, k \quad (2.16)$$

$$\sum_{j=1}^J \sum_{k=1}^K x_{ijk} \leq b_i, \quad \forall i \quad (2.17)$$

$$L_j \geq L_{j+1}, \quad \forall j \quad (2.18)$$

$$W_k \geq W_{k+1}, \quad \forall k \quad (2.19)$$

$$\text{Com } x_{ijk} \in \{0,1\}; L_j, W_k \geq 0; i = 1..n; j = 1..J; k = 1..K \quad (2.20)$$

Onde: (2.11) função objetivo que maximiza o valor total das peças cortadas no padrão;

(2.12) restrição devido ao comprimento ( $L$ ) do objeto;

(2.13) restrição devido à largura ( $W$ ) do objeto;

(2.14) impõem que no máximo um item esteja alocado no retângulo  $L_j \times W_k$ ;

(2.15) e (2.16) garantem que o item  $\ell_i \times w_i$  alocado no retângulo  $L_j \times W_k$  não excede o comprimento  $L_j$  nem a largura  $W_k$ ;

(2.17) limita a alocação do item  $i$  no objeto à sua demanda  $b_i$ ;

(2.18) e (2.19) visam eliminar as simetrias, reduzindo o espaço solução.

A partir do Modelo 4, podemos obter o Modelo 5, para o caso do tabuleiro exato, substituindo-se as restrições:

$$\sum_{i=1}^n \ell_i \cdot x_{ijk} \leq L_j, \quad \forall j, k \quad (2.15)$$

$$\sum_{i=1}^n w_i \cdot x_{ijk} \leq W_k, \quad \forall j, k \quad (2.16)$$

pelas restrições

$$\sum_{i=1}^n \ell_i \cdot x_{ijk} \leq L_k, \quad \forall j, k \quad (2.15a)$$

$$L_k \leq \sum_{i=1}^n \ell_i \cdot x_{ijk} + M_\ell \left( 1 - \sum_{i=1}^n x_{ijk} \right), \quad \forall j, k \quad (2.15b)$$

$$\sum_{i=1}^n w_i \cdot x_{ijk} \leq W_k, \quad \forall j, k \quad (2.16a)$$

$$W_k \leq \sum_{i=1}^n w_i \cdot x_{ijk} + M_w \left( 1 - \sum_{i=1}^n x_{ijk} \right), \quad \forall j, k \quad (2.16b)$$

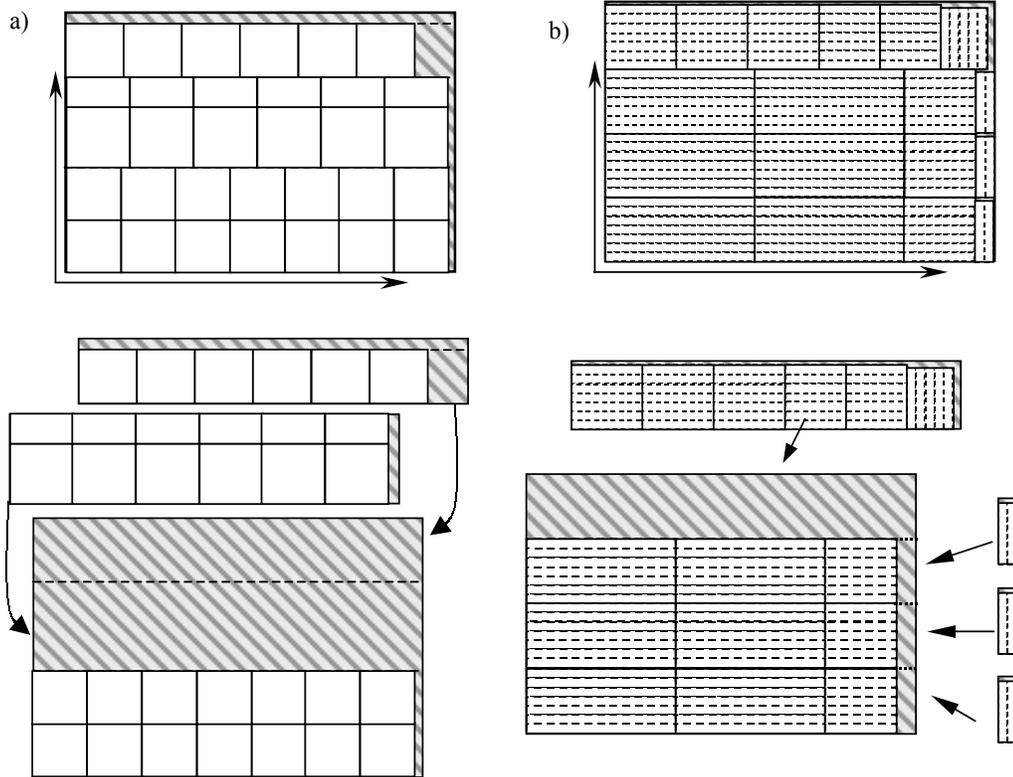
onde  $M_\ell$  e  $M_w$  são valores suficientemente grandes (por exemplo,  $M_\ell = \ell_{max}$  e  $M_w = w_{max}$ ).

As características apresentadas acima garantem aos padrões de corte tabuleiros um menor tempo de processamento de máquina (Arenales *et al*, 2004), aumentando a produtividade da seccionadora, o que é de grande interesse em ambientes de grande demanda, pois reduz a probabilidade de um gargalo na fase de corte da matéria prima para a produção. Com tudo, este ganho de produtividade traz consigo, em geral, um aumento na perda de material (Katsurayama, 2002).

### **Padrão Tabuleiro Composto**

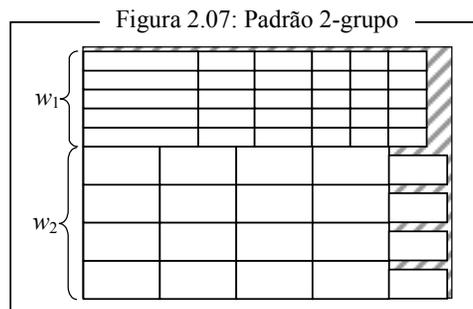
Em uma análise dos padrões de corte adotados pela Fábrica de móveis Luapa, discutida com maiores detalhes na seção 3.7, pôde-se observar uma tendência da indústria em construir padrões que muito se aproximam dos padrões tabuleiros. Na Figura 2.06 a seguir, ilustra-se dois padrões de corte utilizados na Indústria. Note que se orientarmos os padrões, da esquerda para a direita e de baixo para cima, veremos no canto inferior esquerdo um bloco principal que compõe um padrão tabuleiro. Agregado a esse bloco podemos ver blocos menores, também sob a forma de padrões tabuleiros, que juntos compõem o padrão da indústria.

Figura 2.06: Padrões da Indústria



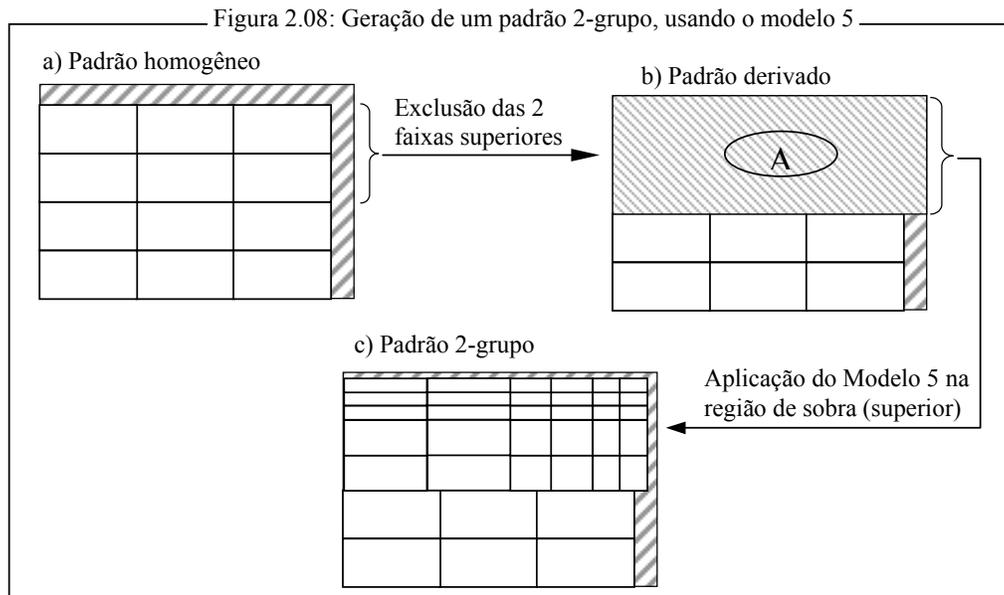
Esse tipo de padrão em 2-estágios, obtido pela composição de padrões tabuleiros, será chamado de padrão tabuleiro composto. O padrão tabuleiro composto é um padrão da classe  $n$ -grupo (Gilmore e Gomory, 1965). Na análise feita dos padrões da indústria, observamos uma tendência em trabalhar com padrões compostos por até três grupos ( $n \leq 3$ ). Esses padrões apresentam uma velocidade de produção próxima a dos padrões tabuleiros, e um melhor aproveitamento da matéria-prima.

Para obtenção de padrões tabuleiros compostos, podemos utilizar o Modelo 1, considerando faixas de no máximo duas larguras distintas, conforme mostra a figura 2.07.



Podemos ainda, tomar um padrão tabuleiro e dele excluir algumas faixas, gerando um padrão derivado do padrão tabuleiro com uma maior região de sobre (região A na

Figura 2.08b). Se usarmos nesta região de sobra o Modelo 5, criaremos um padrão 2-grupo, conforme a Figura 2.08c.



### 2.3 Modelo de Gilmore e Gomory para o $PCE_2$

A forma clássica de resolver o  $PCE_2$  é, supondo conhecidos os padrões de corte  $A_j$  ( $j = 1, \dots, m$ ) e dado uma demanda de  $b_i$  itens ( $\ell_i \times w_i, i=1, \dots, n$ ), determinar quantas vezes cada padrão de corte deve ser usado para cortar um conjunto de objetos de dimensões ( $L \times W$ ) a fim de atender a demanda. Gilmore e Gomory (1961) propõem a solução do Modelo 6 (2.21 à 2.23) através do método de geração de colunas.

$$\min z = \sum_{j=1}^m c_j \cdot x_j \quad (2.21)$$

$$\text{S.a: } \sum_{j=1}^m a_{ij} \cdot x_j \geq b_i, \quad \forall i, i = 1, 2, \dots, n \quad (2.22)$$

$$x_j \in \mathbb{Z}_+, \quad \forall j, j = 1, 2, \dots, m \quad (2.23)$$

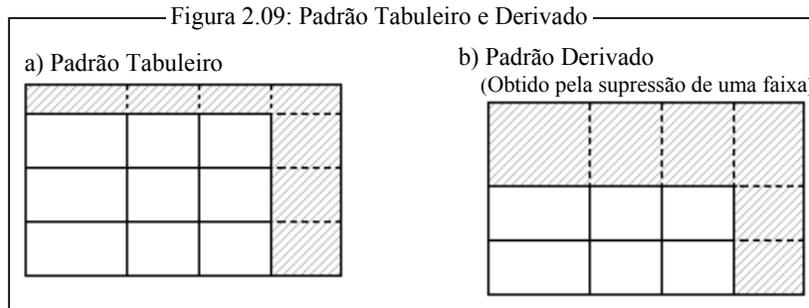
onde  $x_j$  indica a quantidade de vezes que o padrão de corte  $A_j$  foi aplicado, para o atendimento à demanda, e  $c_j$  é a variável custo associada ao padrão de corte  $A_j$ .

Perin e Rangel (1989) desenvolveram o sistema CorteBi, que resolve a relaxação linear do Modelo 6 pelo método de geração de colunas, onde a cada iteração do método simplex um padrão de corte em 2-estágios não-exato é gerado (Modelo 1). A solução inteira é obtida por arredondamento da solução ótima da relaxação linear, e o critério de otimização é a minimização do número de objetos cortados, isto é  $c_j=1$  no modelo 6. Se tomarmos  $c_j=P_j$  no modelo 6, onde  $P_j$  representa a perda do padrão de corte  $A_j$ , estabelecemos o problema de minimizar as sobras (Arenales *et al*, 2004).

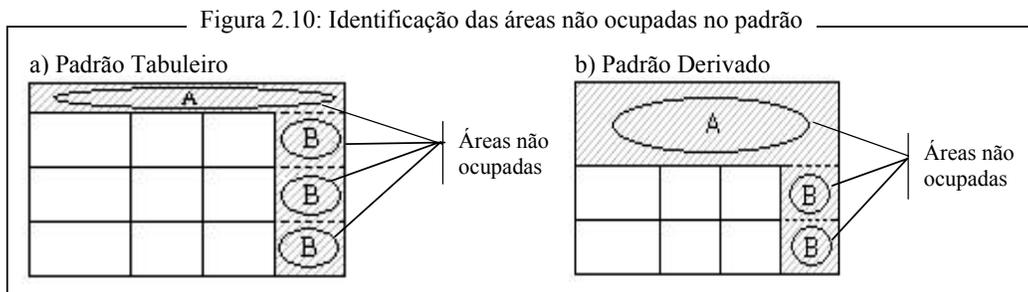
## 2.4 Heurísticas para solução do PCE<sub>2</sub> usando padrões tabuleiros compostos

Como visto na seção 2.2, os padrões de corte adotados pela indústria estudada muito se aproximam dos padrões tabuleiros. Assim, buscaremos nesta seção desenvolver uma heurística para resolução do PCE<sub>2</sub> usando padrões tabuleiros compostos.

Tomaremos como ponto de partida padrões tabuleiros ou derivados destes, pela exclusão de algumas faixas (Figura 2.09).



Num segundo momento, identificamos as áreas não ocupadas nos padrões, conforme a figura 2.10. Na área não ocupada na parte superior dos padrões (região A) aplicaremos, quando possível, um modelo para a geração de um padrão tabuleiro exato, e nas áreas não ocupadas de cada uma das faixas dos padrões (regiões B) aplicaremos um modelo para a geração de um padrão tabuleiro não exato. Ao final do processo, as áreas não ocupadas que restarem serão as sobras dos padrões



Observamos que os padrões utilizados pela indústria conseguem apresentar bons resultados operacionais com índices de perdas dentro de uma faixa aceitável. As heurísticas aqui apresentadas buscam, com base nessas observações, construir um conjunto particular de padrões de corte (*Pool*), onde a característica básica deles será o fato de serem padrões tabuleiros compostos. Com isso, no atendimento à demanda dos itens será considerado apenas os padrões inclusos neste *Pool*.

### Heurística 1

A heurística 1, aqui apresentada, será composta de quatro fases e tomará por base a heurística apresentada por Figueiredo e Rangel (2005).

**Fase 1: Leitura e rearranjo dos dados**

- 1.1) Leitura dos dados:
  - Dimensões do objeto ( $L, W$ );
  - Dimensões dos  $n$  itens ( $\ell_i, w_i$ ) e suas demandas ( $b_i$ ).
- 1.2) Se for considerada a rotação dos itens, gerar novos itens rotacionando os originais;
- 1.3) Ordenação dos itens (pela largura e pelo comprimento).

**Fase 2: Montagem de padrões tabuleiros e derivados**

2.1) Tomamos cada item  $i$  como base para construção de até duas faixas exatas, que deverão ter largura  $w_i$ . Diremos que o item  $i$  é o item pivô destas faixas.

1ª) Homogênea (Figura 2.11a);

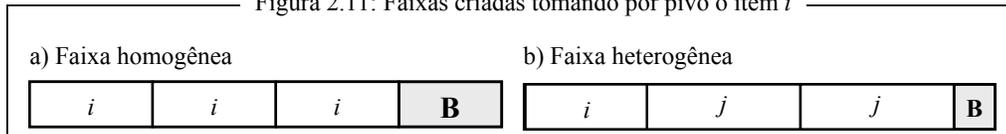
2ª) Heterogênea (Figura 2.11b), que será obtida resolvendo o problema da mochila irrestrito (2.24 à 2.26) a seguir, considerando apenas os itens  $j$  tais que  $w_j = w_i$ .

$$\max \sum_{j=1}^n \ell_j \cdot w_j \cdot x_j \quad (2.24)$$

$$\text{S.a: } \sum_{j=1}^n \ell_j x_j \leq L \quad (2.25)$$

$$x_j \begin{cases} = 1 & \text{se } i = j \\ \geq 0 & \text{se } w_i = w_j, \quad j = 1, 2, \dots, n \\ = 0 & \text{se } w_i \neq w_j \end{cases} \quad (2.26)$$

Figura 2.11: Faixas criadas tomando por pivô o item  $i$



A segunda faixa só será construída se houver ao menos mais um item  $j$  tal que  $w_j = w_i, j \neq i$ . Neste caso, um item  $i$  (item pivô) será incluído uma única vez na faixa;

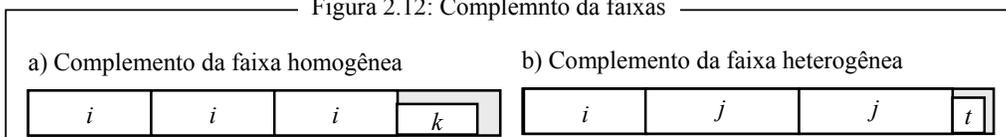
2.2) Para cada área não ocupada das faixas criadas em 2.1 (regiões B, Figura 2.11) nos resolvemos o problema da mochila irrestrito (2.27 à 2.28) considerando itens com  $w_k \leq w_i$  (Figura 2.12):

$$\max \sum_{j=1}^n \ell_j \cdot w_j \cdot x_j \quad (2.27)$$

$$\text{S.a: } \sum_{j=1}^n \ell_j \cdot x_j \leq L_B \quad (2.28)$$

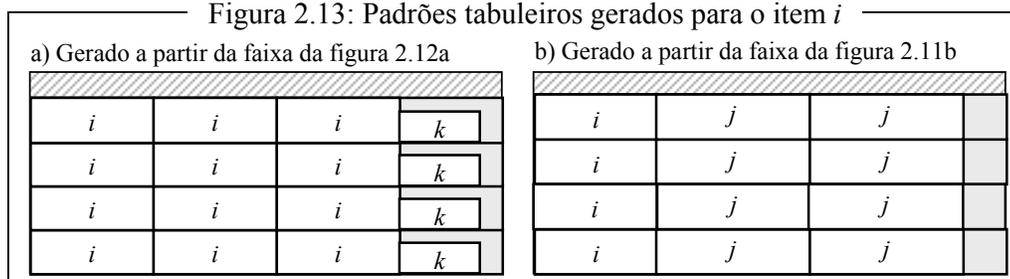
$$x_j \begin{cases} \geq 0 & \text{se } w_j \leq w_i \\ = 0 & \text{se } w_j > w_i \end{cases} \quad j = 1, 2, \dots, n \quad (2.29)$$

Figura 2.12: Complemento da faixas



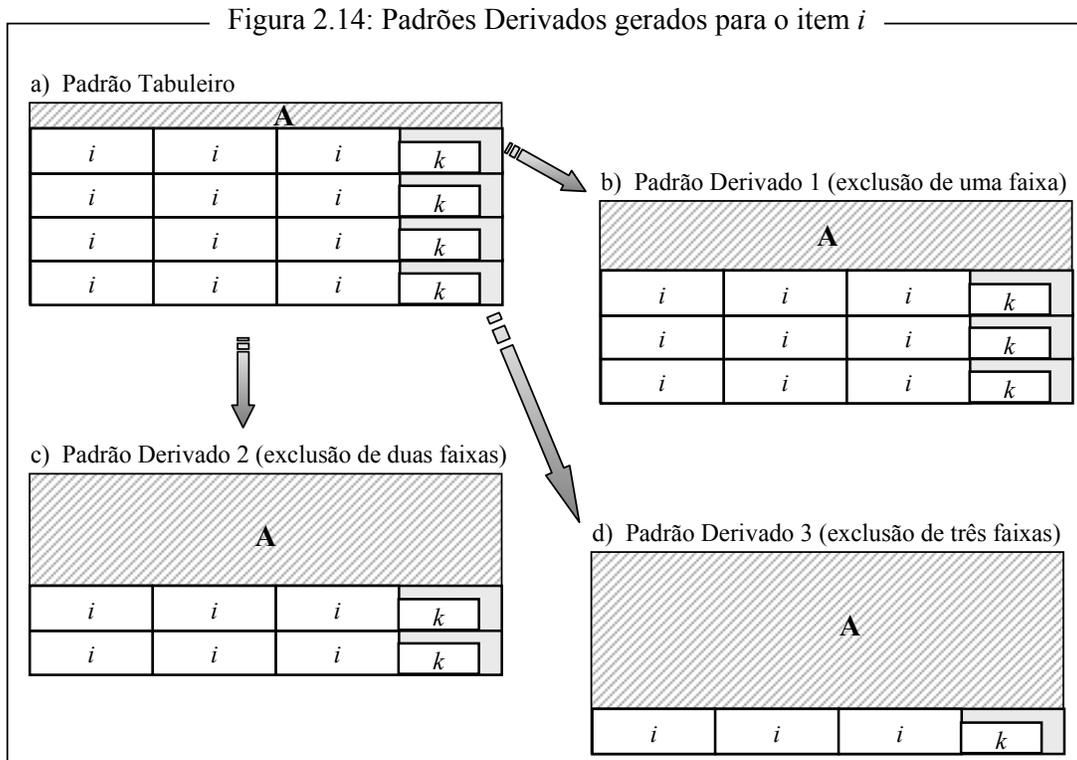
Se for possível ocuparmos a região B, com itens de largura menor que a largura da faixa, estaremos garantindo a construção de padrões tabuleiros, no passo 2.3, com menor índice de sobra.

- 2.3) Para cada faixa criada em 2.1 e 2.2, criamos um padrão tabuleiro, reproduzindo a faixa  $\lfloor W/w_i \rfloor$  vezes (Figura 2.13);



O procedimento de construção de ao menos uma faixa para cada item, passo 2.1, nos garante que todos os itens estarão contemplados em pelo menos um padrão tabuleiro.

- 2.4) Para cada padrão tabuleiro, gerado em 2.3, criamos novos padrões, derivados do padrão tabuleiro pela exclusão de algumas faixas do padrão (Figura 2.14);



Se um padrão tabuleiro possuir  $t$  faixas, poderemos construir até  $t-1$  padrões derivados. No entanto, quando  $t$  for um número muito grande encontrar os  $t-1$

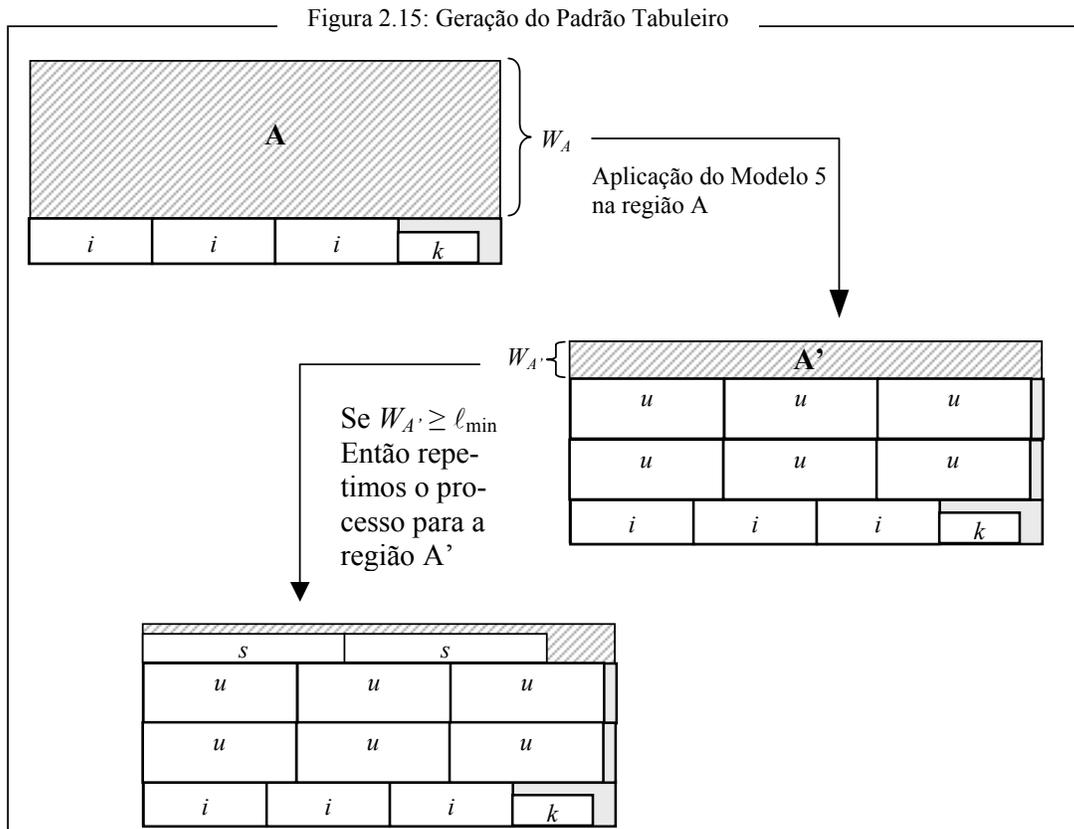
padrões derivados pode ser inviável. Neste caso, podemos estabelecer um número  $k$  (arbitrário) de padrões que desejamos construir para cada item pivô e determinar o passo, ou seja, o número de faixas que deverão ser extraídas para a geração de um novo padrão derivado. Esse passo pode ser calculado pela fórmula:

$$passo = \lceil (quantidade\ de\ faixas) / k \rceil$$

- 2.5) Criamos uma matriz de sobras onde registramos, para cada padrão criado em 2.3 e 2.4, as dimensões da sobra resultante da eliminação das faixas (região A, figura 2.14);

### Fase 3: Montagem do *Pool* de padrões tabuleiros compostos

Cada sobra identificada no passo 2.5, da fase 2, é tomada como um novo objeto, de dimensões  $L \times W_{A'}$ , e sobre ele aplicamos o Modelo 5 (Figura 2.15).



Ao final obteremos um *Pool* formado por no máximo  $2 \cdot (k+1) \cdot n$  Padrões distintos, ou  $2 \cdot (k+1) \cdot 2n$  se considerarmos a rotação do objeto. A expressão  $(k+1)$  tem origem no fato de montarmos para cada item um padrão tabuleiro e até  $k$  padrões derivados deste. O termo  $n$  corresponde a quantidade de itens demandados.

#### Fase 4: Resolução do PCE<sub>2</sub>

O PCE<sub>2</sub> é resolvido de forma aproximada pelo Modelo 7 (2.30 à 2.32), usando o Modelo clássico do PCE<sub>2</sub>, minimizando o número de objetos cortados( $x_j$ ), considerando apenas os padrões contidos no *Pool* gerados na fase 3, ou seja  $A_j \in \{Pool\}$ .

$$\min z = \sum_{j=1}^m x_j \quad (2.30)$$

$$\text{S.a } \sum_{j=1}^m A_j \cdot x_j \geq b, \quad (2.31)$$

$$x_j \in Z_+, A_j \in \{Pool\} \quad \forall j, j=1,2,\dots,m \quad (2.32)$$

#### Heurística 2

Esta heurística é uma alternativa à heurística anterior, diferenciando apenas na fase 3, montagem do *Pool*. Ao invés de aplicarmos o Modelo 5 sobre as sobras dos padrões de corte, identificadas no tópico 2.5 da fase 2, aplicamos o segundo estágio do procedimento proposto por Gilmore e Gomory (1965) para a geração de padrões em dois estágios, resolvendo o problema da mochila (2.33 à 2.37) utilizando apenas uma das faixas criadas nos passos 2.1 e 2.2 da fase 2.

$$\max \sum_{i=1}^m v_i \cdot x_i \quad (2.33)$$

$$\text{S.a } : \sum_{i=1}^m w_i \cdot x_i \leq W_A \quad (2.34)$$

$$x_i \leq y_i \cdot W, \quad i = 1, 2, \dots, m \quad (2.35)$$

$$\sum_{i=1}^m y_i \leq 1 \quad (2.36)$$

$$x_i \geq 0, y_i \in \{0,1\} \quad i = 1, 2, \dots, m \quad (2.37)$$

onde:  $w_i$  é a largura da faixa  $f_i$ , pertencente ao conjunto de faixas obtido na fase 2;  
 $x_i$  é a quantidade de vezes que a faixa  $f_i$  é utilizada;  
 $y_i$  é a igual a 1 se a faixa  $f_i$  for utilizada e igual a 0 em caso contrário;  
 $v_i$  é a área utilizada da faixa  $f_i$ ;  
 $W_A$  é a largura da faixa de sobra.

## *Capítulo 3*

### **Problemas do Corte na Indústria de Móveis**

---

Neste capítulo tem-se por objetivo dar uma visão geral da indústria moveleira no Brasil, mostrando como o setor se encontra organizado e um perfil do setor. Abordaremos, ainda, o Pólo Moveleiro de Votuporanga, onde tomamos uma indústria característica do Pólo, a Fábrica de Móveis Luapa, para o desenvolvimento do trabalho.

O nosso foco reside na análise dos padrões de corte adotados pela indústria. Faremos uma discussão sobre dois dos principais aspectos determinantes da qualidade de um padrão de corte, aos olhos da indústria: o tempo de corte e a sobra de matéria-prima.

### 3.1 Classificação dos Móveis

Podemos usar quatro critérios para classificar um móvel: o uso a que são destinados, a matéria-prima predominante utilizada na produção, o tipo do produto e forma de produção (Justina, 2004).

#### Uso

Segundo a Associação Brasileira das Indústrias do Mobiliário (Abimóvel, 2005) a produção nacional de móveis, no que tange a sua finalidade, é segmentada em:

- Móveis residenciais, com 60% da produção nacional;
- Móveis de escritório, com 25% da produção nacional;
- Móveis institucionais, que correspondem aos móveis para escolas, consultórios, hospitais, hotéis e similares, com 15% da produção nacional.

#### Matéria-prima predominante

Segundo o IBGE, a classificação dos móveis se dá a partir das matérias-primas predominantes: madeira, metal e plástico. A mais expressiva parcela dos móveis produzidos utiliza a madeira como matéria prima (Abimóvel, 2005), que pode, ainda, ser classificada como:

- Madeira maciça, que pode ser extraída em florestas nativas (madeira de lei) ou em de reflorestamento (como pinus e eucalipto);
- Painel de compensado, produto obtido pela colagem de lâminas de madeira sobrepostas;
- Painel de aglomerado, formado a partir da redução da madeira a partículas que são depois impregnadas com resina sintética para formar um colchão que, pela ação controlada de calor, pressão e umidade, transforma-se no painel;
- Painel de *medium-density fiberboard* (MDF), produzido a partir de fibras de madeira, aglutinadas com resinas sintéticas através de temperatura e pressão, possuindo consistência similar à da madeira.

#### Tipo do produto

Os móveis, de madeira, são classificados em dois tipos: retilíneos, que são lisos, com desenho simples de linhas retas e cuja matéria-prima principal constitui-se de painéis (aglomerados, compensados e MDF's) e torneados, que reúnem detalhes mais so-

fisticados de acabamento, misturando formas retas e curvilíneas. Sua principal matéria-prima é a madeira maciça, podendo também incluir painéis de MDF's, passíveis de serem usinados.

### **Forma de produção**

Quanto à forma de produção, esta pode ser: seriada, quando para distribuição comercial, via varejistas e representantes; e sob encomenda, quando direto para o cliente ou sob um agente intermediário.

## **3.2 A Indústria Moveleira no Brasil**

A indústria moveleira no Brasil conta com 16.000 empresas registradas, em sua grande maioria pequenas empresas, gerando entorno de 190.000 empregos diretos. No entanto, o número total de empresas no setor pode ser bem maior, devido as empresas informais que, segundo estimativas da Abimóvel, totalizam quase o mesmo número das empresas formalizadas (Gorini, 1998). A grande maioria das empresas neste setor são o classificadas, de acordo com padrões mundiais, como de pequeno porte (Justina,2004), caracterizadas pela reunião de diversos processos de produção, envolvendo diferentes matérias primas e uma diversidade de produtos finais. Aspectos técnicos e mercadológicos, no entanto, levam as empresas à se especializarem em uma ou duas linhas de produção (por exemplo, de cozinha e banheiro, estofados) (Azevedo, 2003).

Uma característica marcante da indústria moveleira no Brasil, presente em todo o território nacional, é a forte dispersão geográfica. O setor mostra ainda uma segmentação em dois grupos: um grupo grande, em geral, composto por micro e pequenas empresas; um outro grupo, com menor número de empresas, é composto por indústrias de médio e grande porte que respondem pelas lideranças do mercado, tanto no segmento de móveis de escritório, como no de móveis residenciais. Estes dois grupos vivem realidades e necessidades muito distintas no país (Azevedo, 2003).

Apesar da forte dispersão, a indústria brasileira apresenta uma outra característica que é o agrupamento em pólos regionais. Estes pólos estão localizados basicamente na região Sul, com 45,95% da produção nacional, e na região Sudeste, com 39,57%. As demais regiões apresentam índices muito baixos, se comparados a estes, sendo 6,92% no Nordeste, 3,97% no Norte e 3,58% no Centro-Oeste. Dentre os pólos, nós destacamos os de: Bento Gonçalves (RS), São Bento do Sul (SC), Araçatuba (PR), Mirassol, S.J. do Rio Preto, Votuporanga e Grande S. Paulo (SP). Os pólos são compostos por

empresas que, em geral, apresentam um processo de produção descontínuo, o que permite a coexistência de máquinas modernas e obsoletas e até mesmo processos manuais em uma mesma planta (Stipp, 2002).

### **3.3 O Pólo de Votuporanga**

A cidade de Votuporanga esta situada no Noroeste do Estado a 520 km da capital paulista. Favorecida no passado por fatores como a cultura do café na região, o avanço da estrada de ferro Alta Araraquarense, a rota de gado proveniente do Mato Grosso com destino a Barretos e uma rica mata nativa na região, com uma variedade de espécies de madeira de lei, Votuporanga viu cedo surgirem as primeiras serrarias e posteriormente as primeiras fábricas de móveis, que na época apresentavam um caráter artesanal.

A partir de 1962, com a fundação da Indústria de Móveis A. B. Pereira, o cenário de produção artesanal começou a mudar. A indústria implantou na região um processo em que os trabalhadores realizavam poucas tarefas, simples e repetitivas. Isso permitiu reduzir os tempos com processos e trabalhar com uma mão-de-obra menos especializada. O que, conseqüentemente, contribuiu para um menor custo de produção.

A criação do Plano de Amparo e Incentivo Industrial de Votuporanga (PLAMI-VO) em 1970, e posteriormente da Associação Industrial de Votuporanga (AIRVO) reforçou, na região, a cultura de cooperação, levando o Pólo de Votuporanga a um grande crescimento, chegando a ser o segundo maior do país na década de 90, atrás apenas do Pólo de Bento Gonçalves (RS) (Cavali, 2004).

### **3.4 Uma empresa característica**

O Pólo de Votuporanga, formado em 90% por micro e pequenas indústrias, produz essencialmente moveis residenciais. Sendo assim, tomamos uma pequena indústria característica do pólo, a Fábrica de Móveis Luapa, para nossas observações. A indústria em questão produz seis tipos de móveis de dormitórios, em quatro cores, conforme a tabela de produtos (Tabela 3.01). A produção é feita por lotes, obedecendo a uma programação semanal e de acordo com a carteira de pedidos. Em cada lote é produzido apenas um tipo produto (por exemplo Armário de 3 Portas na cor Marfim), os lotes são de tamanho fixo e possuem um conjunto de padrões de cortes que são utilizados para produzi-los.

Tabela 3.01: Produção mensal da Fabrica de Moveis Luapa

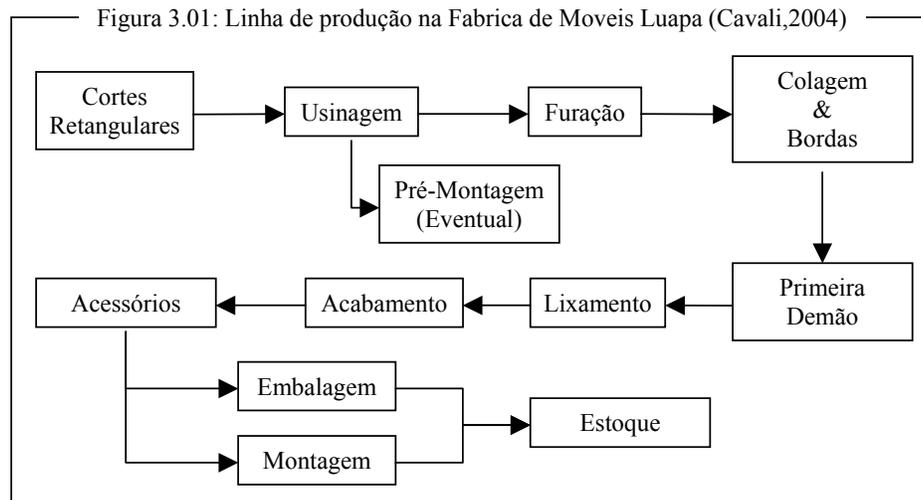
Produto	Demanda Mensal	Dimensão do Lote	Cores
Armários	1000	(3p) 60	Cerejeira Marfín
		(4p) 45	
		(5p) 40	
Cômodas	200	200	Móгно
Criados	300	300	Tabaco
Camas	600	160	

### 3.5 Confeção de um móvel

O processo de confeção de um móvel envolve diversas etapas e equipamentos, relacionados a seguir:

- Elaboração do projeto do móvel;
- Montagem de um protótipo para avaliação;
- Especificação dos padrões de corte que serão utilizados para produção de um lote;
- Entrada do móvel em produção:
  - Setor de corte primário: Este setor, em geral, é composto por dois funcionários que operam uma máquina seccionadora, na qual os objetos são cortados nos itens que irão compor o móvel;
  - Setor de corte secundário: composto por máquinas de corte menores, é utilizado para o processo de aparo das sobras, cortar itens que foram processados de forma agrupada (seção 3.7);
  - Setor de furação: Neste setor são utilizados um ou dois funcionários operando até duas máquinas. Nele são feitos os furos nos itens, para o encaixe das peças e montagem do móvel;
  - Setor de montagem: Este é composto por diversas máquinas, entre elas: lixadora, prensa, entre outras. Nele as peças são montadas, e preparadas para a pintura. É o setor que agrega o maior número de funcionários;
  - Setor de pintura: Este setor é formado por um conjunto de máquinas que operam interligadas. O conjunto é operado com um número de funcionários que varia de 3 a 4. O papel deles no processo de produção é preparar o conjunto, efetuando a limpeza das máquinas, a carga das tintas e o monitoramento do processo de pintura das peças do móvel;
  - Setor de Estocagem: Neste setor, os móveis recebem uma última vistoria, são embalados e armazenados.

Cavali (2004) elaborou um esquema da linha de produção da Fábrica de Móveis Luapa, que reproduzimos na Figura 3.01.



A Luapa possui aproximadamente 10 máquinas, sendo quase todas manuais. Neste trabalho vamos nos ater ao setor de corte primário. Outras informações, referentes ao planejamento da produção, podem ser obtidas em Cavali (2004) e Silva (2003).

### 3.6 O processo de corte do Objeto

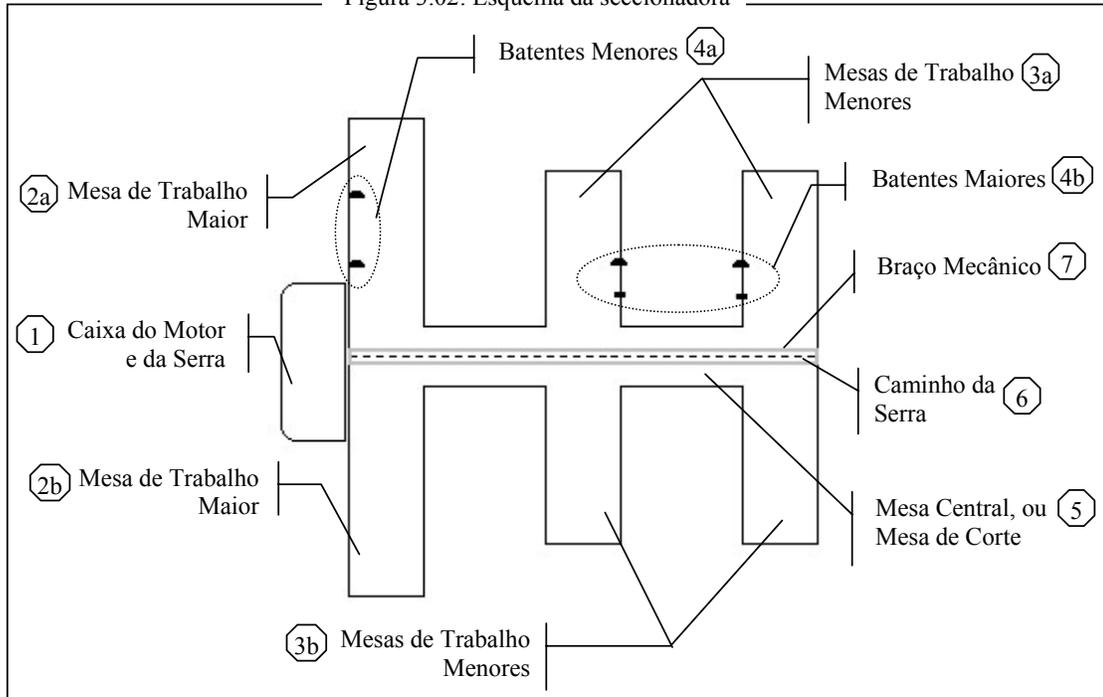
Como visto na seção 3.5, no setor de corte primário o objeto é cortado utilizando-se um seccionadora. Existem no mercado seccionadoras automáticas, que efetuam os cortes dos objetos por meio de um software, minimizando a participação humana no processo. Contudo, em nosso estudo tomamos por base a seccionadora da Fábrica de Móveis Luapa, que é uma seccionadora básica com uma alta participação humana no controle do processo de corte.

#### Seccionadora

A seccionadora é composta por um conjunto de mesas com uma serra circular atrelada a um “carro”, que quando acionado percorre toda a extensão da mesa, cortando os objetos. A seccionadora observada, apresenta uma velocidade de corte de 14 m/min, onde todo o processo de carga dos objetos e o controle dos cortes é manual, o que torna crucial a escolha de um bom conjunto de padrões de corte.

A seguir temos o desenho esquemático de uma seccionadora (Figura 3.02).

Figura 3.02: Esquema da seccionadora



Dentre os itens identificados no esquema da seccionadora, destacamos:

- As mesas de trabalho, que estão divididas em duas áreas:
  - Mesa de trabalho maior (2a e 2b): Manipulação e corte das faixas extraídas dos objetos;
  - Mesas de trabalho menores (3a e 3b): manipulação e corte dos objetos;
- Batentes, são calços metálicos instalados sobre eixos graduados, fixos nas mesas de trabalho. Estão divididos em dois grupos:
  - Batentes Maiores (4b): estão localizados sobre as mesas de trabalho menores (3a), determinam a largura da faixa que será cortada do objeto, é possível o ajuste para o corte de até duas larguras distintas;
  - Batentes Menores (4a): estão localizados sobre a mesa de trabalho maior (2a), permitem também duas regulagens de corte, são utilizados para o corte dos itens nas faixas obtidas dos objetos, sendo possível o ajuste para o corte de até dois itens distintos;
- O braço mecânico (7), é uma coluna metálica que, quando iniciado o processo de corte, abaixa fixando o material a ser cortado e, ao término do processo, sobe liberando o manuseio do material. O tempo estimado para as ações do braço mecânico é de 5s.

## **Corte do objeto**

O corte de um objeto inicia-se com a escolha do padrão de corte a ser adotado. O processo de corte é um conjunto de ações necessárias para a manipulação da seccionadora e do objeto para a obtenção dos itens. Nele temos:

- **Ajuste dos Batentes:** o tempo de ajuste dos batentes oscila entre 50s e 2min, dependendo da quantidade de batentes a serem ajustados. O ajuste mais comum, dois batentes maiores (4b) e um dos batentes menores (4a), leva 50s para ser realizado.
- **Carga dos objetos:** estes são empilhados sobre as mesas de trabalho menores (3b), respeitando o limite da capacidade de corte da máquina, que é dado em função da espessura máxima que a máquina admite cortar, no presente caso é de 60mm. O tempo de carga depende da espessura do material, o MDF de 3mm consome mais tempo de carga, entorno de 1min 50s, os MDF's de 9, 12, 15 e 20mm consomem menos tempo, entorno de 1min. A diferença se dá em função da quantidade de objetos a serem carregados (20 no caso do MDF de 3mm) e do objeto de 3mm ser muito maleável, o que dificulta o manuseio.
- **Ajuste dos objetos:** após empilhados sobre a mesa, os objetos precisam ser ajustados, para que sejam cortados por igual. O tempo de ajuste verificado foi de 1min para os MDF's de 3mm e 30s para os demais.
- **Corte longo:** quando a serra percorre toda a mesa de corte (5), efetuando o corte de uma faixa do objeto. O tempo de realização de um corte longo é de 20s, considerando além do tempo de corte a ação do braço mecânico (7) que, como visto, consome 5s.
- **Manuseio do material:** após um corte longo, a faixa resultante é rotacionada e posicionada sobre a mesa de trabalho maior (2b), para que sejam efetuados os cortes curtos, que irão gerar os itens. O tempo gasto no manuseio do material é 40s para os MDF's de 3mm e 30s para os demais.
- **Corte curto:** quando a serra percorre apenas o necessário para o corte dos itens das faixas obtidas no corte longo. O tempo médio para a realização deste corte é de 8s

Os tempos mencionados são aproximados e correspondem as médias dos tempos tomados em visita à Empresa, realizada no período de 21 à 25 de novembro de 2005.

A seguir, acompanharemos as etapas de corte de dois objetos, para observar os tempos em dois processo práticos.

### **• Padrão de Corte A**

O padrão de corte A (Figura 3.03a) é um padrão em 3-estágios, que, como veremos, nos obriga a uma série de manuseios no objeto e ajustes nos batentes (Figuras 3.03b à 3.06) que o tornam um padrão ruim para o processo de corte, apesar de ter um índice de perda de 4,89%.

Figura 3.03: 1º Estágio - Extração da 1ª faixa e aparo da sobra

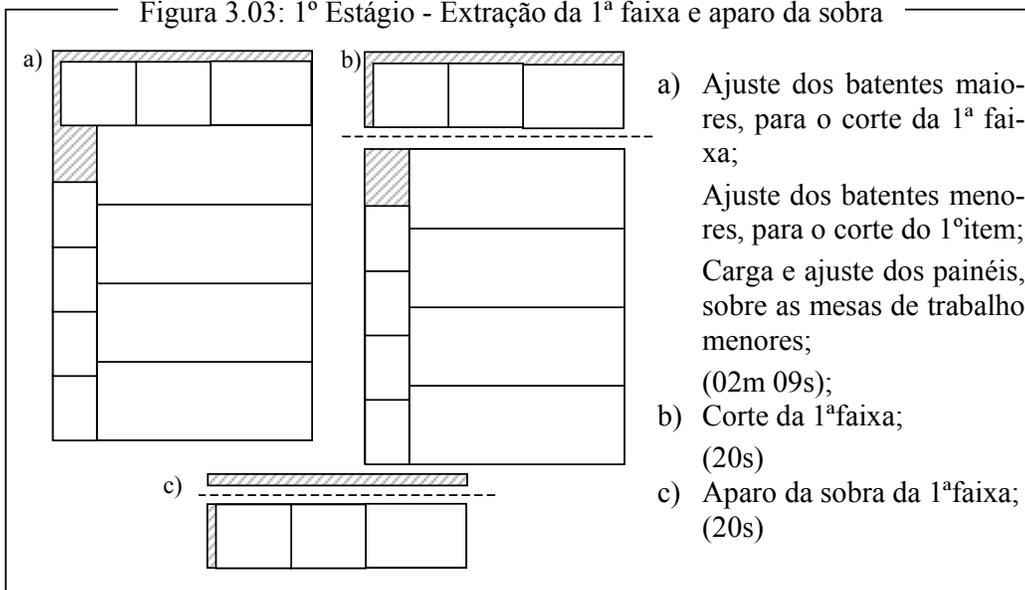


Figura 3.04: 2º Estágio - Corte da 1ª faixa

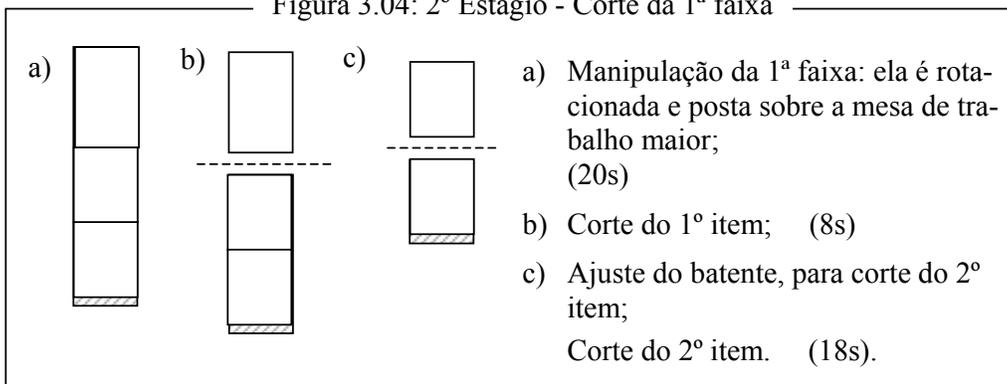


Figura 3.05: 2º Estágio - Corte da 2ª faixa  
- Aparo do 3º item da 1ª faixa

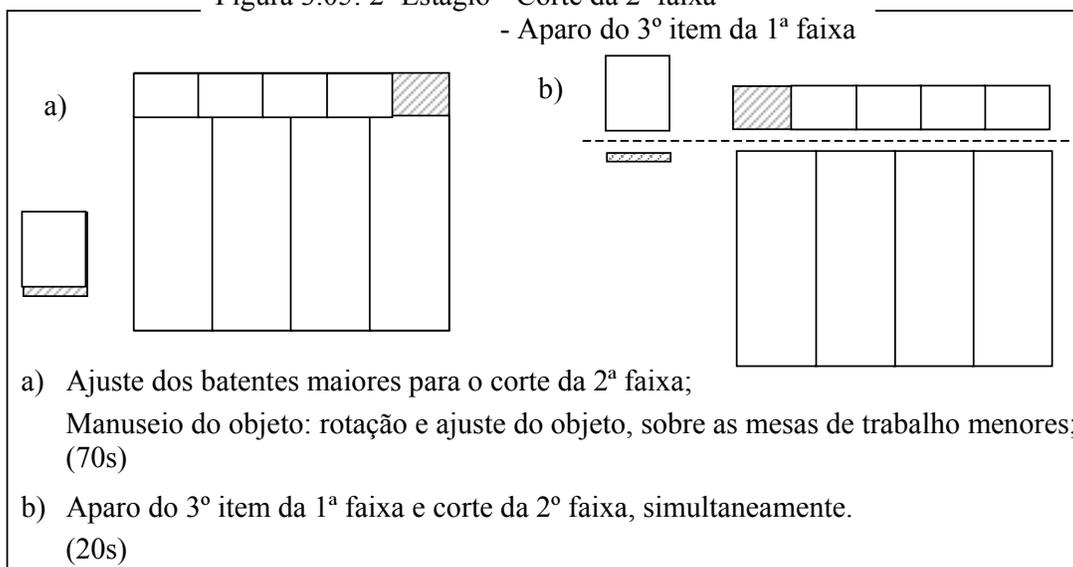
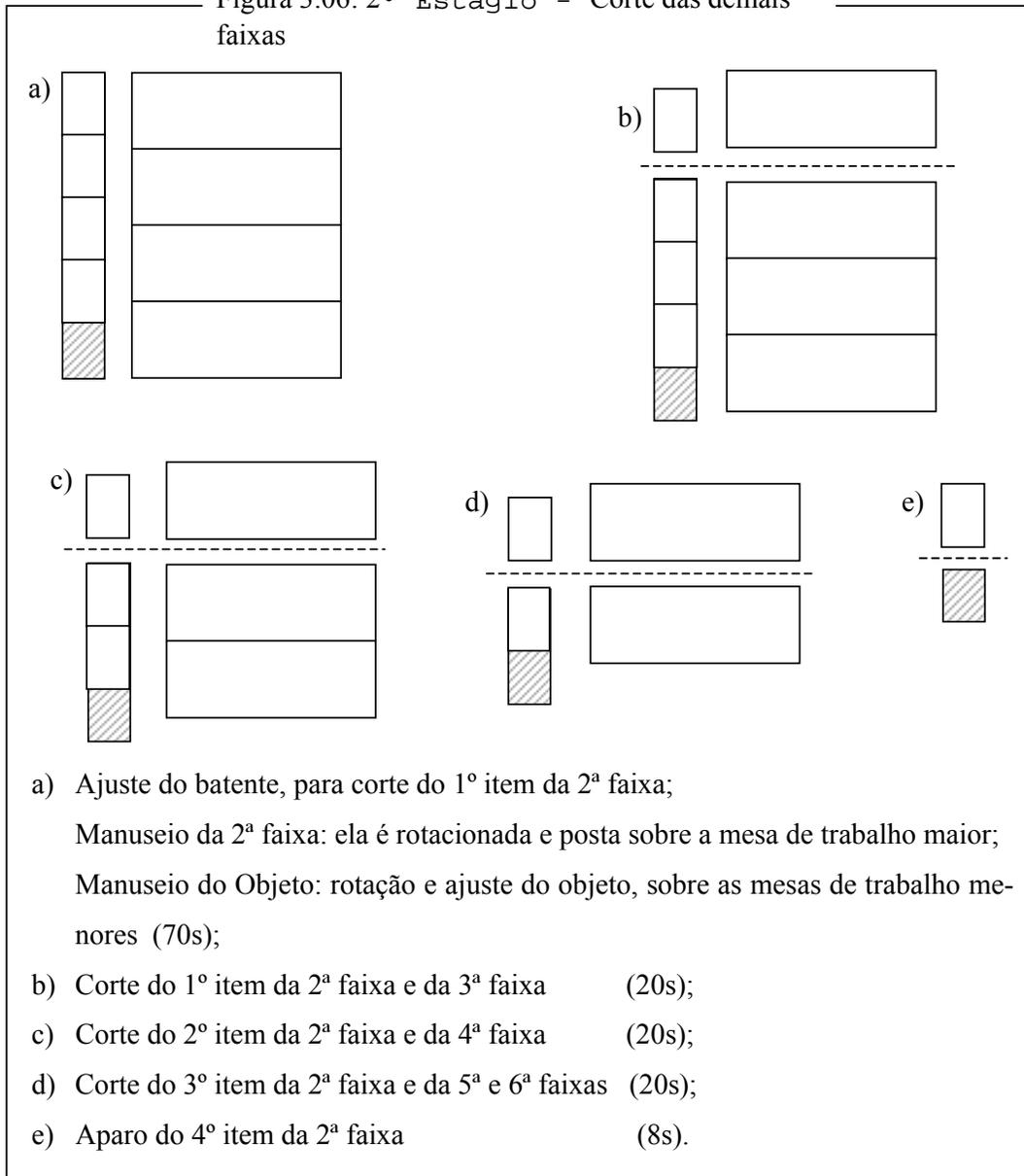


Figura 3.06: 2º Estágio - Corte das demais faixas



O processo exibido nas figuras 3.03b à 3.06 irão consumir entorno de 7m 23s para o corte completo do objeto, de acordo com o padrão de corte A (Figura 3.03a)

• **Padrão de Corte B**

Suponhamos, agora, uma pequena alteração no padrão A, tornando-o um padrão de corte em 2-estágios (Figura 3.07a), e observemos os reflexos no processo de realização do corte do objeto (Figuras 3.07b à 3.13).

Figura 3.07: 1º Estágio - Extração da 1ª

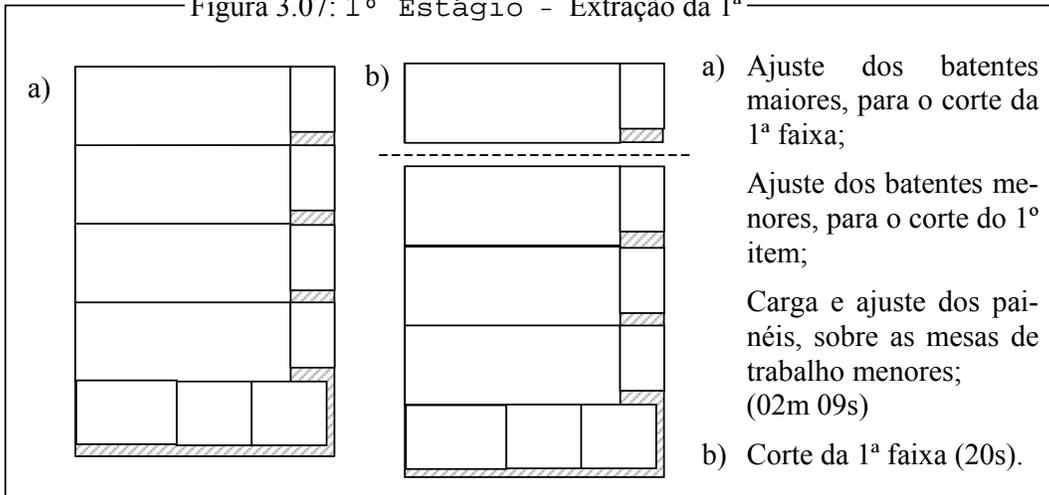


Figura 3.08: 1º Estágio - Extração da 2ª faixa

2º Estágio - Corte dos itens da 1ª faixa

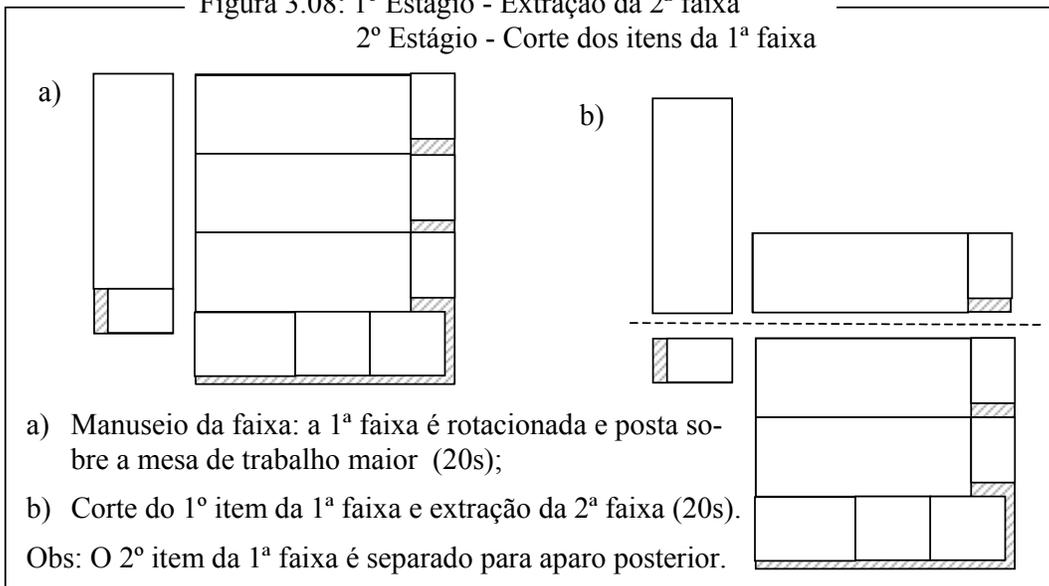


Figura 3.09: 1º Estágio - Extração da 3ª faixa

2º Estágio - Corte dos itens da 2ª faixa

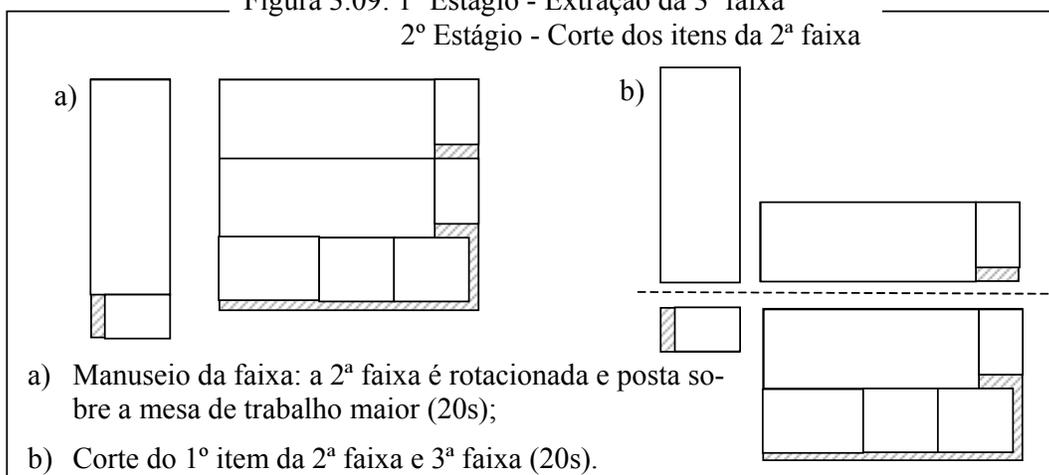


Figura 3.10: 1º Estágio - Extração da 4ª faixa  
2º Estágio - Corte dos itens da 3ª faixa

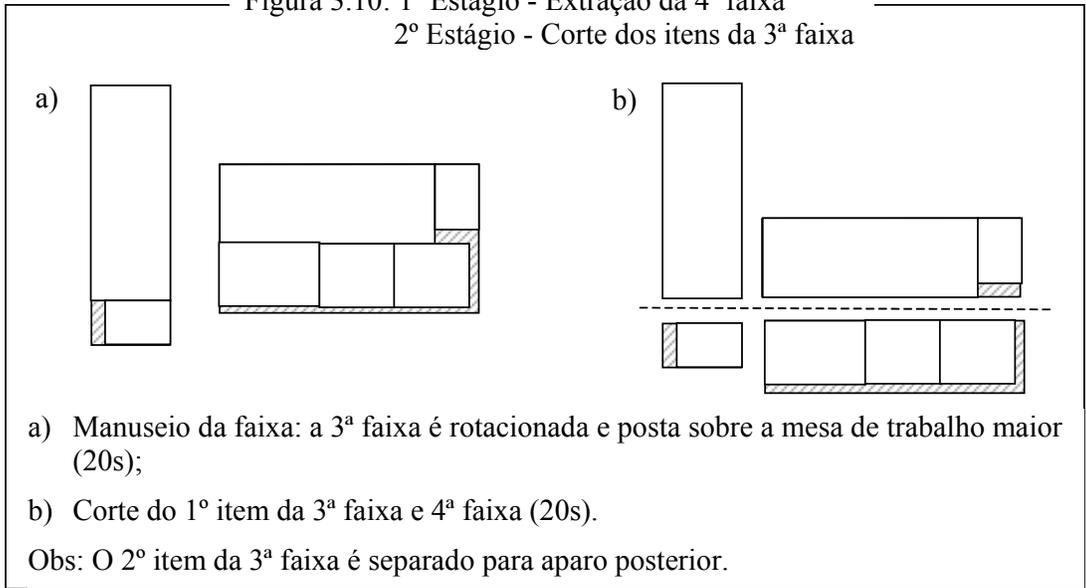


Figura 3.11: 1º Estágio - Extração da 5ª faixa (aparo)  
2º Estágio - Corte dos itens da 4ª faixa

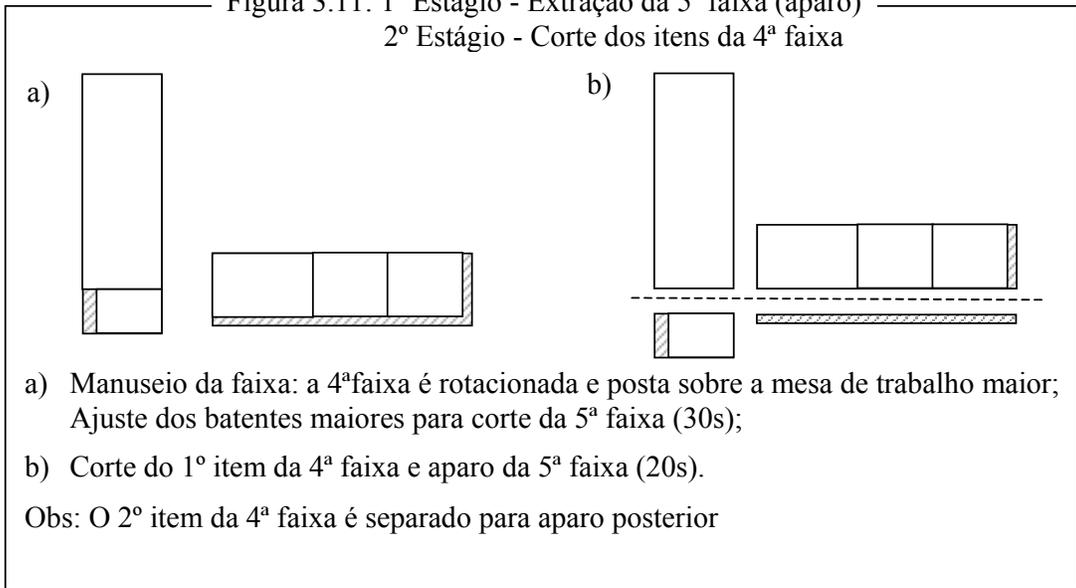


Figura 3.12: 2º Estágio - Corte dos itens da 5ª faixa

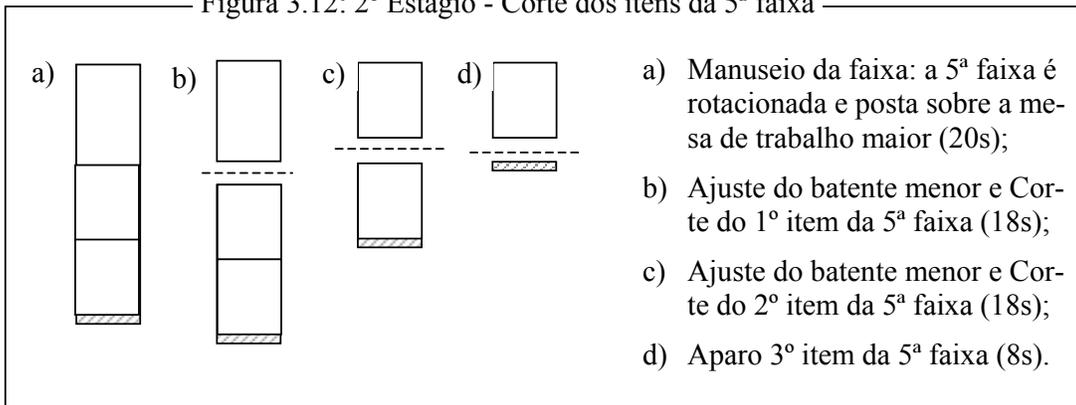
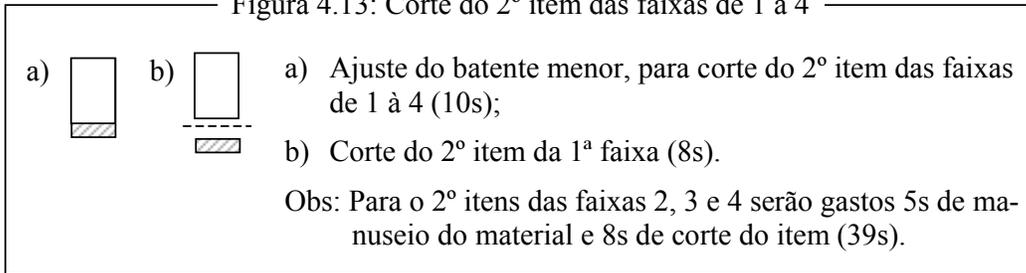


Figura 4.13: Corte do 2º item das faixas de 1 à 4



O processo exibido nas Figuras 3.07b à 3.13 irá consumir entorno de 5m 26s para o corte completo do objeto, de acordo com o padrão de corte B (Figura 3.07a). Com isso podemos ver que o padrão de corte A terá um tempo de corte 40% superior ao tempo do padrão de corte B. Essa diferença deve-se, basicamente, à necessidade de rotação do objeto. O que é inevitável em padrões com mais de dois estágios. Isso justifica a preferência da indústria por padrões de corte 2-estágios.

É fácil ver, ainda, que se tivermos muitas faixas de larguras distintas ou uma grande diversidade de itens, o tempo de ajuste dos batentes irá aumentar o tempo de corte do padrão.

### 3.7 Análise dos padrões de corte da indústria

Para avaliar a qualidade de um padrão, a indústria analisa dois aspectos importantes: o índice de perda do material e o tempo de corte. No primeiro, busca-se trabalhar com padrões de corte que apresentem os menores índices de perda possíveis. No segundo, a preferência é por padrões de corte que necessitem de poucos manuseios e ajustes.

De uma forma intuitiva, somos levados a escolher padrões de corte que busquem atender aos dois aspectos. No entanto, em geral esta escolha não é tão simples, em diversas ocasiões precisamos sacrificar um aspecto em função do outro. Neste caso, é preciso levar em conta: o custo da matéria-prima, a carteira de pedidos, os prazos de entrega, a capacidade de produção da fábrica, o número de funcionários, etc.

De uma forma geral, as pequenas indústrias trabalham com um conjunto de padrões de cortes fixo para a produção de um lote de cada produto, de forma que a quantidade a ser produzida é sempre um múltiplo do lote do produto, observados a demanda e o estoque do produto.

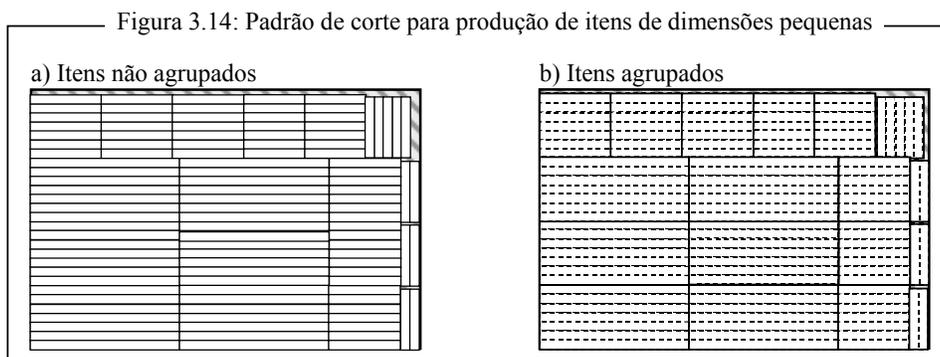
Os padrões adotados pela indústria, via de regra, são padrões de corte guilhotinados em 2 estágios, tais que as faixas possuem, em geral, duas larguras distintas. A

quantidade de itens diferentes por faixa também é limitada (2 itens). Esses limites se justificam quando olhamos para os tempos de ajuste dos batentes, exposto na seção 3.6. Se considerarmos apenas os ajustes dos batentes maiores, o tempo de ajuste deve ficar entre 40 e 50s, podendo ser feito o ajuste para o corte de até duas faixas de larguras distintas. Mais de duas larguras distintas exigirão mais de uma operação de ajuste. O mesmo ocorre com o número de itens distintos por faixa, onde o tempo gasto para o ajuste dos batentes menores, para até dois itens, é de aproximadamente 10s. Mais de dois itens exigirão operações adicionais de ajuste.

### Agrupamento de itens

Como mencionado, a seccionadora é a primeira máquina de corte e sobre ela recai uma grande carga de trabalho. A fim de minimizar esta carga de trabalho, reduzindo o risco de um gargalo na produção, a indústria usa o recurso de agrupar itens pequenos, cortando-os como um item maior. Posteriormente, estes itens agrupados são individualizados em uma máquina secundária, composta por um conjunto de serras circulares, com uma menor demanda de trabalho.

Se observarmos o padrão de corte da Figura 3.14a, veremos um padrão de corte para a produção de cinco itens com largura de 60mm. O corte do objeto, de acordo com este padrão exige um grande número de ajustes na seccionadora, o que eleva muito o tempo de corte do padrão. Por outro lado, o padrão de corte mostrado na Figura 3.14b agiliza o trabalho da seccionadora, pois o corte dos itens menores (linhas pontilhadas) é transferido para uma máquina secundária.



### 3.8 Composição do preço do produto

A matéria-prima principal utilizada para a confecção dos produtos na Empresa visitada é o painel de MDF, em diversas espessuras. Um produto, em geral, é composto por itens de diversas espessuras. Assim, os padrões de corte utilizados agrupam itens de

um mesmo produto e de mesma espessura. O painel MDF é comprado por  $m^3$ , no entanto, seu preço de tabela é dado em  $m^2$  e varia conforme a espessura (Tabela 3.02)

Tabela 3.02: Tabela de Preços do MDF

Espessura	Valor (R\$)
3mm	3,50
9mm	7,30
12mm	9,05
15mm	10,30
18mm	14,00
20mm	15,15
25mm	21,30

Fonte: Empresa Luapa, nov/2005

Além dos painéis de MDF, diversos outros itens compõem o conjunto de matérias-primas utilizadas para a confecção de um produto (Tabela 3.03).

Tabela 3.03: Tabela de Composição do custo da matéria-prima de 3 produtos

	Armário 4 Portas		Armário 5 Portas		Comoda	
	Marfim	Mogno	Marfim	Mogno	Marfim	Mogno
Madeiras	2%	2%	2%	2%	0%	0%
Painéis de MDF	52%	54%	53%	55%	49%	51%
Pinturas	19%	15%	20%	17%	18%	15%
Acessórios	14%	15%	11%	12%	20%	21%
Ferragens	3%	3%	3%	3%	2%	2%
Mat.Consumo	5%	6%	6%	6%	5%	5%
Embalagem	5%	5%	5%	5%	6%	6%

De acordo com a gerencia de produção da Luapa, para estabelecer o preço de custo do produto, é necessário levar em conta as despesas fixas da indústria, que são rateadas por cada unidade produzida. Assim, a fórmula de cálculo do preço de custo é:

$$PC = MP + DF_{unit}$$

onde:  $PC$  é o preço de custo;

$MP$  é o custo da matéria-prima;

$DF_{unit}$  é o valor das despesas fixas, rateado por unidade produzida.

No cálculo do preço de venda do produto é considerado o preço de custo acrescido da margem de lucro e das despesas variáveis que, tal como as despesas fixas, são rateadas pelas unidades produzidas. Sendo a fórmula de cálculo do preço de venda:

$$PV = \frac{PC}{1 - (ML + DV_{unit})}$$

onde:  $PV$  é o preço de venda;

$PC$  é o preço de custo;

$ML$  é a margem de lucro da indústria;

$DV_{unit}$  é o valor das despesas variáveis, rateadas por unidade produzida.

## Sobras

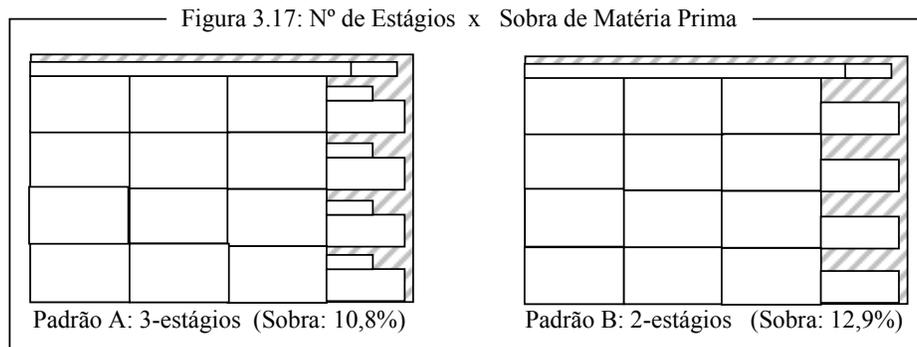
As sobras dos painéis de MDF de 15, 20 e 25mm são facilmente aproveitadas na confecção de pinázios (itens que compõem a estrutura dos produtos), o que reduz significativamente a perda dessas matérias primas. Já as sobras dos painéis de MDF de 3, 9 e 12mm são um fator de preocupação para o gerente de produção, pois estas dificilmente terão como ser reaproveitadas. A Empresa estima que a perda, por sobra, de matéria prima esteja entre 6% e 9% do volume de painéis cortados.

### 3.9 Conclusão

Durante a fase de pesquisa, nos deparamos com uma certa dificuldade em obter números atualizados, referentes ao panorama da indústria moveleira no Brasil (seção 3.2). Mesmo os trabalhos acadêmicos mais recentes (por exemplo, Azevedo (2003) e Justina (2004)), utilizam dados publicados por Gorini (1998) e do censo do IBGE de 2000. Uma outra dificuldade encontrada foi na qualidade das informações fornecidas por entidades do setor consultadas. Algumas das informações fornecidas apresentaram inconsistências que inviabilizaram o seu aproveitamento neste trabalho. Ainda assim, foi possível estabelecer critérios de avaliação da qualidade de um padrão de corte. Para isso, consideramos a sobra de matéria prima e o tempo de corte do objeto.

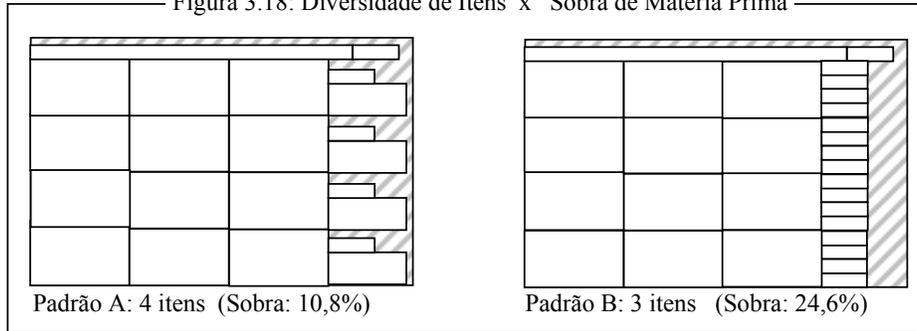
- A sobra de matéria prima

Se o objetivo é minimizar a sobra de matéria prima, podemos usar padrões em 3 ou mais estágios. Pois, quanto maior o número de estágios admitidos, maior será a possibilidade de “encaixar” um item em uma área de sobra (Figura 3.17).



Além disso, quanto maior a diversidade de itens, maior a possibilidade de encontrar uma combinação de itens que melhor aproveite o objeto (Figura 3.18).

Figura 3.18: Diversidade de Itens x Sobra de Matéria Prima



- O tempo de corte do padrão

Se o objetivo é minimizar o tempo de produção, o aumento do número de estágios acarreta uma perda na velocidade de produção, conforme visto na seção 3.6. Procuramos então, na medida do possível, trabalhar com padrões 2-estágios exatos. Um outro fator que reduz a velocidade de produção é um elevado número de itens distintos, pois uma alta diversidade de itens implica num aumento do tempo gasto com ajustes dos batentes. Portanto, quanto menor a diversidade de itens melhor será o tempo de corte. Vale observar que, na prática, a Indústria não executa os cortes por estágio, ou seja, primeiro os cortes do 1º estágio, em seguida os do 2º estágio e assim por diante. Ela executa simultaneamente cortes de estágios diferentes, o que permite um ganho de tempo no processo de corte.



Neste capítulo exibiremos os resultados dos testes realizados com o Modelo 5 (seção 2.2) para a construção de padrões tabuleiro e um estudo comparativo entre as Heurísticas 1 e 2, apresentadas na seção 2.4, o sistema CorteBi (Perin e Rangel, 1989; Rangel, 1989) e os procedimentos adotados pela Indústria estudada. No desenvolvimento das heurísticas e nos testes realizados, foi utilizado um equipamento com processador AMD-Athlon 2200, com 256 de RAM e o software XPress<sup>MP</sup> (Dash Optimization, 2004) sob a plataforma Windows 98. O Xpress<sup>MP</sup> tem agregado a si uma linguagem de modelagem, o Mosel, e um “*solver*”, o Xpress-Optimizer, para resolver problemas de programação linear, inteira e quadrática utilizando um algoritmo *branch-and-cut*, na solução de problemas inteiros mistos. O Mosel permite uma interação entre o modelo e o “*solver*”, o que possibilita a implementação de um algoritmo de geração de colunas (Heipcke, 2004; Colombani e Heipcke, 2004), conforme pode ser visto nos Apêndices A2 e A3. O uso do Xpress<sup>MP</sup>, como ferramenta de desenvolvimento, nos deu a agilidade necessária para as alterações e testes dos modelos. Os exemplares utilizados neste estudo foram baseados nos dados gentilmente cedidos pela Fábrica de Móveis Luapa.

#### 4.1 Estudo computacional da geração de padrões tabuleiro usando o Modelo 5

Na avaliação do Modelo 5, consideramos um objeto com dimensões ( $L \times W$ ) 2750 x 1830 e cinco conjuntos de dados (Quadro 4.01), o conjunto Cj-01 com seis itens, o conjunto Cj-02 com 9 itens, e os demais (Cj-03, Cj-04 e Cj-05) com dez itens cada. Estes conjuntos formarão uma base para a análise de desempenho do Modelo 5 em diversas situações, envolvendo a variação do número de itens e a diversidade de dimensões dos itens.

Quadro 4.01: Conjuntos para análise de desempenho do Modelo 5

Cj-01		Cj-02		Cj-03		Cj-04		Cj-05	
Item	$\ell_i \times w_i$								
01	500 x 430	01	452 x 1049	01	453 x 647	01	453 x 647	01	454 x 647
02	452 x 535	02	499 x 452	02	454 x 700	02	450 x 710	02	454 x 710
03	535 x 500	03	452 x 429	03	430 x 454	03	404 x 454	03	454 x 530
04	452 x 430	04	1050 x 535	04	454 x 535	04	554 x 530	04	500 x 483
05	452 x 500	05	500 x 535	05	315 x 483	05	460 x 483	05	430 x 510
06	535 x 430	06	535 x 430	06	452 x 500	06	422 x 510	06	500 x 420
		07	212 x 700	07	452 x 430	07	400 x 430	07	500 x 530
		08	430 x 60	08	535 x 500	08	585 x 500	08	430 x 500
		09	500 x 60	09	535 x 430	09	600 x 415	09	430 x 415
				10	315 x 700	10	515 x 700	10	430 x 700

Obs: Na aplicação do Modelo 5 foi desconsiderada a restrição 2.17, referente a demanda dos itens

Começamos com a aplicação direta do Modelo 5 em cada um dos conjuntos do Quadro 4.01. Para cada conjunto foram realizados três testes, um considerando permitida a rotação dos itens e outros dois com orientação fixa dos itens (inicialmente fixamos a orientação  $\ell_i \times w_i$  e, em seguida, a orientação  $w_i \times \ell_i$ ). Na Tabela 4.01 estão registrados dados relativos a orientação dos itens (Orientação dos Itens), número de restrições, variáveis e nós gerados pelo modelo (respectivamente, N° Restr., N° Var. e N° Nós), a diferença percentual verificada entre a melhor solução inteira encontrada e a solução da relaxação linear (GAP), a sobra de material dada pela solução inteira encontrada e o tempo de processamento, limitado à 3600s.

Analisando os resultados da Tabela 4.01, considerando a rotação dos itens, podemos perceber que a aplicação do Modelo 5 só apresentou resultados satisfatórios para o conjunto Cj-01. Para os conjuntos Cj-03 e Cj-05 os tempos de processamentos (entorno de 5min) é comprometedora, se levarmos em conta que só foi gerado um padrão, porém os índices de sobra de matéria prima são muito bons (menores que 1%). Para os conjuntos Cj-02 e Cj-04 os resultados foram muito ruins, tanto os tempos de processamento quanto os índices de sobra.

Tabela 4.01: Resultados dos testes do Modelo 5 aplicado aos conjuntos do Quadro 4.01

	Orientação de Itens	Nº Restr.	Nº Var.	GAP	Nº Nós	Tempo (seg)	Sobra Material
Cj-01	Rotação Permitida	199	432	0,0%	2402	5,8	0,9 %
	Fixa ( $\ell_i \times w_i$ )	199	222	0,0%	174	2,5	0,9 %
	Fixa ( $w_i \times \ell_i$ )	199	222	0,0%	296	2,1	2,2 %
Cj-02	Rotação Permitida	6526	23041	98,9 %	23929	3600	19,3%
	Fixa ( $\ell_i \times w_i$ )	2086	3617	82,5 %	285343	3600	3,5 %
	Fixa ( $w_i \times \ell_i$ )	1969	3435	21,4 %	194449	3600	6,3 %
Cj-03	Rotação Permitida	246	1095	0,0%	65474	305	0,25%
	Fixa ( $\ell_i \times w_i$ )	236	433	0,0%	545	5,1	1,45%
	Fixa ( $w_i \times \ell_i$ )	253	464	0,0%	988	6,3	2,72%
Cj-04	Rotação Permitida	199	712	18,1%	609270	3600	6,30%
	Fixa ( $\ell_i \times w_i$ )	199	362	0,0%	9461	39,2	6,27%
	Fixa ( $w_i \times \ell_i$ )	199	362	35,8%	880076	3600	15,37%
Cj-05	Rotação Permitida	160	712	0,0 %	83290	291	0,6%
	Fixa ( $\ell_i \times w_i$ )	199	362	0,0 %	276	3,5	3,4 %
	Fixa ( $w_i \times \ell_i$ )	199	362	0,0 %	183	1,6	1,5 %

Nos demais resultados da tabela 4.01, observamos que quando consideramos a orientação fixa dos itens, o número de restrições e variáveis do problema é bem menor e com isso, exceto os conjuntos Cj-02 e Cj-04, os resultados nos tempos de processamento melhoram significativamente e que os índices de sobra da matéria prima ficam em um patamar aceitável pela Indústria (inferiores a 6%). No caso dos conjuntos Cj-02 e Cj-04 nem mesmo a orientação fixa dos itens foi capaz de apresentar uma boa resposta. Em geral, o GAP verificado nestes conjuntos se manteve alto e os índices de sobra de material foram superiores à 6%.

Observando atentamente, percebemos que o conjuntos Cj-02 e Cj-04 possuem os mais altos índices de diversidade nas dimensões dos itens (número de dimensões distintas por número total de dimensões). Na tabela 4.02, apresentamos uma análise da diversidade de dimensões dos conjuntos do Quadro 4.01, considerando-os como de baixa diversidade (até 50%) e alta diversidade (acima de 50%).

Tabela 4.02: Diversidade de dimensões dos conjuntos

	Nº de Itens	Dim. Distintas / Total Dim.	Diversidade (%)
Cj-01	6	4 / 12	33,3%
Cj-02	9	11 / 18	61,1%
Cj-03	10	10 / 20	50%
Cj-04	10	20 / 20	100%
Cj-05	10	12 / 20	60%

Buscaremos, ao longo da próxima secção, analisar o efeito da variação da diversidade de dimensões dos itens e da quantidade de itens de um conjunto no processamento do Modelo 5.

#### 4.1.1 Análise da quantidade de itens e diversidade de dimensões no desempenho do Modelo 5

Tomaremos por base os conjuntos Cj-03, Cj-04 e Cj-05, que possuem a mesma quantidade de itens (10), e construiremos quatro subconjuntos para cada um, com diferentes quantidades de itens.

#### Conjuntos com baixa diversidade nas dimensões

O Quadro 4.02, abaixo, apresenta os subconjuntos criados a partir do conjunto Cj-03, com baixa diversidade nas dimensões dos itens.

Quadro 402: Conjuntos baixa diversidade nas dimensões dos itens

Cj-03		Cj-03-A		Cj-03-B		Cj-03-C		Cj-03-D	
Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$
01	453 x 647	01	454 x 700	01	430x 454	01	430x 454	01	430 x 454
02	454 x 700	02	430 x 454	02	454x 535	02	452x 500	02	430 x 535
03	430 x 454	03	454 x 535	03	452x 500	03	535x 500	03	430 x 535
04	454 x 535	04	315 x 483	04	452x 430	04	535x 430		
05	315 x 483	05	452 x 500	05	535x 500				
06	452 x 500	06	535 x 500	06	535x 430				
07	452 x 430	07	535 x 430						
08	535 x 500	08	315 x 700						
09	535 x 430								
10	315 x 700								

Na Tabela 4.03 estão registrados os resultados obtidos com o processamento dos subconjuntos Cj-03, utilizando o Modelo 5. Para cada conjunto foram realizados três testes, um considerando permitida a rotação dos itens e outros dois com orientação fixa dos itens (inicialmente fixamos a orientação  $\ell_i \times w_i$  e, em seguida, a orientação  $w_i \times \ell_i$ ).

Tabela 4.03: Resultados dos testes do Modelo 5 aplicado aos subconjuntos de Cj-03

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-03-A	Rotação Permitida	246	879	62	0,62%
	Fixa ( $\ell_i \times w_i$ )	236	433	4,6	3,16%
	Fixa ( $w_i \times \ell_i$ )	253	374	6,9	3,59%
Cj-03-B	Rotação Permitida	199	432	9	0,91%
	Fixa ( $\ell_i \times w_i$ )	199	222	1,4	0,91%
	Fixa ( $w_i \times \ell_i$ )	199	222	2,1	2,17%
Cj-03-C	Rotação Permitida	164	292	3	1,86%
	Fixa ( $\ell_i \times w_i$ )	199	152	1,4	5,55%
	Fixa ( $w_i \times \ell_i$ )	199	152	1,6	3,64%
Cj-03-D	Rotação Permitida	164	152	2	5,55%
	Fixa ( $\ell_i \times w_i$ )	199	82	0,5	5,55%
	Fixa ( $w_i \times \ell_i$ )	199	82	0,8	5,55%

Nos testes realizados com os subconjuntos de Cj-03, considerando permitida a rotação dos itens, o Modelo 5 mostrou-se bem eficiente para os conjuntos com até seis itens, tanto no que tange o tempo de processamento quanto no percentual de sobra de material. Para o conjunto Cj-03-A, com oito itens, o tempo de processamento já não foi tão bom, porém ainda aceitável. Quando realizamos o teste fixando a orientação dos itens, verificamos tempos excelentes. Contudo, os percentuais de sobra tiveram uma sensível piora e, neste caso, os conjuntos menores foram os mais afetados, apesar de terem apresentado perdas dentro do esperado pela Indústria.

### Conjuntos com alta diversidade nas dimensões

O Quadro 4.03 apresenta os subconjuntos criados a partir do conjunto Cj-04, com alta diversidade nas dimensões dos itens.

Quadro 4.03: Conjuntos alta diversidade nas dimensões dos itens

Cj-04		Cj-04-A		Cj-04-B		Cj-04-C		Cj-04-D	
Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$
01	453 x 647	01	453 x 647	01	453x 647	01	453x 647	01	400 x 430
02	450 x 710	02	450 x 710	02	450x 710	02	554x 530	02	585 x 500
03	404 x 454	03	554 x 530	03	554x 530	03	400x 430		
04	554 x 530	04	460 x 483	04	430x 400	04	600x 415		
05	460 x 483	05	400 x 430	05	585 x 500				
06	422 x 510	06	585 x 500	06	600x 415				
07	400 x 430	07	600 x 415						
08	585 x 500	08	515 x 700						
09	600 x 415								
10	515 x 700								

Na Tabela 4.04 estão registrados os resultados obtidos nos testes com os subconjuntos de Cj-04 (Quadro 4.03), utilizando o Modelo 5. Note que a solução ótima foi encontrada na maioria dos casos. O GAP de integralidade, após uma hora de processo, para o conjunto Cj-04-A foi de 14,2% quando permitida a rotação dos itens e 15,7% com a orientação fixa ( $w_i \times \ell_i$ ).

Os dados da Tabela 4.04 mostram que os resultados para o processamento de conjuntos com uma grande diversidade nas dimensões dos itens, são críticos para o subconjuntos com 8 itens (Cj-04-A). Para este subconjunto nem mesmo o recurso de fixar a orientação dos itens nos garante um bom desempenho do modelo. Os conjuntos com 6 e 4 itens (Cj-04-B e Cj-04C) apresentaram um tempo de processamento ruim, quando considerada a rotação dos itens. Porém, quando processados com a orientação fixa dos itens apresentaram resultados satisfatórios. O conjunto com apenas 2 itens, apesar de apresentar um baixo tempo de processamento, apresentou um índice de sobra de materi-

al muito alto. Vale ressaltar que as coincidências nos índices de sobra de material, verificada na Tabela 4.04, entre os processamentos com rotação permitida de itens e com orientação fixa, se deve ao fato de não existirem, nos subconjuntos de Cj-04, ao menos dois itens com uma das dimensões iguais. Com isso o padrão tabuleiro exato será gerado com apenas um item, ou seja, será um padrão homogêneo (seção 2.2)

Tabela 4.04: Resultados dos testes do Modelo 5 aplicado aos subconjuntos de Cj-04

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-04-A	Rotação Permitida	199	572	3600*	5,76%
	Fixa ( $\ell_i \times w_i$ )	199	292	3600*	5,76%
	Fixa ( $w_i \times \ell_i$ )	199	292	29	16,69%
Cj-04-B	Rotação Permitida	199	432	91	5,76%
	Fixa ( $\ell_i \times w_i$ )	199	222	2,1	5,76%
	Fixa ( $w_i \times \ell_i$ )	199	222	3,3	16,69%
Cj-04-C	Rotação Permitida	199	292	67	5,76%
	Fixa ( $\ell_i \times w_i$ )	199	152	0,9	5,76%
	Fixa ( $w_i \times \ell_i$ )	199	152	1,4	16,69%
Cj-04-D	Rotação Permitida	199	152	1	11,84%
	Fixa ( $\ell_i \times w_i$ )	199	82	0,5	11,84%
	Fixa ( $w_i \times \ell_i$ )	199	82	0,5	16,69%

\* Melhor Solução encontrada, após uma hora de processo

### Conjuntos com alta diversidade em uma das dimensões e baixa na outra

Encontrar um conjunto com uma baixa diversidade em ambas as dimensões dos itens não é muito comum, na prática. Buscaremos então, avaliar se a baixa diversidade em apenas uma das dimensões dos itens do conjunto viabiliza a aplicação do Modelo 5. Para isso, foram criados a partir do conjunto Cj-05, com baixa diversidade em uma das dimensões dos itens, três subconjuntos com diferentes quantidades de itens (Quadro 4.04).

Quadro 4.04: Conjuntos alta diversidade em uma das dimensões dos itens e baixa na outra

Cj-05		Cj-05-A		Cj-05-B		Cj-05-C	
Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$	Item	$\ell_i \times w_i$
01	454 x 647	01	454 x 647	01	454 x 647	01	430 x 510
02	454 x 710	02	454 x 710	02	454 x 710	02	500 x 420
03	454 x 530	03	454 x 530	03	430 x 510	03	500 x 530
04	500 x 483	04	500 x 483	04	500 x 420	04	430 x 500
05	430 x 510	05	430 x 510	05	500 x 530		
06	500 x 420	06	500 x 420	06	430 x 500		
07	500 x 530	07	500 x 530				
08	430 x 500	08	430 x 500				
09	430 x 415						
10	430 x 700						

Os teste foram realizados considerando-se apenas as orientações fixas  $\ell_i \times w_i$  e  $w_i \times \ell_i$ . Pois se considerarmos a rotação dos itens cairemos no caso anterior, (alta diversidade nas dimensões. Os resultados obtidos, apresentados na Tabela 4.05, nos mos-

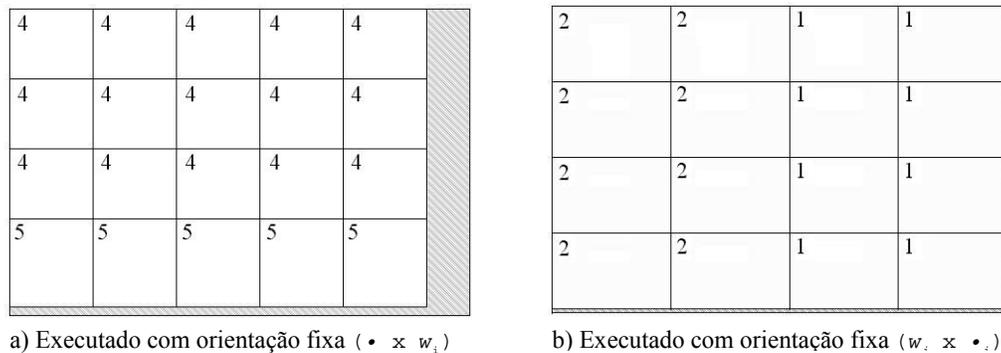
tram bons tempos de processamento. No entanto, verificamos novamente que quanto menor a quantidade de itens, maiores são os índices de sobra de material.

Tabela 4.05: Resultados dos testes do Modelo 5 aplicado aos subconjuntos de Cj-05

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-05-A	Fixa ( $\ell_i \times w_i$ )	199	292	2,1	3,4 %
	Fixa ( $w_i \times \ell_i$ )	199	292	1,8	1,5 %
Cj-05-B	Fixa ( $\ell_i \times w_i$ )	199	222	2,1	10,4 %
	Fixa ( $w_i \times \ell_i$ )	199	222	0,8	1,5 %
Cj-05-C	Fixa ( $\ell_i \times w_i$ )	199	152	1,4	10,4 %
	Fixa ( $w_i \times \ell_i$ )	199	152	1,3	12,2 %

Verificamos os padrões gerados (ver exemplos na Figura 4.01) e constatamos que quanto maior a diversidade de dimensões na largura ( $w_i$ ), maior será a tendência de formação de faixas homogêneas e maior será a possibilidade de combinação de faixas diferentes no padrão. Quando a maior diversidade de dimensões ocorrer no comprimento ( $\ell_i$ ), maior será a possibilidade de formação de faixas heterogêneas, com uma forte tendência de compor um padrão com apenas uma faixa.

Figura 4.01: Padrões gerados pelo Modelo 5 para conjunto Cj-05-B



#### 4.1.2 Análise do impacto de itens de dimensões pequenas no Modelo 5

Nesta etapa buscaremos estabelecer uma relação entre o tempo de processo do Modelo 5 e as dimensões dos itens envolvidos. Começaremos classificando a dimensão do item, com relação às dimensões do objeto, em grande, média, pequena e muito pequena, conforme a Tabela 4.06. Tomaremos por base, para nossa análise, o conjunto Cj-01 (Quadro 4.01). A escolha do conjunto Cj-01 deve-se ao fato dele ser um conjunto de itens de dimensões médias, conforme a classificação pela Tabela 4.06, com uma baixa diversidade nas dimensões e com 6 itens, que apresentou bons resultados nos tempos de processamento e nos índices de sobra de material (Tabela 4.01). Neste conjunto, introduziremos inicialmente itens com uma dimensão grande, pequena e muito pequena para avaliar o quanto esses itens afetam a execução do Modelo 5.

Tabela 4.06: Classificação das dimensões  $w_i$  e  $\ell_i$  em função de  $W$  e  $L$ , respectivamente

Em relação à Largura		Em relação ao Comprimento	
$\left\lfloor \frac{W}{w_i} \right\rfloor \leq 2$	$w_i$ é uma dimensão grande (grd)	$\left\lfloor \frac{L}{\ell_i} \right\rfloor \leq 2$	$\ell_i$ é uma dimensão grande (grd)
$3 \leq \left\lfloor \frac{W}{w_i} \right\rfloor \leq 7$	$w_i$ é uma dimensão média (med)	$3 \leq \left\lfloor \frac{L}{\ell_i} \right\rfloor \leq 7$	$\ell_i$ é uma dimensão média (med)
$8 \leq \left\lfloor \frac{W}{w_i} \right\rfloor \leq 10$	$w_i$ é uma dimensão pequena (peq)	$8 \leq \left\lfloor \frac{L}{\ell_i} \right\rfloor \leq 10$	$\ell_i$ é uma dimensão pequena (peq)
$10 < \left\lfloor \frac{W}{w_i} \right\rfloor$	$w_i$ é uma dimensão muito pequena (mpq)	$10 < \left\lfloor \frac{L}{\ell_i} \right\rfloor$	$\ell_i$ é uma dimensão muito pequena (mpq)

### Conjunto com itens de dimensões grandes

A partir do conjunto Cj-01 construiremos três conjuntos de dados (Quadro 4.05): o primeiro com a substituição do item 1 do conjunto por um outro com uma dimensão grande (Cj-01-A1); no segundo, substituiremos dois itens do conjunto por itens com uma mesma dimensão grande (Cj-01-A2); e no terceiro, substituiremos dois itens do conjunto por itens com dimensões grandes diferentes (Cj-01-A3).

Quadro 4.05: Conjuntos com itens de dimensões grandes

Cj-01-A1		Cj-01-A2		Cj-01-A3	
	$\ell_i$ x $w_i$		$\ell_i$ x $w_i$		$\ell_i$ x $w_i$
Item 1	1500 x 430	Item 1	1500 x 430	Item 1	1500 x 430
Item 2	452 x 535	Item 2	1500 x 535	Item 2	1000 x 535
Item 3	535 x 500	Item 3	535 x 500	Item 3	535 x 500
Item 4	452 x 430	Item 4	452 x 430	Item 4	452 x 430
Item 5	452 x 500	Item 5	452 x 500	Item 5	452 x 500
Item 6	535 x 430	Item 6	535 x 430	Item 6	535 x 430

Observando os resultados do processamento desses conjuntos (Tabela 4.07), podemos notar que os itens com dimensões grandes não constituem um problema para o tempo de processamento.

Tabela 4.07: Resultados dos testes do Modelo 5 aplicado aos subconjuntos de Cj-05

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-05-A1	Rotação Permitida	199	432	6,9	0,9%
Cj-05-A2	Rotação Permitida	199	432	5,6	0,9%
Cj-05-A3	Rotação Permitida	199	432	5,0	0,9%

### Conjunto com itens de dimensões pequenas

A partir do conjunto Cj-01 construiremos cinco conjuntos de dados (Quadro 4.06): o primeiro com a substituição de um item do conjunto por um outro com uma dimensão pequena (Cj-01-B1); no segundo, substituiremos dois itens do conjunto por itens com uma mesma dimensão pequena (Cj-01-B2); no terceiro, substituiremos dois

itens do conjunto por itens com dimensões pequenas diferentes (Cj-01-B3); no quarto, substituiremos três itens do conjunto por itens com uma mesma dimensão pequena (Cj-01-B4); e no quinto, substituiremos três itens do conjunto por itens com dimensões pequenas diferentes (Cj-01-B5).

Quadro 4.06: Conjuntos com itens de dimensões pequenas

Cj-01-B1			Cj-01-B2			Cj-01-B3		
	$l_i$	$w_i$		$l_i$	$w_i$		$l_i$	$w_i$
Item 1	260	430	Item 1	260	430	Item 1	260	430
Item 2	452	535	Item 2	260	535	Item 2	300	535
Item 3	535	500	Item 3	535	500	Item 3	535	500
Item 4	452	430	Item 4	452	430	Item 4	452	430
Item 5	452	500	Item 5	452	500	Item 5	452	500
Item 6	535	430	Item 6	535	430	Item 6	535	430

Cj-01-B4			Cj-01-B5		
	$l_i$	$w_i$		$l_i$	$w_i$
Item 1	260	430	Item 1	260	430
Item 2	260	535	Item 2	300	535
Item 3	260	500	Item 3	250	500
Item 4	452	430	Item 4	452	430
Item 5	452	500	Item 5	452	500
Item 6	535	430	Item 6	535	430

Observando os resultados do processamento desses conjuntos (Tabela 4.08), podemos notar que os itens com dimensões pequenas afetam de forma significativa o tempo de processamento, principalmente o conjunto Cj-01-B5, que contém três itens com dimensões pequenas e distintas. Nos testes realizados, a execução do Modelo 5 com orientação fixa dos itens apresentou, em geral, bons resultados, configurando-se um forma alternativa para o processamento desses conjuntos.

Tabela 4.08: Resultados dos testes do Modelo 5 aplicado aos conjuntos de Cj-01-B

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-01-B1	Rotação Permitida	421	942	94,9	1,5 %
	Fixa ( $l_i \times w_i$ )	273	308	1,4	1,4 %
	Fixa ( $w_i \times l_i$ )	307	346	3,9	2,7 %
Cj-01-B2	Rotação Permitida	421	942	92,3	1,5 %
	Fixa ( $l_i \times w_i$ )	273	308	8,9	1,4 %
	Fixa ( $w_i \times l_i$ )	307	346	11,2	2,7 %
Cj-01-B3	Rotação Permitida	421	942	72,4	1,5 %
	Fixa ( $l_i \times w_i$ )	273	308	2,7	1,4 %
	Fixa ( $w_i \times l_i$ )	307	346	3,4	12,8 %
Cj-01-B4	Rotação Permitida	421	942	162,6	1,5 %
	Fixa ( $l_i \times w_i$ )	273	308	6,5	1,4 %
	Fixa ( $w_i \times l_i$ )	307	346	10,8	2,7 %
Cj-01-B5	Rotação Permitida	478	1075	252,2	1,5 %
	Fixa ( $l_i \times w_i$ )	310	351	2,9	1,4 %
	Fixa ( $w_i \times l_i$ )	307	346	6,7	2,7 %

### Conjunto com itens de dimensões muito pequenas

Por último, a partir do conjunto Cj-01 construiremos seis conjuntos de dados (Quadro 4.07), de forma similar aos conjuntos do Quadro 4.06. Este seis conjuntos foram construídos de forma que: nos dois primeiros substituiremos um item do conjunto por um outro com uma dimensão muito pequena (Cj-01-C1), visando analisar a influência do tamanho da dimensão do item no modelo (Cj-01-C2); no terceiro, substituiremos dois itens do conjunto por itens com uma mesma dimensão muito pequena (Cj-01-C3); no quarto, substituiremos dois itens do conjunto por itens com dimensões muito pequenas diferentes (Cj-01-C4); no quinto, substituiremos três itens do conjunto por itens com uma mesma dimensão muito pequena (Cj-01-C5); e no sexto, substituiremos três itens do conjunto por itens com dimensões muito pequenas diferentes (Cj-01-C6).

Quadro 4.07: Conjuntos com itens de dimensões muito pequenas

Cj-01-C1			Cj-01-C2			Cj-01-C3		
	$l_i$	$w_i$		$l_i$	$w_i$		$l_i$	$w_i$
Item 1	100	430	Item 1	50	430	Item 1	100	430
Item 2	452	535	Item 2	452	535	Item 2	100	535
Item 3	535	500	Item 3	535	500	Item 3	535	500
Item 4	452	430	Item 4	452	430	Item 4	452	430
Item 5	452	500	Item 5	452	500	Item 5	452	500
Item 6	535	430	Item 6	535	430	Item 6	535	430

Cj-01-C4			Cj-01-C5			Cj-01-C6		
	$l_i$	$w_i$		$l_i$	$w_i$		$l_i$	$w_i$
Item 1	100	430	Item 1	100	430	Item 1	100	430
Item 2	50	535	Item 2	100	535	Item 2	50	535
Item 3	535	500	Item 3	100	500	Item 3	150	500
Item 4	452	430	Item 4	452	430	Item 4	452	430
Item 5	452	500	Item 5	452	500	Item 5	452	500
Item 6	535	430	Item 6	535	430	Item 6	535	430

Os resultados do processamento destes conjuntos (Tabela 4.09), mostram que os conjuntos compostos por itens com dimensões muito pequenas, em geral, são críticos para o Modelo 5. Se observarmos, veremos que nos conjuntos Cj-01-A's e Cj-01-B's o número de variáveis (Nº Var.) é ordem de 500 e 1000, respectivamente. Nos conjuntos Cj-05-C's estes mesmos número saltam para 5800 quando a menor dimensão é 100mm, e para 21300 quando a menor dimensão é 50mm. Mostrando que quanto menor a dimensão do item maior deverá ser o número de variáveis criadas pelo Modelo 5, ou seja, mais pesado será o processamento. Podemos observar também que o número de itens com dimensões muito pequenas agrava este problema.

Tabela 4.09: Resultados dos testes do Modelo 5 aplicado aos conjuntos Cj-01-C's

	Orientação de Itens	Nº Restr.	Nº Var.	GAP	Nº Nós	Tempo (seg)	Sobra Material
Cj-01-C1	Rotação Permitida	2520	5877	70,6 %	193020	3600	4,8 %
	Fixa ( $\ell_i \times w_i$ )	680	781	0,0 %	386	16,8	1,4 %
	Fixa ( $w_i \times \ell_i$ )	739	842	0,0 %	573	14,4	2,7 %
Cj-01-C2	Rotação Permitida	9012	21302	92,3 %	44580	3600	7,6 %
	Fixa ( $\ell_i \times w_i$ )	1272	1469	0,0 %	672	49,6	1,5 %
	Fixa ( $w_i \times \ell_i$ )	1414	1617	0,0 %	29986	576	2,8 %
Cj-01-C3	Rotação Permitida	2520	5877	81,9 %	167270	3600	8,4 %
	Fixa ( $\ell_i \times w_i$ )	680	781	0,0 %	3769	38,9	1,5 %
	Fixa ( $w_i \times \ell_i$ )	739	842	0,0 %	2961	30,0	2,7 %
Cj-01-C4	Rotação Permitida	9012	21302	80,7 %	43640	3600	11,0 %
	Fixa ( $\ell_i \times w_i$ )	1272	1469	0,0%	1219	43,3	1,5 %
	Fixa ( $w_i \times \ell_i$ )	1414	1617	0,0%	46289	506	2,8 %
Cj-01-C5	Rotação Permitida	2520	5877	81,6 %	172580	3600	7,4 %
	Fixa ( $\ell_i \times w_i$ )	680	781	0,0%	70174	288,8	1,5 %
	Fixa ( $w_i \times \ell_i$ )	739	842	0,0%	14312	101,1	2,5 %
Cj-01-C6	Rotação Permitida	9012	21302	97,4 %	8426	3600	9,6%
	Fixa ( $\ell_i \times w_i$ )	1272	1469	0,0 %	8183	146,6	1,5 %
	Fixa ( $w_i \times \ell_i$ )	1414	1617	0,0 %	39669	480	2,8 %

Nos testes só foi viável executar o Modelo 5, para os conjuntos Cj-05-C's, com a orientação fixa dos itens. Quando consideramos a rotação dos itens, o número de variáveis do problema tende a crescer muito. Pois, como visto na seção 2.2, no Modelo 5 o número de variáveis é dado em função do número faixas, possíveis, ao longo de  $L$  e de  $W$ , ou seja:

$$N^{\circ} \text{ de variáveis} = n \cdot J \cdot K$$

onde:  $n$  é o número de itens do conjunto;  
 $J = \lfloor L/\ell_{\min} \rfloor$  é a quantidade de faixas, possíveis, ao longo de  $L$ ;  
 $K = \lfloor W/w_{\min} \rfloor$  é a quantidade de faixas, possíveis, ao longo de  $W$

### Considerando o agrupamento de itens

Tomemos, novamente, o conjunto Cj-02 (Quadro 4.01). Se analisarmos as dimensões dos itens que compõem o conjunto, poderemos classificar as dimensões (Tabela 4.10), conforme a Tabela 4.06.

Tabela 4.10: Classificação das dimensões do conjunto Cj-02

	Em relação a W	Em relação a L
Grandes	3 (1050, 1049, 700)	2 (1050, 1049)
Médias	6 (535, 500, 499, 452, 430, 429)	7 (700, 535, 500, 499, 452, 430, 429)
Pequenas	1 (212)	0
Muito peq.	1 (60)	2 (212, 60)

Conforme visto, uma baixa diversidade nas dimensões do itens melhora o processamento do conjunto pelo Modelo 5. Assim, podemos reorganizar o conjunto Cj-02,

conforme a Quadro 4.08. No processo de reorganização do conjunto Cj-02, buscamos fazer com que todas as dimensões pequenas e muito pequenas coincidam com  $w_i$  (Cj-02-A1) ou com  $\ell_i$  (Cj-02-A2).

Quadro 4.08: Reorganização do conjunto Cj-02

Cj-02-A1			Cj-02-A2		
	$\ell_i$	$w_i$		$\ell_i$	$w_i$
Item 1	452	1049	Item 1	452	1049
Item 2	452	499	Item 2	452	499
Item 3	452	429	Item 3	452	429
Item 4	535	1050	Item 4	535	1050
Item 5	535	500	Item 5	535	500
Item 6	535	430	Item 6	535	430
Item 7	700	212	Item 7	212	700
Item 8	430	60	Item 8	60	430
Item 9	500	60	Item 9	60	500

Os resultados do processamento desses conjuntos, com orientação fixa dos itens, podem ser verificados na tabela 4.11. O maior problema que enfrentamos nestes conjuntos é a presença de itens com dimensões muito pequenas. No capítulo 4, vimos que como estes itens demandam muito tempo de máquina (seccionadora) para serem produzidos, a indústria opta por cortá-los de forma agrupada, transferindo para uma máquina secundária a tarefa de individualizá-los.

Tabela 4.11: Resultados dos testes do Modelo 5 aplicado aos conjuntos Cj-02-A's

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-02-A1	Fixa ( $\ell_i \times w_i$ )	1087	1863	1838	1,8 %
	Fixa ( $w_i \times \ell_i$ )	1198	2029	708	2,3 %
Cj-02-A2	Fixa ( $\ell_i \times w_i$ )	1198	2029	358	1,8 %
	Fixa ( $w_i \times \ell_i$ )	1087	1863	59,8	3,0 %

Se trabalharmos com a mesma idéia, podemos agrupar os itens de dimensões muito pequenas em itens de dimensões pequenas ou médias. Para isso podemos usar, como critério de quantos itens agrupar, a Tabela 4.05. Multiplicamos a dimensão desejada ( $d_i$ ) por um número inteiro ( $v_i$ ), não esquecendo de considerar a espessura da serra quando for o caso, de tal forma que a nova dimensão ( $d'_i$ ) seja um valor inteiro ( $d'_i = v_i \cdot (d_i + \delta) - \delta$ ), dentro de pelo menos um dos seguintes intervalos:

- $3 \leq \left\lfloor \frac{W}{d'_i} \right\rfloor \leq 7$  ou  $3 \leq \left\lfloor \frac{L}{d'_i} \right\rfloor \leq 7$  (Média)
- $8 \leq \left\lfloor \frac{W}{d'_i} \right\rfloor \leq 10$  ou  $8 \leq \left\lfloor \frac{L}{d'_i} \right\rfloor \leq 10$  (Pequena)

Feito isto, se o problema for restrito, efetuamos também um ajuste na demanda do item no problema, de modo que:

$$b'_i = \left\lceil \frac{b_i}{v_i} \right\rceil$$

onde:  $b'_i$  é a demanda ajustada;

$b_i$  é a demanda anterior

$v_i$  é o número de itens  $i$  agrupados

A partir do conjunto Cj-02, criamos novos conjuntos considerando os agrupamentos realizados, obtendo os conjuntos Cj-02-B's (Quadro 4.09).

Quadro 4.09: Conjuntos obtidos de Cj-02 pelo agrupamento de itens

Cj-02-B1			Cj-02-B2			Cj-02-B3		
	$\ell_i$	$w_i$		$\ell_i$	$w_i$		$\ell_i$	$w_i$
Item 1	452	1049	Item 1	452	1049	Item 1	452	1049
Item 2	452	499	Item 2	452	499	Item 2	452	499
Item 3	452	429	Item 3	452	429	Item 3	452	429
Item 4	535	1050	Item 4	535	1050	Item 4	535	1050
Item 5	535	500	Item 5	535	500	Item 5	535	500
Item 6	535	430	Item 6	535	430	Item 6	535	430
Item 7	428	700	Item 7	428	700	Item 7	212	700
Item 8	444	430	Item 8	252	430	Item 8	252	430
Item 9	444	500	Item 9	252	500	Item 9	252	500

Observe que, considerando um objeto de dimensões 2750 x 1830, no conjunto Cj-02-B1 nós transformamos todas as dimensões pequenas e muito pequenas em dimensões médias, no conjunto Cj-02-B2 transformamos a dimensão pequena em dimensão média e as dimensões muito pequenas em dimensões pequenas e no conjunto Cj-02-B3 transformamos apenas as dimensões muito pequenas em dimensões pequenas. Com esses agrupamentos, continuamos trabalhando com conjuntos de nove itens, porém sem a presença de itens com dimensões muito pequenas.

Na Tabela 4.12 podemos constatar que, com o agrupamento de itens, houve um considerável melhora nos tempos de processamento. Quanto aos índices de perda, apesar deles se manterem, os resultados não são conclusivos.

Tabela 4.12: Resultados dos testes do Modelo 5 aplicado aos conjuntos Cj-02-B's

	Orientação de Itens	Nº Restr.	Nº Var.	Tempo (seg)	Sobra Material
Cj-02-B1	Fixa ( $\ell_i \times w_i$ )	199	327	2,9	1,7 %
	Fixa ( $w_i \times \ell_i$ )	199	327	2,4	2,9 %
Cj-02-B2	Fixa ( $\ell_i \times w_i$ )	310	519	18,1	1,7 %
	Fixa ( $w_i \times \ell_i$ )	307	511	7,9	2,9 %
Cj-02-B3	Fixa ( $\ell_i \times w_i$ )	347	583	10,1	1,7 %
	Fixa ( $w_i \times \ell_i$ )	361	603	13,2	2,9 %

### **4.1.3 Conclusão**

Apesar dos padrões de corte gerados pelo Modelo 5 terem apresentado, em geral, bons índices de sobra (abaixo de 6%), o Modelo 5 mostrou-se inviável para a aplicação em conjuntos com mais de 12 itens, ou 6 se considerarmos a rotação dos itens, devido aos elevados tempos computacionais apresentados para estes conjuntos. Os melhores resultados, considerando-se o tempo computacional e o índice de sobra de material, foram obtidos com os conjuntos que continham de 4 à 6 elementos. Abaixo de quatro elementos o modelo apresentou problemas com os índices de sobra.

O desempenho do Modelo 5 também foi afetado pela diversidade nas dimensões dos itens. Exemplos com uma grande dispersão nas dimensões dos itens tendem a precisar de um maior tempo de processamento e uma elevação no índice de sobra. Quanto maior a dispersão das dimensões, maior será a probabilidade de geração de padrões homogêneos, pois menores serão as possibilidades de combinarmos itens diferentes para a confecção de faixas exatas, ou seja, faixas em que todos os itens tenham a mesma largura, ou o mesmo comprimento.

Um outro fator abordado, que afetou muito o processamento do Modelo 5, foi o tamanho dos itens processados. A presença de itens pequenos, ou seja, itens com pelo menos uma dimensão pequena, eleva o tempo de processamento. Esse problema, no entanto, pode ser contornado efetuando o processamento com orientação fixa dos itens. Já quando os exemplares envolveram itens de dimensões muito pequenas, nem mesmo a opção de manter os itens com orientação fixa garantiu um bom tempo de processamento.

Com isso, temos que a aplicação do Modelo 5 é muito suscetível as características do conjunto que está sendo processado. Não sendo recomendado o processamento de conjuntos com mais de seis itens, se estes apresentarem uma grande diversidade nas dimensões dos itens. Observamos ainda que, sempre que possível, devemos usar o agrupamento dos itens pequenos.

## **4.2 Estudo computacional das heurísticas propostas**

Para uma melhor análise dos dados, o resultado obtido pelas heurísticas propostas na seção 2.4 foram comparados aos resultados gerados pelo sistema CorteBi (Cavali e Rangel, 2004), e pelos padrões utilizados pela indústria. Na análise foram consideradas as perdas ocasionadas pela sobra de matéria prima e o atendimento à demanda na produção de um lote de armários de 5 portas (40 unid.), um lote de cômodas (160 unid.)

e um lote de criados (300 unid.). O objetivo foi testar as heurísticas para produtos com uma quantidade de itens bem diferentes. Nas tabelas 4.13 à 4.15 apresentamos a composição destes produtos, agrupando seus diversos itens conforme suas espessuras, onde Painel indica a espessura do objeto, Item indica o código e o nome dos itens, Qtde indica a quantidade do item utilizados no produto e  $l_i \times w_i$  são, respectivamente, o comprimento e a largura de cada item. Os objetos utilizados possuem dimensões ( $L \times W$ ) de 2750x1830, exeto os objetos de 15mm que possuem dimensões de 2750x1850.

tabela 4.13: Componentes da Criado

Painel	Item		Qtde	$l_i \times w_i$ (mm x mm)		
03 mm	crd-0301	½Contra fdo do criado	4	445	x	213
	crd-0302	Lateral do criado	2	410	x	383
	crd-0303	Fundo das Gavetas	3	388	x	377
12 mm	crd-1201	Lateral da gaveta	6	390	x	110
	crd-1206	Contra fdo gaveta méd	3	370	x	110
15 mm	crd-1501	Tampo e Base	2	600	x	440
	crd-1502	Frente gaveta	3	450	x	132

tabela 4.14 Componentes do Armário de 5 portas

Painel	Item		Qtde	$l_i \times w_i$ (mm x mm)		
03 mm	a5p-0301	Tampo da Base	6	2500	x	565
	a5p-0302	Fundo gaveta	4	647	x	453
	a5p-0303	Contra fdo gaveteiro gde	2	710	x	454
	a5p-0304	Contra fdo gaveteiro peq	2	454	x	454
	a5p-0305	Contra fdo camiseiro	5	1080	x	454
	a5p-0306	Contra fdo maleiro	5	530	x	454
	a5p-0307	Divisória do camiseiro	1	1050	x	500
	a5p-0308	½ Lateral sapateira	2	483	x	215
09mm	A5p-0901	Prateleira do Calceiro	1	450	x	510
12mm	A5p-1201	Lateral gaveta gde	8	454	x	180
	A5p-1202	Contra fdo gaveta gde	4	635	x	180
15 mm	a5p-1501	Porta camiseiro	5	1049	x	452
	a5p-1502	Porta maleiro	5	499	x	452
	a5p-1503	Porta do gaveteiro	2	452	x	429
	a5p-1504	Lateral do camiseiro	2	1050	x	535
	a5p-1505	Lateral do maleiro	2	535	x	500
	a5p-1506	Lateral do gaveteiro	2	535	x	430
	a5p-1507	Frente gavetão	4	700	x	212
20 mm	a5p-2001	Molduras tras. e diant. base	12	2500	x	60
	a5p-2002	Lateral da base	12	445	x	60
	a5p-2003	Travessa da base	26	445	x	40
	a5p-2004	Travessa da corrediça	1	490	x	60
	a5p-2005	Sarrafo tras. maleiro	3	500	x	60
	a5p-2006	Sarrafo tras. camiseiro	5	1050	x	60
	a5p-2007	Sarrafo tras. gaveteiro	3	430	x	60
	pnz-2001	Pinázio grosso (20-1)	4	440	x	60
	pnz-2002	Pinázio grosso (20-2)	2	1060	x	60
	25 mm	pnz-2501	Pinázio grosso (25-01)	2	430	x
pnz-2502		Pinázio grosso (25-02)	4	500	x	60
pnz-2503		Pinázio grosso (25-03)	4	1050	x	60

Tabela 4.15: Componentes da Cômoda

Painel	Item		Qtde	$l_i \times w_i$ (mm x mm)		
03 mm	cmd-0301	Contra fundo	2	710	x	535
	cmd-0302	Lateral	2	1062	x	530
	cmd-0303	Fundo das Gavetas	6	647	x	453
09 mm	cmd-0901	Travessa colméia gde	3	630	x	50
	cmd-0902	Travessa colméia med	2	433	x	50
	cmd-0903	Travessa colméia peq	3	295	x	50
12 mm	cmd-1201	Lateral gaveta multiuso	2	440	x	65
	cmd-1202	Contra fdo gav multiuso	1	635	x	50
	cmd-1203	Lateral da gaveta gde	6	454	x	180
	cmd-1204	Contra fdo gaveta gde	3	635	x	180
	cmd-1205	Lateral da gaveta méd	4	454	x	135
	cmd-1206	Contra fdo gaveta méd	2	635	x	135
15 mm	cmd-1501	Tampo e Base	2	970	x	570
	cmd-1502	Frente gaveta multiuso	1	700	x	75
	cmd-1503	Frente gaveta gde	3	700	x	212
	cmd-1504	Frente gaveta méd	2	700	x	163
	pnz-1501	Pinázio grosso (15-1)	0	430	x	60
	pnz-1502	Pinázio grosso (15-2)	0	500	x	60

Para realização dos testes, foram montados conjuntos de dados por produto e painel, conforme o exposto no Apêndice A. Em alguns casos, como no conjunto para o painel de 09mm do armário de 5 portas, foram inseridos itens adicionais com demanda zero. Estes itens adicionados são componentes de um outro móvel ou pinázios, itens utilizados em larga escala pela indústria para a confecção das estruturas dos móveis. O objetivo da adição destes itens foi melhorar o aproveitamento do objeto.

Os testes foram realizados em 2 fases: na primeira, realizamos uma análise dos índices de sobra e tempos de resolução das duas heurísticas propostas; e na segunda, fazemos uma comparação dos resultados obtidos com a Heurística 2 com os resultados obtidos pelo sistema CorteBi e os informados pela Indústria.

#### 4.2.1 Fase 1: Heurística 1 x Heurística 2

Na tabela 4.16, são apresentados os dados relativos às sobras verificadas no processo de produção de um lote de Armários 5p, um lote de cômodas e um lote de criados, pelas duas heurísticas. A coluna Nr Padrões indica o número de padrões distintos que foram utilizados, Nr Objetos indica o número total de objetos cortados e para a Sobra de Material, calculada conforme a fórmula (2.02) no capítulo 2, informamos o valor máxi-

mo, a média e o desvio padrão. Na execução da Heurística 2 foi sempre considerado a rotação dos itens, enquanto da Heurística 1, em alguns casos (assinalados com “\*” na tabela 4.16), foi executada considerando a orientação fixa dos itens.

Tabela 4.16: Comparativo do aproveitamento de matéria-prima entre as Heurísticas 1 e 2

		Heurística 1					Heurística 2				
		Nr Padrões	Nr Objetos	Sobra de Material			Nr Padrões	Nr Objetos	Sobra de Material		
				Max.	Méd.	Min.			Max.	Méd.	Min.
Criado	03 mm	5	72	6,3 %	4,2 %	3,2%	3	73	11,2 %	5,5 %	3,2%
	12 mm *	5	23	5,3 %	1,2 %	0,2%	3	24	2,0 %	1,4 %	1,1%
	15 mm	2	45	5,5 %	3,7 %	3,1%	2	45	5,5 %	3,7 %	3,1%
Cômoda	03 mm	2	124	5,9%	5,8%	5,8%	2	124	5,9%	5,8%	5,8%
	09 mm *	3	7	10,5%	4,4%	3,3%	3	7	1,8%	1,5%	0,2%
	12 mm *	7	47	8,6%	4,5%	1,5%	8	45	1,7%	0,7%	0,2%
	15 mm *	5	83	27,9%	12,5%	9,7%	3	64	5,0%	4,3%	3,7%
Arm. 5p	03 mm	9	130	9,0%	5,8%	0,3%	9	133	9,0%	6,0%	0,3%
	09 mm	1	2	2,0%	2,0%	2,0%	1	2	2,0%	2,0%	2,0%
	12 mm	2	10	4,7%	4,4%	1,5%	2	10	2,2%	2,1%	1,5%
	15 mm	10	57	6,0%	4,1%	1,6%	9	57	5,4%	4,0%	1,5%
	20 mm *	10	33	15,1%	8,5%	2,3%	10	30	6,6%	3,1%	0,5%
	25 mm *	3	5	10,6%	7,0%	2,6%	2	4	2,3%	2,0%	1,0%

\* Execução da Heurística 1 considerando a orientação fixa dos itens

Se concentrarmos nossa análise da Tabela 4.16 no número de padrões utilizados, poderemos ver uma vantagem da Heurística 2 sobre a Heurística 1. A utilização de um maior número de padrões pode acarretar em um aumento da carga de trabalho, pois aumentamos a possibilidade de um padrão ser utilizado para cortar poucos objetos, o que torna pouco produtivo o uso da seccionadora. Observando agora o número de objetos cortados, veremos que, com exceção de alguns painéis, a Heurística 1 exibe o menor consumo de objetos. As exceções (os painéis de 12 e 15mm para a cômoda e 20 e 25mm para o armário) são conjuntos compostos por muitos itens pequenos, críticos para a execução do modelo 5 (seção 4.1) usado na Heurística 1. Só sendo viável, para estes casos, a execução da Heurística 1 com orientação fixa dos itens (Tabela 4.17), o que reduziu o espaço solução não permitindo um melhor aproveitamento do objeto. Observamos ainda que a Heurística 2 apresenta, em geral, uma menor amplitude nos índices de sobra (Max–Min, na Tabela 4.16). O que nos sugere um conjunto de padrões de corte com um índice de perda mais uniforme.

Um outro aspecto, que foi observado em nossos testes, é o tempo computacional. O tempo consumido pela Heurística 1 para encontrar os padrões de corte para um conjunto de itens é da ordem de dezena de minutos, chegando a centenas. Enquanto o tem-

po utilizado pela Heurística 2 é de dezenas de segundos. O tempo mais alto verificado pela Heurística 2 foi de 90s, contra mais de 10h para a Heurística 1 (Tabela 4.17).

Tabela 4.17: Comparativo do tempo de processamento das Heurísticas 1 e 2

		Heurística 1			Heurística 2	
		Qtde Itens	Tempo (segundos)	Observações	Tempo (segundos)	Observações
Criado	03 mm	3	1188	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>1 item de dimensões mpq</li> </ul>	5,0	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	12 mm	2	3600	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>2 itens de dimensões mpq</li> </ul>	4,4	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	15 mm	2	2859	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>1 item de dimensões mpq</li> </ul>	3,0	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
Cômuda	03 mm	3	5	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>	4,0	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	09 mm	3	292	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>3 itens de dimensões mpq</li> </ul>	8,6	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	12 mm	6	136	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>4 itens de dimensões peq</li> <li>2 itens de dimensões mpq</li> <li>Processamento dividido em 2 conjuntos de 3 itens cada</li> </ul>	36,2	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	15 mm	6	64	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>2 itens de dimensões peq</li> <li>3 itens de dimensões mpq</li> <li>Processamento dividido em 2 conjuntos de 4 itens cada</li> <li>2 itens mpq com demanda 0</li> </ul>	19,0	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
Arm. 5p	03 mm	8	16600	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>1 item de dimensões peq</li> </ul>	33,4	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	09 mm	4	27	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>3 itens de dimensões mpq</li> <li>Itens mpq com demanda 0</li> </ul>	5,4	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	12 mm	3	1920	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>3 itens de dimensões peq</li> </ul>	7,7	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	15 mm	9	46	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> <li>1 item de dimensões peq</li> <li>2 itens de dimensões mpq</li> <li>Itens mpq com demanda 0</li> </ul>	66,2	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	20 mm	9	3070	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>9 itens de dimensões mpq</li> </ul>	94,4	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>
	25 mm	3	69	<ul style="list-style-type: none"> <li>Orientação fixa dos itens</li> <li>3 itens de dimensões mpq</li> </ul>	9,2	<ul style="list-style-type: none"> <li>Considerada rotação dos itens</li> </ul>

Obs.: Dimensões peq = Dimensões pequenas; Dimensões mpq = Dimensões muito pequenas

#### 4.2.2 Fase 2: Heurística 2 x CorteBi x Indústria

Nesta fase efetuaremos uma comparação dos resultados obtidos com o *Pool* de padrões tabuleiros compostos, pela aplicação da Heurística 2, com os resultados obtidos

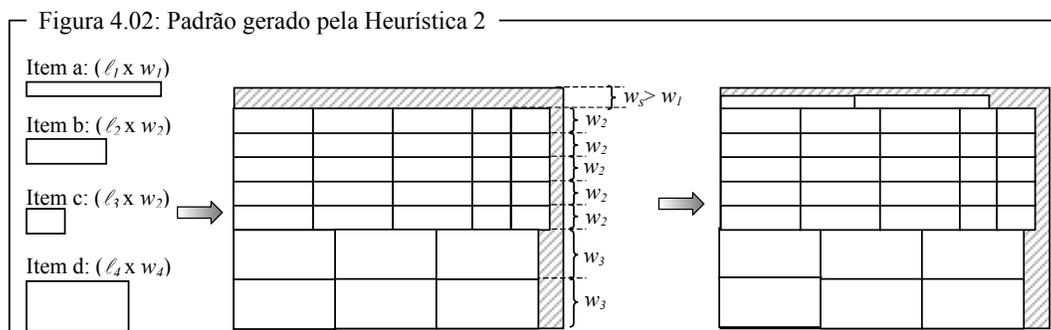
peelo sistema CorteBi, que trabalha com padrões em 2-estágios, e com os padrões adotados pela indústria. Iniciaremos, novamente, a análise pelos dados relativos ao aproveitamento de material na produção de um lote de Armários 5p, um lote de cômodas e um lote de criados, apresentados Tabela 4.18.

Tabela 4.18: Comparativo do aproveitamento da matéria prima entre Heurística 2, CorteBi e Indústria

		Heurística 2					CorteBi					Indústria				
		Nr Padr.	Nr Obj.	Sobra de Material Max.	Méd.	Min.	Nr Padr.	Nr Obj.	Sobra de Material Max.	Méd.	Min.	Nr Padr.	Nr Obj.	Sobra de Material Max.	Méd.	Min.
Criado	03 mm	3	73	11,2 %	5,5 %	3,2%	3	72	5,2%	1,9%	1,7%	2	74	6,5%	5,0%	4,8%
	12 mm	3	24	2,0 %	1,4 %	1,1%	2	25	1,1%	1,1%	1,1%	2	24	1,9%	1,3%	1,1%
	15 mm	2	45	5,5 %	3,7 %	3,1%	2	45	5,5%	3,1%	2,7%	2	45,5 *	5,5%	4,9%	3,3%
Cômoda	03 mm	2	124	5,9%	5,8%	5,8%	2	124	5,9%	5,8%	5,8%	2	124	5,9%	5,8%	5,8%
	09 mm	3	7	1,8%	1,5%	0,2%	3	7	2,8%	1,5%	0,2%	1	6,5 *	3,3%	3,3%	3,3%
	12 mm	8	45	1,7%	0,7%	0,2%	6	48	2,6%	0,6%	0,2%	6	46	2,8%	1,4%	1,7%
	15 mm	3	64	5,0%	4,3%	3,7%	4	63	5,8%	4,8%	4,4%	3	63,5 *	4,9%	4,4%	1,3%
Arm. 5p	03 mm	9	133	9,0%	6,0%	0,3%	7	135	9,0%	6,2%	0,3%	7	134	9,0%	6,6%	0,3%
	09 mm	1	2	2,0%	2,0%	2,0%	1	2	1,7%	1,7%	1,7%	1	2	1,8%	1,8%	1,8%
	12 mm	2	10	2,2%	2,1%	1,5%	2	11	4,7%	4,6%	4,0%	2	10,5 *	4,5%	3,3%	2,3%
	15 mm	9	57	5,4%	4,0%	1,5%	7	60	8,4%	5,4%	3,6%	6	57	4,3%	2,6%	2,3%
	20 mm	10	30	6,6%	3,1%	0,5%	9	31	2,6%	1,9%	0,1%	8	32	6,5%	2,7%	1,5%
	25 mm	2	4	2,3%	2,0%	1,0%	3	5	2,8%	2,5%	2,3%	3	4,5 *	3,3%	3,2%	3,2%

(\*) Valores obtidos considerando a realização de “½ padrão de corte”

Analisando a quantidade de padrões utilizados, numa comparação com o sistema CorteBi, verificamos a tendência da Heurística 2 em trabalhar com um número superior de padrões. No entanto, a análise feita nos padrões gerados mostrou que os padrões do CorteBi tendem a apresentar uma maior diversidade de faixas. A Heurística 2 gera padrões considerando duas faixas distintas. No caso da largura da faixa de sobra ( $w_s$ ) ser maior que a menor das larguras dos itens ( $w_{min}$ ) a Heurística 2 admite construir mais uma faixa, afim de reduzir sobra de material (Figura 4.02). A relevância desta observação reside no exposto no capítulo 3, onde vimos que padrões com uma maior diversidade de faixas demandam um maior tempo de ajustes da seccionadora.





Podemos perceber que ao atendermos à demanda usando os padrões gerados pela Heurística 2, obtemos uma economia de matéria prima de 1,6 % em relação ao CorteBi (dez painéis) e de 1,0 % em relação a Indústria (seis painéis). Cabe ressaltar que os valores verificados pela Indústria foram obtidos considerando a utilização de ½ padrão de corte, o que não ocorre na Heurística 2 e nem no Cortebi. Se desconsiderarmos a utilização de ½ padrão de corte, o consumo de objetos da Indústria, na tabela 4.19, será igual ou superior a do CorteBi.

Um outro ponto interessante, que surge na análise da tabela 4.19, é fato do CorteBi apresentar uma perda total menor que a da Heurística 2, mesmo consumindo mais objetos. Isto nos remete a um aspecto que ainda não consideramos, o excedente de produção.

A análise dos excedentes de produção (Tabela 4.20) nos mostra que, de uma forma geral, com a aplicação da Heurística 2 conseguimos um atendimento da demanda com um menor número de itens excedentes. Tanto na comparação com o CorteBi como com a Indústria. Apesar de um excedente de produção não implicar, diretamente, em uma perda, devemos estar atentos a ele, pois a este excedente está atrelado um custo de armazenamento e, no caso de pequenas empresas, um custo operacional, visto que na falta de um espaço para criação de um estoque intermediário este excedente acaba armazenado na própria área de produção.

Tabela 4.20: Resumo do excedente de produção

			Heurística 2		CorteBi		Indústria	
Prod.	Painel	Demanda	Produção	Excedente	Produção	Excedente	Produção	Excedente
Criado	03mm	2700	2709	9 ( 0,3% )	2778	78 ( 2,8% )	2767	67 ( 2,5% )
	12mm	2700	2708	8 ( 0,3% )	2828	128 ( 4,5% )	2710	10 ( 0,4% )
	15mm	1500	1536	36 ( 2,3% )	1532	32 ( 2,1% )	1532	32 ( 2,1% )
	SubTot.	6900	6953	53 ( 0,8% )	7138	238 ( 3,3% )	7009	109 ( 1,6% )
Cômoda	03mm	1600	1600	0 ( 0,0% )	1600	0 ( 0,0% )	1600	0 ( 0,0% )
	09mm	1280	1407	127 ( 9,0% )	1418	138 ( 9,7% )	1280,5	0,5 ( 0,0% )
	12mm	2880	2934	54 ( 1,8% )	3143	263 ( 8,4% )	2950	70 ( 4,3% )
	15mm	1280	1320	40 ( 3,0% )	1309	29 ( 2,2% )	1299,5	19,5 ( 2,4% )
	SubTot.	7040	7261	221 ( 3,0% )	7470	430 ( 5,8% )	7189	149 ( 2,1% )
Armário 5p	03mm	1080	1258	178 ( 14,1% )	1280	200 ( 15,6% )	1268	188 ( 14,8% )
	09mm	40	64	24 ( 37,5% )	70	30 ( 42,9% )	64	24 ( 37,5% )
	12mm	480	528	48 ( 9,1% )	558	78 ( 14,0% )	558	78 ( 14,0% )
	15mm	880	902	22 ( 2,4% )	961	81 ( 8,4% )	893	13 ( 1,5% )
	20mm	2480	2596	116 ( 4,5% )	3031	551 ( 18,2% )	2619	139 ( 5,3% )
	SubTot.	4960	5348	388 ( 7,3% )	5900	940 ( 15,9% )	5562	602 ( 10,8% )
TOTAL		18900	19562	662 ( 3,4% )	20323	1423 ( 7,0% )	19760	860 ( 4,4% )

No resumo do excedente de produção por produto, apresentado na Tabela 4.20, não consideramos o excedente dos pinázios. Pois, como a demanda por pinázios é grande, qualquer excedente em sua produção é facilmente compensada na produção de um próximo lote.

### 4.2.3 Fase 3: Programação de lotes duplos

Para reduzir o tempo gasto com *setup* das máquinas e melhorar o aproveitamento da mão-de-obra, a Indústria estudada busca, como alternativa, a programação de lotes duplo. Estes lotes duplos poderão ser: homogêneos, se agruparmos dois lotes de um mesmo produto; e heterogêneo, se agruparmos lotes de produtos diferentes.

Na produção de lotes duplos homogêneos (Tabela 4.21), verificamos que a Heurística 2 obteve praticamente os mesmos índices de sobra que a Indústria, porém com um menor consumo de objetos, 1,4 % a menos no lote de Criados, 1,8 % no lote de Cômodas e 4,4 % no lote de Armários de 5 portas. A mesma análise feita para o Corte-Bi, em relação à Indústria, revela um quadro diferente. Para os lote de Criados e Cômodas o CorteBi apresenta os menores índices de sobras (2,1 % e 4,4 %, respectivamente) e os menores consumos de objetos (3,1 % e 2,4 %, respectivamente). Para o lote de Armários de 5 portas, o CorteBi apresenta um índice de sobra meio ponto percentual maior que a Indústria, porém isto é compensado pela redução, no consumo, de um objeto. Com isso, no que tange ao consumo de objeto e índices de sobra, a Heurística 2 e o CorteBi se mostram boas alternativas a prática da Indústria.

Tabela 4.21: Resumo do aproveitamento de material por lote do produto

Lote	Produtos	Heurística 2			CorteBi			Indústria		
		Nr. Padr.	Nr. Obj	Sobras	Nr. Padr.	Nr. Obj	Sobras	Nr. Padr.	Nr. Obj	Sobras
Lote Duplo Homogêneo	Criados	8	283	4,2%	7	278	2,1%	6	287	4,4%
	Cômodas	18	477	4,3%	15	479	4,4%	12	481	4,5%
	Armários 5p	40	459	5,0%	29	479	5,4%	27	480	4,9%
Lote Duplo Heterogêneo	Arm. 5p + Criado	49	368	3,7%	36	386	4,4%	33	384	4,7%
	Arm. 5p + Cômoda	47	478	4,5%	41	477	4,5%	39	480	4,7%
	Cômoda + Criado	24	381	2,8%	22	385	3,6%	18	383	4,5%

Na análise dos lotes duplos heterogêneos, com a aplicação da Heurística 2 obtivemos, na comparação com a Indústria, uma redução no consumo de objetos nos três casos pesquisados (lote duplo de Armário+Criado, de Armário+Cômoda e Cômoda+Criado), sendo bastante significativa a redução no lote duplo Armário+Criado, que apresentou um economia de 4,2 % no número de objetos cortados (16 objetos). Os ín-

dices de sobra e consumo menor de objetos indicam um melhor aproveitamento do material cortado por parte da Heurística 2. Isso coloca a Heurística 2 como uma boa alternativa à prática adotada pela Indústria. O CorteBi, apesar de apresentar menores índices de sobra, só apresenta um bom resultado no lote duplo Armário+Cômoda onde apresenta uma economia de 0,6 % no consumo de objetos (três objetos). Nos outros dois lotes pesquisados o apresentaram um consumo maior de objetos (0,4 %).

Com respeito ao excedente de produção, é possível perceber, pela tabela 4.22, que a Heurística 2 apresenta um melhor resultado, quando comparado aos resultados obtidos pelo CorteBi e pela Indústria.

Tabela 4.22: Resumo do excedente de produção nos lotes duplos

Lt	Produto	Demanda	Heurística 2		CorteBi		Indústria	
			Produção	Excedente	Produção	Excedente	Produção	Excedente
Homogêneo	Criado	13800	13869	69 ( 0,5%)	13910	110 ( 0,8%)	14018	218 ( 1,6%)
	Cômoda	14080	14240	160 ( 1,1%)	14620	540 ( 3,8%)	14387	307 ( 2,2%)
	Armário 5p	9720	10315	595 ( 5,8%)	10865	1145 ( 10,5%)	11124	1404 ( 12,6%)
Heterogêneo	Arm 5p + Criado	4960	5153	193 ( 3,7%)	5711	751 ( 13,2%)	5561	601 ( 10,8%)
	Arm 5p + Cômoda	9760	10237	477 ( 4,7%)	10814	1054 ( 9,7%)	10388	628 ( 6,0%)
	Cômoda + Criado	13940	9516	196 ( 2,1%)	9711	391 ( 4,0%)	9527	207 ( 2,2%)

## Conclusão

Elaboramos duas heurísticas para geração de padrões tabuleiros compostos, a primeira tomando por base o modelo de Scheithauer (2002) adaptado para gerar o padrão tabuleiro exato (Yanasse e Morábito, 2006). Como nos testes que fizemos, essa heurística apresentou tempos elevados de processamento (seção 4.1), desenvolvemos uma segunda heurística, tomando por base o método de Gilmore e Gomory (1965).

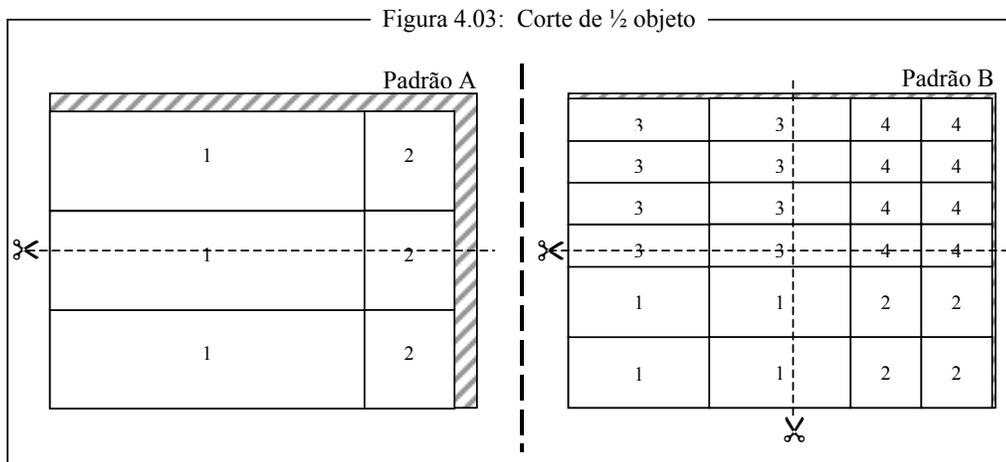
Comparamos os resultados da Heurística 2 com os resultados do sistema CorteBi e com os obtidos pela Indústria, pela aplicação de seus padrões de corte. Esta comparação se deu em três testes distintos: o primeiro, considerando lotes simples, ou seja, o processamento de um lote de um produto; o segundo, considerando lotes duplos homogêneos, ou seja, o processamento de dois lotes de um único produto; e o terceiro, considerando lotes duplos heterogêneos, ou seja, o processamento de dois lotes de produtos diferentes.

Os resultados confirmaram a Heurística 2 como uma boa alternativa à prática da Indústria estudada. Os padrões de corte gerados pela Heurística 2, que compõem um *Pool* de padrões tabuleiros compostos, obedecem às diretrizes estabelecidas pela Indústria para um bom padrão de corte, ou seja, são padrões em dois estágios que, em sua

maioria, apresentam apenas duas faixas distintas, com poucos tipos de itens diferentes. Os resultados apresentados pela Heurística 2 indicaram uma redução no consumo de objetos. Os padrões de corte gerados pela Heurística 2 apresentaram, em geral, menores índices de sobre média de material e indicaram uma menor produção de itens excedentes, tanto na comparação com o CorteBi quanto com a Indústria. No entanto a Heurística 2 exibe uma característica que não é bem vista pela Indústria. O elevado número de padrões de corte utilizados. O número de padrões de corte utilizados pela Heurística 2 foi superior aos do CorteBi e aos adotados pela Indústria. Um alto número de padrões de corte aumenta a probabilidade de uma sub-utilização da seccionadora, ou seja, de operarmos a seccionadora muito abaixo de sua capacidade, sem falar nos transtornos operacionais de carga e ajuste dos objetos e o aumento no tempo gasto com ajuste dos batentes.

Durante os testes comparativos da Heurística 2 com a Indústria, vimos que em algumas situações a Indústria processa apenas metade do padrão de corte, ou seja, corta somente  $\frac{1}{2}$  padrão de corte. Com isso ela consegue reduzir a produção de itens excedentes. A Heurística 2 e o CorteBi não estão aptos a processar  $\frac{1}{2}$  padrão de corte. Isto se deve a dois fatores:

1. A Heurística 2 e o CorteBi buscam sempre uma solução inteira para o problema, não admitindo portanto uma solução fracionária como solução válida;
2. Ainda que eles admitam uma solução fracionária, isto não é suficiente para considerar o  $\frac{1}{2}$  padrão de corte, pois é preciso uma análise do objeto para verificar a viabilidade deste tipo de corte. Na Figura 4.03 apresentamos dois padrões de corte em que questionamos essa viabilidade. Na Figura 4.03a temos que o  $\frac{1}{2}$  padrão de corte implicará num aumento da sobre de material, e no padrão B a questão é como efetuar o  $\frac{1}{2}$  padrão de corte.



Quando a Indústria pensa no “ $\frac{1}{2}$  padrão de corte”, ela deseja reduzir a produção de itens excedentes utilizando, aproximadamente, metade de um objeto. Ela não pensa numa utilização diferente do objeto ( $\frac{1}{4}$ ,  $\frac{1}{3}$ ,  $\frac{2}{3}$ ,  $\frac{3}{4}$ , ...), pois isso complicaria o controle do estoque dos objetos, que passaria a ter um estoque de objetos composto por retalhos de diversos tamanhos, dificultando a programação do corte dos objetos. Contudo, incluir na Heurística 2 e no CorteBi a possibilidade de utilização de  $\frac{1}{2}$  objeto, pode ser interessante, pois terá efeito direto sobre a produção de itens excedentes.



Nesta última seção, discorreremos sobre as investigações realizadas ao longo da dissertação e teceremos os possíveis desdobramentos delas, identificando pontos que merecem uma melhor avaliação, ou mesmo serem investigadas.

Na seção 2.1, discutimos a qualidade de um padrão de corte, tomando por base o seu percentual de perda. Vimos que este percentual é composto por duas partes: uma devido ao desgaste da serra, durante o processo de corte; e outra referente aos retângulos não demandados e sem utilidade para o processo produtivo, as sobras de matéria-prima. Vimos que adotar o percentual de perda total (desgaste da serra + sobra de matéria-prima) é importante do ponto de vista contábil, porém pode conduzir a um erro de avaliação quando desejamos aferir a eficiência de um padrão de corte. Por isso, sugerimos que a análise da eficiência de um padrão de corte seja efetuada tomando por base,

apenas, a perda relativa as sobras de matéria-prima, aqui denominada percentual de sobra ou índice de sobra. Uma outra questão relativa a qualidade do padrão de corte, levantada na seção 2.1 e discutida na seção 4.2, é o excedente de produção que, apesar de não caracterizar uma perda, implica em um custo adicional (financeiro e/ou operacional) para o armazenamento dos itens excedentes. Neste caso a avaliação de qualidade não deve ser feita para um único padrão de corte, mas sim para o conjunto de padrões de corte adotados para o atendimento da demanda.

O entendimento do processo de corte realizado na Indústria, com todas as particularidades na movimentação dos objetos, e a análise dos padrões de corte adotados, permitiu identificar um tipo de padrão que agrega padrões tabuleiros, o qual nomeamos de padrão tabuleiro composto. Os padrões tabuleiros compostos, que pertencem a classe dos padrões  $n$ -grupos, exibem as facilidades de corte dos padrões tabuleiros, porém com um melhor aproveitamento do objeto.

Na busca de uma solução que utilizasse apenas padrões tabuleiros compostos, criamos dois processos heurísticos (seção 2.4). O primeiro, a Heurística 1, tomou por base o modelo para a geração de padrões tabuleiros não-exatos de Scheithauer (2002) adaptado para o caso exato (Yanasse e Morábito, 2006). Esta heurística mostrou-se muito sensível a quantidade de itens distintos considerados e ao processamento de itens com dimensões pequenas. No modelo de Scheithauer (Modelo 4, seção 2.2), a variável  $x$  tem dimensão  $n \cdot J \cdot K$  (onde:  $n$  é o número de itens;  $J = \lfloor L/\ell_{\min} \rfloor$  é o número máximo de faixas ao longo do comprimento do objeto; e  $K = \lfloor W/\ell_{\min} \rfloor$  é o número máximo de faixas ao longo da largura do objeto). Itens com uma das dimensões pequenas tende a elevar o valor de  $J$  ou de  $K$ , ou de ambos se considerarmos permitida a rotação dos itens, e quanto maior o produto  $n \cdot J \cdot K$  mais pesado é o processamento do modelo.

Nos testes realizados (seção 4.1) percebemos que, quando da existência de itens com dimensões pequenas, o processamento da Heurística 1 considerando a orientação fixa dos itens, em geral, apresentou melhores tempos computacionais. No entanto, houve restrição do universo de soluções, pois a Heurística 1, e também a Heurística 2, quando consideram a rotação, ou não, dos itens o fazem para todos os itens do conjunto. Para contornar isso, seria necessário trabalharmos com uma “rotação seletiva” dos itens, ou seja, apenas os itens com problemas teriam suas orientações fixadas.

A Heurística 2, toma por base o processo descrito por Gilmore e Gomory (1965) para a geração de padrões de corte em dois estágios, ou seja, cria faixas e as combina para obter os padrões de corte desejados, ou seja, são criadas até duas faixas para cada item e, então, o padrão de corte é gerado considerando no máximo duas faixas distintas (seção 2.4). Com isso a Heurística 2 mostrou-se bem menos sensível à quantidade de itens distintos e não apresentou sensibilidade à variação das dimensões dos itens.

Contudo, apesar dos resultados obtidos pela Heurística 2 terem se mostrado, de forma geral, melhores que os adotados pela Indústria, dois grandes problemas a desabonam: o primeiro é a grande quantidade de padrões distintos usados no processo e o segundo, quase que consequência do primeiro, é a subutilização da seccionadora em algumas operações de corte. A seqüência de ajustes da seccionadora para o corte de uma quantidade de objetos, limitada à sua capacidade de corte, é definida como ciclo da serra (Yanasse *et al*). A minimização do número de ciclos da serra contribui para a redução do número de padrões distintos usados no processo de produção, pois o número de vezes que alguns padrões de corte são usados aumenta, o que pode favorecer a taxa de utilização da seccionadora. A redução do número de padrões de corte, também pode ser obtida com a inclusão de variáveis de “*setup*”, penalizando a diversidade de padrões (Figueiredo e Rangel, 2005). Se a diversidade de padrões utilizados no processo de corte é reduzida, a tendência é aumentar a quantidade de vezes que os padrões escolhidos serão usados. Podemos, ainda, desenvolver uma forma de penalizar soluções que apresentem baixas taxas de utilização da seccionadora, principalmente quando esta taxa é inferior a 50% da capacidade da máquina.

Uma outra questão verificada, é o fato da Heurística 2 trabalhar com um *Pool*, de padrões tabuleiros compostos, com várias dezenas de padrões, sendo que a solução ótima utiliza poucos padrões. Apesar da Heurística 2 apresentar tempos plenamente aceitáveis pela Indústria, essa questão indica um esforço computacional desnecessário. A idéia é incorporar a Heurística 2 ao CorteBi que, por já considerar a técnica de geração de colunas na criação dos planos de corte, irá reduzir o tamanho do *Pool*.



A1. Conjuntos utilizados nos testes do Capítulo 4

	Armário 5 portas	Cômoda	Criado																																																																								
<b>03 mm</b> 2750 x 1830	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>a5p-0301</td><td>2500</td><td>565</td><td>240</td></tr> <tr><td>a5p-0302</td><td>647</td><td>453</td><td>160</td></tr> <tr><td>a5p-0303</td><td>710</td><td>454</td><td>80</td></tr> <tr><td>a5p-0304</td><td>454</td><td>454</td><td>80</td></tr> <tr><td>a5p-0305</td><td>1080</td><td>454</td><td>200</td></tr> <tr><td>a5p-0306</td><td>530</td><td>454</td><td>200</td></tr> <tr><td>a5p-0307</td><td>1050</td><td>500</td><td>40</td></tr> <tr><td>a5p-0308</td><td>483</td><td>215</td><td>80</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	a5p-0301	2500	565	240	a5p-0302	647	453	160	a5p-0303	710	454	80	a5p-0304	454	454	80	a5p-0305	1080	454	200	a5p-0306	530	454	200	a5p-0307	1050	500	40	a5p-0308	483	215	80	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>cmd-0301</td><td>710</td><td>535</td><td>320</td></tr> <tr><td>cmd-0302</td><td>1062</td><td>530</td><td>320</td></tr> <tr><td>cmd-0303</td><td>647</td><td>453</td><td>960</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	cmd-0301	710	535	320	cmd-0302	1062	530	320	cmd-0303	647	453	960	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>crd-0301</td><td>445</td><td>213</td><td>1200</td></tr> <tr><td>crd-0302</td><td>410</td><td>383</td><td>600</td></tr> <tr><td>crd-0303</td><td>388</td><td>377</td><td>900</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	crd-0301	445	213	1200	crd-0302	410	383	600	crd-0303	388	377	900				
Item	$l_i$	$w_i$	$b_i$																																																																								
a5p-0301	2500	565	240																																																																								
a5p-0302	647	453	160																																																																								
a5p-0303	710	454	80																																																																								
a5p-0304	454	454	80																																																																								
a5p-0305	1080	454	200																																																																								
a5p-0306	530	454	200																																																																								
a5p-0307	1050	500	40																																																																								
a5p-0308	483	215	80																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
cmd-0301	710	535	320																																																																								
cmd-0302	1062	530	320																																																																								
cmd-0303	647	453	960																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
crd-0301	445	213	1200																																																																								
crd-0302	410	383	600																																																																								
crd-0303	388	377	900																																																																								
<b>09 mm</b> 2750 x 1830	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>A5p-0901</td><td>450</td><td>510</td><td>40</td></tr> <tr><td>cmd-0901</td><td>630</td><td>50</td><td>0</td></tr> <tr><td>cmd-0902</td><td>433</td><td>50</td><td>0</td></tr> <tr><td>cmd-0903</td><td>295</td><td>50</td><td>0</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	A5p-0901	450	510	40	cmd-0901	630	50	0	cmd-0902	433	50	0	cmd-0903	295	50	0	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>cmd-0901</td><td>630</td><td>50</td><td>480</td></tr> <tr><td>cmd-0902</td><td>433</td><td>50</td><td>320</td></tr> <tr><td>cmd-0903</td><td>295</td><td>50</td><td>480</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	cmd-0901	630	50	480	cmd-0902	433	50	320	cmd-0903	295	50	480																																					
Item	$l_i$	$w_i$	$b_i$																																																																								
A5p-0901	450	510	40																																																																								
cmd-0901	630	50	0																																																																								
cmd-0902	433	50	0																																																																								
cmd-0903	295	50	0																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
cmd-0901	630	50	480																																																																								
cmd-0902	433	50	320																																																																								
cmd-0903	295	50	480																																																																								
<b>12 mm</b> 2750 x 1830	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>A5p-1201</td><td>454</td><td>180</td><td>320</td></tr> <tr><td>A5p-1202</td><td>635</td><td>180</td><td>160</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	A5p-1201	454	180	320	A5p-1202	635	180	160	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>cmd-1201</td><td>440</td><td>65</td><td>320</td></tr> <tr><td>cmd-1202</td><td>635</td><td>50</td><td>160</td></tr> <tr><td>cmd-1203</td><td>454</td><td>180</td><td>960</td></tr> <tr><td>cmd-1204</td><td>635</td><td>180</td><td>480</td></tr> <tr><td>cmd-1205</td><td>454</td><td>135</td><td>640</td></tr> <tr><td>cmd-1206</td><td>635</td><td>135</td><td>320</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	cmd-1201	440	65	320	cmd-1202	635	50	160	cmd-1203	454	180	960	cmd-1204	635	180	480	cmd-1205	454	135	640	cmd-1206	635	135	320	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>crd-1201</td><td>390</td><td>110</td><td>1800</td></tr> <tr><td>crd-1206</td><td>370</td><td>110</td><td>900</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	crd-1201	390	110	1800	crd-1206	370	110	900																				
Item	$l_i$	$w_i$	$b_i$																																																																								
A5p-1201	454	180	320																																																																								
A5p-1202	635	180	160																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
cmd-1201	440	65	320																																																																								
cmd-1202	635	50	160																																																																								
cmd-1203	454	180	960																																																																								
cmd-1204	635	180	480																																																																								
cmd-1205	454	135	640																																																																								
cmd-1206	635	135	320																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
crd-1201	390	110	1800																																																																								
crd-1206	370	110	900																																																																								
<b>15 mm</b> 2750 x 1850	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>a5p-1501</td><td>1049</td><td>452</td><td>200</td></tr> <tr><td>a5p-1502</td><td>499</td><td>452</td><td>200</td></tr> <tr><td>a5p-1503</td><td>452</td><td>429</td><td>80</td></tr> <tr><td>a5p-1504</td><td>1050</td><td>535</td><td>80</td></tr> <tr><td>a5p-1505</td><td>535</td><td>500</td><td>80</td></tr> <tr><td>a5p-1506</td><td>535</td><td>430</td><td>80</td></tr> <tr><td>a5p-1507</td><td>700</td><td>212</td><td>160</td></tr> <tr><td>pnz-1501</td><td>440</td><td>60</td><td>0</td></tr> <tr><td>pnz-1502</td><td>1060</td><td>60</td><td>0</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	a5p-1501	1049	452	200	a5p-1502	499	452	200	a5p-1503	452	429	80	a5p-1504	1050	535	80	a5p-1505	535	500	80	a5p-1506	535	430	80	a5p-1507	700	212	160	pnz-1501	440	60	0	pnz-1502	1060	60	0	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>cmd-1501</td><td>970</td><td>570</td><td>320</td></tr> <tr><td>cmd-1502</td><td>700</td><td>75</td><td>160</td></tr> <tr><td>cmd-1503</td><td>700</td><td>212</td><td>480</td></tr> <tr><td>cmd-1504</td><td>700</td><td>163</td><td>320</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	cmd-1501	970	570	320	cmd-1502	700	75	160	cmd-1503	700	212	480	cmd-1504	700	163	320	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>crd-1501</td><td>600</td><td>440</td><td>600</td></tr> <tr><td>crd-1504</td><td>450</td><td>132</td><td>900</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	crd-1501	600	440	600	crd-1504	450	132	900
Item	$l_i$	$w_i$	$b_i$																																																																								
a5p-1501	1049	452	200																																																																								
a5p-1502	499	452	200																																																																								
a5p-1503	452	429	80																																																																								
a5p-1504	1050	535	80																																																																								
a5p-1505	535	500	80																																																																								
a5p-1506	535	430	80																																																																								
a5p-1507	700	212	160																																																																								
pnz-1501	440	60	0																																																																								
pnz-1502	1060	60	0																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
cmd-1501	970	570	320																																																																								
cmd-1502	700	75	160																																																																								
cmd-1503	700	212	480																																																																								
cmd-1504	700	163	320																																																																								
Item	$l_i$	$w_i$	$b_i$																																																																								
crd-1501	600	440	600																																																																								
crd-1504	450	132	900																																																																								
<b>20 mm</b> 2750 x 1830	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>a5p-2001</td><td>2500</td><td>60</td><td>480</td></tr> <tr><td>a5p-2002</td><td>445</td><td>60</td><td>480</td></tr> <tr><td>a5p-2003</td><td>445</td><td>40</td><td>1040</td></tr> <tr><td>a5p-2004</td><td>490</td><td>60</td><td>40</td></tr> <tr><td>a5p-2005</td><td>500</td><td>60</td><td>120</td></tr> <tr><td>a5p-2006</td><td>1050</td><td>60</td><td>200</td></tr> <tr><td>a5p-2007</td><td>430</td><td>60</td><td>120</td></tr> <tr><td>pnz-2001</td><td>440</td><td>60</td><td>160</td></tr> <tr><td>pnz-2002</td><td>1060</td><td>60</td><td>80</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	a5p-2001	2500	60	480	a5p-2002	445	60	480	a5p-2003	445	40	1040	a5p-2004	490	60	40	a5p-2005	500	60	120	a5p-2006	1050	60	200	a5p-2007	430	60	120	pnz-2001	440	60	160	pnz-2002	1060	60	80																																		
Item	$l_i$	$w_i$	$b_i$																																																																								
a5p-2001	2500	60	480																																																																								
a5p-2002	445	60	480																																																																								
a5p-2003	445	40	1040																																																																								
a5p-2004	490	60	40																																																																								
a5p-2005	500	60	120																																																																								
a5p-2006	1050	60	200																																																																								
a5p-2007	430	60	120																																																																								
pnz-2001	440	60	160																																																																								
pnz-2002	1060	60	80																																																																								
<b>25 mm</b> 2750 x 1830	<table border="1"> <thead> <tr> <th>Item</th> <th><math>l_i</math></th> <th><math>w_i</math></th> <th><math>b_i</math></th> </tr> </thead> <tbody> <tr><td>pnz-2501</td><td>430</td><td>60</td><td>80</td></tr> <tr><td>pnz-2502</td><td>500</td><td>60</td><td>160</td></tr> <tr><td>pnz-2503</td><td>1050</td><td>60</td><td>160</td></tr> </tbody> </table>	Item	$l_i$	$w_i$	$b_i$	pnz-2501	430	60	80	pnz-2502	500	60	160	pnz-2503	1050	60	160																																																										
Item	$l_i$	$w_i$	$b_i$																																																																								
pnz-2501	430	60	80																																																																								
pnz-2502	500	60	160																																																																								
pnz-2503	1050	60	160																																																																								

## A.2. O Modelo 5 na sintaxe do Xpress-Mosel

Aqui exibimos a implementação do Modelo 5, extraído de Yanasse e Morábito (2006), para a geração de padrões tabuleiro, no Xpress-Mosel.

```
#####  
model "Modelo5"  
  
! ### Bibliotecas  
  
uses "mmsystem", !  
     "mmive", !  
     "mmxslp", !  
     "mmxprs", ! pacote p/ resolver prob. de otmiz.  
     "mmodbc" ! pacote para ler o arquivo  
  
! ### Declaração das Sub-rotinas utilizadas  
  
forward procedure Le_Dados  
forward procedure Ordena_Itens  
forward procedure Gera_Padros  
forward procedure Export_Padrao  
forward procedure Gera_lGE  
  
! ### Parametrizações do ambiente de processamento  
  
declarations  
     TMPMAXPROC=3600 !\ Tempo máximo de processamento (em segundos)  
                    !/ para a geração de um padrão de corte  
end-declarations  
  
setparam("zerotol",1e-3)  
setparam("XPRS_presolve",1) ! Liga o presolver  
setparam("XPRS_cutstrategy",-1) ! Gera automaticamente os planos de corte  
setparam("XPRS_MAXTIME",TMPMAXPROC) ! Parametriza o tempo máximo de processamento  
  
! ### Declaração de variáveis globais da heurística  
  
declarations  
     nr_Itens: integer ! N° de Itens s/ rotação  
     nr_ItRot: integer ! N° de Itens c/ rotação  
  
     NmArqDados: string ! Nome do Arq. de dados a ser lido  
     NmArqSol: string ! Nome do Arq. de saída a ser lido  
     NmArqTxt: string ! Nome do Arq. exportação Defig7  
  
     I:range ! range do n° de Itens s/ Rotação  
     II:range ! range do n° de Itens c/ Rotação  
  
! Vetores para o armazenamento dos dados dos itens Originais, sem  
! rotação, lidos no arquivo de dados:  
It_Width: array(I) of integer ! Vetor Largura  
It_Lenght: array(I) of integer ! Vetor Comprimento  
It_Demanda: array(I) of integer ! Vetor Demanda  
It_Weigth: array(I) of integer ! Vetor Peso (ainda não implementado)  
  
! Vetores para o armazenamento dos dados dos itens após inclusão dos  
! itens rotacionados  
ItR_Width: array(II) of integer ! Vetor Largura  
ItR_Lenght: array(II) of integer ! Vetor Comprimento  
ItR_Demanda: array(II) of integer ! Vetor Demanda  
ItR_Weigth: array(II) of integer ! Vetor Peso (ainda não implementado)  
ItR_ParRot: array(II) of integer ! Vetor de Armaz. do Par Rotacionado  
ItR_ItOrig: array(II) of integer ! Vetor de Armaz. do Item Original  
vt_AuxDem: array(II) of integer ! Vetor Aux de Demanda dos Itens
```

```

L: integer           ! Comprimento do Objeto
W: integer           ! Largura do Objeto

Serra: integer       ! Desgaste da serra

Rot_Itens:integer    ! Flag de rotação dos itens:
                    ! 1-N Admitida 2-Admitida

Rot_Obj:integer      ! Flag de rotação dos Objetos:
                    ! 1-N Admitida 2-Admitida

Flg_Continua:integer ! Flag que indica se o processamento é
                    ! continuação do anterior

n:integer            ! Var. aux. para controle de laço.

! Variáveis utilizadas no controle da gerão dos padrões de corte
RFx:range            ! Range de Faixas
vf:integer           ! Ponteiro da Faixa
RPd:range            ! Range de Faixas
vp:integer           ! Ponteiro do Padrao
Mt_Padros: array(RPd,1..7) of integer ! Matriz de Padros
Mt_FxPadros:array(RPd,RFx,II) of integer ! Matriz das Faixas dos Padros
Mt_ItRPadros:array(RPd,II) of integer ! Matriz dos itens dos Padros
                    !(c/ rotação)
Mt_ItPadros:array(RPd,I) of integer ! Matriz dos itens dos Padros
                    !(s/ rotação)
Mt_SbPadros:array(RPd,1..2) of integer ! Matriz c/ as dimensoes das
                    ! Sobras dos Padros
vNr_Fx: integer      ! Nr. de faixas no padrão

end-declarations

! ### Início do Processo

Le_Dados             ! Leitura dos dados

Rot_Itens             ! Criação dos Itens Rotacionados

Ordena_Itens         ! Ordenação dos itens pela largura e comprimento

!#### Criação dos padrões

declarations
  H:range             ! Range de Padrões
  FW:range            ! Range de Faixas ao longo de W
  FL:range            ! Range de Faixas ao longo de W

  Nr_Lj:integer       ! N° de faixas verticais (Lj x W) do item i
  Nr_Wk:integer       ! N° de faixas horizontais (L x Wk) do item i

  hh:integer          ! Contador do nr de Padrões
end-declarations

Gera_Padros          ! Criação dos Padros

!## Saída da matriz de padros para o arq. " *.sol "
fopen(NmArqSol,F_APPEND)
  writeln("\n### matriz de padros")
  write(" It ")
  forall(p in RPd)
    write(" ",strfmt(p,5))
  write("\n")
  write("---- ")
  forall(p in RPd)
    write(" ----")
  write("\n")
  forall(i in I) do
    write("( ",strfmt(i,2),") ")
    forall(p in RPd)
      write(" ",strfmt(Mt_ItPadros(p,i),4))
    write("\n")
  end-do
fclose(F_APPEND)

```

```

!## Removendo a compensação do desgaste da serra

L -= Serra
W -= Serra

forall(i in II)do
    ItR_Width(i) -= Serra
    ItR_Lenght(i)-= Serra
end-do

!### Exporta Padrões para a rotina Defig7 (CorteBi)

Export_Padrao

! ### Fim do Processo

#####
Procedure Le_Dados
-----

Efetua a leitura dos arquivos de dados

#####

procedure Le_Dados

    ! Entrada do nome do arquivo de dados
    write("Informe o nome do Arquivo de Dados (s/ extensão): ")
    fflush
    readln(NmArqDados)

    ! Determinação dos nomes dos arquivos a serem utilizados
    NmArqDados := NmArqDados+".dat" ! Nome do Arq. de dados
    a
    ser lido
    NmArqSol := NmArqDados-".dat"+" .sol" ! Nome do Arq. de saída,
    para
    gravação da solução
    final
    NmArqTxt := NmArqDados-".dat"+" .txt" ! Nome do Arq. exportação
    Defig7

    ! Leitura dos dados iniciais
    initializations from NmArqDados
    nr_Itens; L; W; Serra
end-initializations

    ! Inicialização dos vetores com os dados dos itens
    forall(i in 1..nr_Itens)do
        It_Lenght(i):= 0
        It_Width(i) := 0
        It_Demand(i):= 0
        It_Weigth(i):= 0
    end-do

    ! Leitura dos itens
    initializations from NmArqDados
    [It_Lenght,It_Width,It_Demand,It_Weigth] as "vt_Itens"
end-initializations

!### Gravação em arquivo dos itens lidos
fopen(NmArqSol,F_OUTPUT)
writeln("Arquivo: ",NmArqDados,"\n\n")
writeln("Relação de Itens - Originais")
forall(i in I)
    writeln("Item ",strfmt(i,2),": ",strfmt(It_Width(i),4)," x ",

```

```

        strfmt(It_Lenght(i),4), " - ",strfmt(It_Demand(i),4))
    fclose(F_OUTPUT)

end-procedure

#####
Procedure Rot_Itens
-----

    Cria os itens rotacionados

#####

procedure Rot_Itens

    nr_ItRot := 2*nr_Itens ! Total de Itens, considerando Rotação

    ! Registra nos vetores ItR os itens originais e seus pares
    ! rotaciona dos
    forall(i in I)do

        ! Itens Originais
        ItR_Lenght(i) := It_Lenght(i)
        ItR_Width(i) := It_Width(i)
        ItR_Demand(i) := It_Demand(i)
        ItR_Weigth(i) := It_Weigth(i)
        ItR_ParRot(i) := i+nr_Itens
        ItR_ItOrig(i) := i

        ! Pares rotacionados
        ItR_Lenght(i+nr_Itens) := It_Width(i)
        ItR_Width(i+nr_Itens) := It_Lenght(i)
        ItR_Demand(i+nr_Itens) := 0
        ItR_Weigth(i+nr_Itens) := 0
        ItR_ParRot(i+nr_Itens) := i
        ItR_ItOrig(i+nr_Itens) := i
    end-do

    !## Compensação do desgaste da serra
    L += Serra
    W += Serra

    forall(i in II)do
        ItR_Width(i) += Serra
        ItR_Lenght(i) += Serra
    end-do

end-procedure

#####
Procedure Ordena_Itens
-----

    Ordena os vetores ItR

#####

procedure Ordena_Itens

    declarations
        vt_AuxOrder:array(1..6) of integer ! Vetor Aux. P/ ordenação
                                           ! dos Itens
    end-declarations

    ! Processo de ordenação
    forall(ii in II) do

        ! A comparação será feita do último item até o item corrente
        n := nr_ItRot

```

```

while (n > ii) do

    ! Critério para ordenação dos itens
    !(1º menor largura; 2º menor comprimento)
    if((ItR_Width(ii)<ItR_Width(n)) or
       ((ItR_Width(ii)=ItR_Width(n)) and
        (ItR_Lenght(ii)<ItR_Lenght(n))))
    then

        ! Vt_AuxOrder recebe os dados do item corrente (ii)
        vt_AuxOrder(1) := ItR_Width(ii)
        vt_AuxOrder(2) := ItR_Lenght(ii)
        vt_AuxOrder(3) := ItR_Demand(ii)
        vt_AuxOrder(4) := ItR_Weigth(ii)
        vt_AuxOrder(5) := ItR_ParRot(ii)
        vt_AuxOrder(6) := ItR_ItOrig(ii)

        !Efetua a troca dos ponteiros para o par rotacionado
        if(ItR_ParRot(ii)<>n)then
            tR_ParRot(ItR_ParRot(ii)):= n           I-
            tR_ParRot(ItR_ParRot(n)):= ii           I-
            tR_ParRot(ii):=ItR_ParRot(n)
            ItR_ParRot(n):= vt_AuxOrder(5)
        end-if

        ! O item corrente passa a ser o n-ésimo item
        ItR_Width(ii) := ItR_Width(n)
        ItR_Lenght(ii) := ItR_Lenght(n)
        ItR_Demand(ii) := ItR_Demand(n)
        ItR_Weigth(ii) := ItR_Weigth(n)
        ItR_ItOrig(ii) := ItR_ItOrig(n)

        ! O n-ésimo item passa a ser o item corrente
        ItR_Width(n) := vt_AuxOrder(1)
        ItR_Lenght(n) := vt_AuxOrder(2)
        ItR_Demand(n) := vt_AuxOrder(3)
        ItR_Weigth(n) := vt_AuxOrder(4)
        ItR_ItOrig(n) := vt_AuxOrder(6)

    end-if
    n -= 1

end-do

end-do

!## ! Gravação em arquivo dos itens com rotação, ordenados
fopen(NmArqSol,F_APPEND)
writeln("\nRelação de Itens c/ Rotação - Pós-Ordenação")
forall(ii in II)
    writeln("Iem ",strfmt(ii,2),": ",strfmt(ItR_Width(ii),4),
           " x ",strfmt(ItR_Lenght(ii),4), " - ",
           strfmt(ItR_Demand(ii),4)," - ",
           strfmt(ItR_ParRot(ii),2)," - ",
           strfmt(ItR_ItOrig(ii),2))
fclose(F_APPEND)

end-procedure

#####
Procedure Gera_Padros
-----

Cria até cinco padrões para cada faixa registrada na matriz
Mt_Faixas
- O primeiro toma por base um padrão tabuleiro gerado para a
  faixa selecionada;
- Os demais são padrões tomam por base padrões derivados do
  padrão tabuleiro, pela exclusão de algumas faixas
#####

```

```
procedure Gera_Padros
```

```
! ### Gerando o Padrão
```

```
vp:=1
```

```
!### Declaração das variáveis
```

```
declarations
```

```
  vNrMax_Lj: integer      ! Qtde de faixas Lj x W
```

```
  vNrMax_Wk: integer      ! Qtde de faixas Wk x L
```

```
  vAux: integer           ! Var. Auxiliar
```

```
  tempo_Proc: real
```

```
end-declarations
```

```
!# Determinação do nr max de faixas que poderão ser construída
```

```
if(Rot_Itens=1)then
```

```
  vAux := min(i in II)(ItR_Length(i))
```

```
  vNrMax_Lj := (L div vAux)+1
```

```
  vAux := min(i in II)(ItR_Width(i))
```

```
  vNrMax_Wk := (W div vAux)+1
```

```
else
```

```
  vAux := min(i in II)(ItR_Length(i))
```

```
  vNrMax_Lj := (L div vAux)+1
```

```
  vNrMax_Wk := (W div vAux)+1
```

```
end-if
```

```
!# Declaração das Variáveis
```

```
declarations
```

```
  vJ = 1..vNrMax_Lj      ! Ponteiro para as faixas Lj x W (1<=j<=P)
```

```
  vK = 1..vNrMax_Wk      ! Ponteiro para as faixas Wk x L (1<=k<=Q)
```

```
  vWk: array(vK) of mpvar      ! Vetor contendo:
```

```
! -> A larg. das faixas (L x Wk)
```

```
! -> Na última posição armazena a
```

```
! larg. da faixa de sobra
```

```
  vLj: array(vJ) of mpvar      ! Vetor contendo:
```

```
! -> O compr. das faixas (Lj x W)
```

```
! -> Na última posição armazena a
```

```
! compr. da faixa de sobra
```

```
  vX: array(vK,vJ,II) of mpvar ! Var. binária, /1-se o item ii está em  $L_j \times W_k$ 
```

```
! \0- caso contrário
```

```
  vZ: linctr
```

```
! Função Objetivo
```

```
end-declarations
```

```
!### Determinação do tipo da Variável #####
```

```
forall(k in vK)
```

```
  forall(j in vJ)
```

```
    forall(ii in II)
```

```
      vX(k,j,ii) is_binary;
```

```
forall(j in vJ) do
```

```
  vLj(j) is_integer
```

```
end-do
```

```
forall(k in vK) do
```

```
  vWk(k) is_integer;
```

```
end-do
```

```
!### Declarando a função objetivo #####
```

```
vZ := sum(i in II,j in vJ,k in vK)(vX(k,j,i)*(ItR_Width(i)-Serra)  
                                     *(ItR_Length(i))-Serra)
```

```

!### Declarando as restrições #####

! Restrições físicas do objeto
restr_12 := sum(j in vJ)(vLj(j)) <= L;
restr_13 := sum(k in vK)(vWk(k)) <= W;

! Garante a alocação de apenas um item no retângulo  $L \times W$ 
forall(k in vK)
  forall(j in vJ)
    restr_14(k,j) := sum(i in II)(vX(k,j,i)) <= 1;

! Garante que o item alocação em  $L \times W$  tenha comprimento  $L$ 
forall(k in vK)
  forall(j in vJ)do
    restr_15a(k,j):=sum(i in II)(vX(k,j,i)*ItR_Lenght(i)) <=
      vLj(j)+L*(1-sum(i in II)(vX(k,j,i)))
    restr_15b(k,j):=vLj(j)<=sum(i in II)(vX(k,j,i)*ItR_Lenght(i))
      +L*(1-sum(i in II)(vX(k,j,i)))
  end-do

! Garante que o item alocação em  $L \times W$  tenha largura  $W$ 
forall(k in vK)
  forall(j in vJ) do
    restr_16a(k,j):=sum(i in II)(vX(k,j,i)*ItR_Width(i)) <=
      vWk(k)+W*(1-sum(i in II)(vX(k,j,i)))
    restr_16b(k,j):=vWk(k)<=sum(i in II)(vX(k,j,i))
      *ItR_Width(i))+W*(1-sum(i in II)(vX(k,j,i)))
  end-do

! Ordena as faixas em  $L$ 
forall(j in vJ)
  if(j < vNrMax_Lj) then
    restr_18(j):= vLj(j) - vLj(j+1) >= 0;
  end-if

! Ordena as faixas em  $W$ 
forall(k in vK)
  if(k < vNrMax_Wk)then
    restr_19(k):= vWk(k) - vWk(k+1) >= 0;
  end-if

! Restições de não-negatividade
forall(j in vJ) do
  restr_20a(j):= vLj(j) >= 0
end-do

forall(k in vK) do
  restr_20b(k):= vWk(k) >= 0
end-do

!# Resolução do Problema
tempo_Proc:= gettime
maximize(XPRS_DUAL+XPRS_PRI,vZ)
tempo_Proc:= gettime-tempo_Proc

!### Atualização a Matriz de Padrões de Corte

! Dimensoes do Padrao
Mt_Padroes(vp,1) := W
Mt_Padroes(vp,2) := L

! Flag de tipo do Padrao (Não implementado)
Mt_Padroes(vp,5) := 1

```

```

! Flag de Orientação do Obj (1-Ñ Rotacionado 2-Rotacionado
! (Não implementado)
Mt_Padros(vp,6) := 1

! Tempo de processamento do padrão
Mt_Padros(vp,7) := integer(tempo_Proc)

! Montagem da Matriz de Faixas do Padrao (Mt_FxPadros) e
! da Matriz de Itens dos Padros (Mt_ItPadros)
kk:=0

forall(k in vK)

  if(sum(j in vJ,i in II)(getsol(vX(k,j,i)))>0)then
    kk+=1
    forall(j in vJ)
    forall(i in II)do
      Mt_FxPadros(vp,kk,i)+= (round(getsol(vX(k,j,i))))
      Mt_ItRPadros(vp,i) += (round(getsol(vX(k,j,i))))
      Mt_ItPadros(vp,ItR_ItOrig(i))+=(round(getsol(vX(k,j,i))))
    end-do
  end-if

! Perda Toral
Mt_Padros(vp,3):=round((1-(sum(i in I)(Mt_ItPadros(vp,i)
*It_Lenght(i)*It_Width(i)) )
/((Mt_Padros(vp,1)-Serra)
*(Mt_Padros(vp,2)-Serra)))*10000)

! Perda p/ Sobra
Mt_Padros(vp,4):= round((1-(sum(i in I)(Mt_ItPadros(vp,i)
*(It_Lenght(i)+Serra)*(It_Width(i)+Serra)))
/(Mt_Padros(vp,1)*Mt_Padros(vp,2)))*10000)

end-procedure

#####
Procedure Exporta_Padros
-----

Realiza a exportação dos dados dos padrões para que possa ser
desenhado no programa Defig7, desenvolvido por Tiago ?????
(2004)

#####

procedure Exporta_Padros

forall(p in RPd)do
  if(p<10) then
    flag := "00"
  elif(p<100)then
    flag := "0"
  else
    flag := ""
  end-if

  fopen(NmArqTxt-" .txt "+ "-" +flag+strfmt(p,1)+".txt",F_OUTPUT)
  writeln(NmArqTxt-" .txt "+ "-" +flag+strfmt(p,1))
  if(Mt_Padros(p,6)=1)then
    writeln(L," ",W)
  else
    writeln(W," ",L)
  end-if

  writeln(Serra)
  writeln(strfmt(Mt_Padros(p,3)/100,6,2))
  writeln(2*nr_Itens)

```

```

forall(i in II)
  writeln(ItR_Lenght(i)," ",ItR_Width(i))

vNrFaixas:=0
forall(f in RPF)
  if(sum(i in II)(Mt_FxPadroes(p,f,i ))>0) then
    vNrFaixas += 1
  end-if

writeln(vNrFaixas)
forall(f in RPF)do
  if(sum(i in II)(Mt_FxPadroes(p,vNrFaixas-f+1,i))>0) then
    write(1," ")
    forall(i in II)
      write(strfmt(Mt_FxPadroes(p,vNrFaixas-f+1,i),3)," ")
    write("\n")
  end-if
end-do
fclose(F_OUTPUT)
end-do

end-procedure

```

end-model

### A.3. A Heurística 2 na sintaxe do Xpress-Mosel

A Heurística 2 efetua o atendimento a demanda, utilizando um pool de padrões tabuleiros compostos criados a partir do procedimento de Gilmore e Gomory (1965) para a criação de padrões de corte em dois estágios

#####

model "Heuristica2"

```

! ### Bibliotecas

uses "mmsystem", !
     "mmive", !
     "mmxslp", !
     "mmxprs", ! pacote p/ resolver prob. de otmiz.
     "mmodbc" ! pacote para ler o arquivo

! ### Declaração das Sub-rotinas utilizadas

forward procedure Le_Dados
forward procedure Rot_Itens
forward procedure Ordena_Itens
forward procedure Gera_Faixas
forward procedure Compl_Faixas_1(vFw:integer,vL:integer, vItem:integer,
                                Qtd_Item:integer,v:integer)
forward procedure Compl_Faixas_2(vf:integer)
forward procedure Gera_Padroes
forward procedure Com-
                                pl_Padroes(vFx:integer,vQtd_vFx:integer,
                                r,vFlag:integer)
forward procedure Exporta_Padroes

! ### Parametrizações do ambiente de processamento

declarations
  TMPMAXPROC_1=600 !\ Tempo máximo de processamento (em segundos)
                  !/ para a geração de um padrão de corte

```

```

TMPMAXPROC_2=2500          !\ Tempo máximo de processamento para cálculo
                          !/ do atendimento a demanda

Tempo_proc1:real          !\ Variáveis auxiliares usadas para determinação
Tempo_proc2:real          !/ do tempo de processamento da Heurística

end-declarations

setparam("XPRS_presolve",1) ! Liga o presolver
setparam("XPRS_cutstrategy",-1) ! Gera automaticamente os planos de corte
setparam("XPRS_MAXTIME",TMPMAXPROC_1)! Parametriza o tempo máximo de processamento

! ### Declaração de variáveis globais da heurística

declarations
  nr_Itens: integer        ! N° de Itens s/ rotação
  nr_ItRot: integer        ! N° de Itens c/ rotação

  NmArqDados: string      ! Nome do Arq. de dados a ser lido
  NmArqSol: string        ! Nome do Arq. de saída a ser lido
  NmArqTxt: string        ! Nome do Arq. exportação Defig7

  I:range                 ! range do n° de Itens s/ Rotação
  II:range                ! range do n° de Itens c/ Rotação

  ! Vetores para o armazenamento dos dados dos itens Originais, sem
  ! rotação, lidos no arquivo de dados:
  It_Width: array(I) of integer ! Vetor Largura
  It_Lenght: array(I) of integer ! Vetor Comprimento
  It_Demand: array(I) of integer ! Vetor Demanda
  It_Weigth: array(I) of integer ! Vetor Peso (não implementado)

  ! Vetores para o armazenamento dos dados dos itens após inclusão dos
  ! itens rotacionados
  ItR_Width: array(II) of integer ! Vetor Largura
  ItR_Lenght: array(II) of integer ! Vetor Comprimento
  ItR_Demand: array(II) of integer ! Vetor Demanda
  ItR_Weigth: array(II) of integer ! Vetor Peso (não implementado)
  ItR_ParRot: array(II) of integer ! Vetor Armaz. Par Rotacionado
  ItR_ItOrig: array(II) of integer ! Vetor Armaz. Item Original

  L: integer              ! Comprimento do Objeto
  W: integer              ! Largura do Objeto

  Serra: integer         ! Desgaste da serra

  ! Variáveis utilizadas no controle da geração das faixas
  RFx:range              ! Range de Faixas
  vf:integer             ! Ponteiro da Faixa
  Mt_Faixas: array(RFx,1..6) of integer! Matriz de faixas
  Mt_ItFaixas:array(RFx,II) of integer! Matriz de itens das Faixas
  Mt_SbFaixas:array(RFx,1..2) of integer! Matriz c/ as dimensoes das
  ! Sobras das Faixas

  ! Variáveis utilizadas no controle da geração dos padrões de corte
  RPD:range              ! Range de Padrões de corte
  RPF:range              ! Range de Faixas dos Padroes
  vp:integer             ! Ponteiro do Padrao
  vQtd_Fx:integer        ! Nr. de faixas no padrão
  Mt_Padroes: array(RPD,1..7) of integer ! Matriz de Padroes
  Mt_FxPadroes:array(RPD,RPF,II) of integer ! Matriz das Faixas dos Padroes
  Mt_CutPadroes:array(RPD,II) of integer ! Matriz de Corte dos Padroes
  Mt_ItPadroes: array(RPD,I) of integer ! Matriz dos itens dos Padroes
  Mt_SbPadroes: array(RPD,1..2) of integer ! Matriz c/ as dimensões
  ! das Sobras dos Padroes

end-declarations

! ### Início do Processo

Le_Dados                  ! Leitura dos dados

Tempo_proc1:=gettime     ! Registro do início do processo

```

```

Rot_Itens          ! Criação dos Itens Rotacionados
Ordena_Itens      ! Ordenação dos itens pela largura e comprimento
Gera_Faixas      ! Criação das faixas
Gera_Padros      ! Criação dos Padros
Atend_Demanda    ! Resolve o Probl de atendimento a demanda

! ### Fim do Processo

```

```

#####
Procedure Le_Dados
-----

```

Efetua a leitura dos arquivos de dados

```

#####

```

```

procedure Le_Dados

```

```

! Entrada do nome do arquivo de dados
write("Informe o nome do Arquivo de Dados (s/ extensão): ")
fflush
readln(NmArqDados)

```

```

! Determinação dos nomes dos arquivos a serem utilizados
NmArqDados := NmArqDados+".dat" ! Nome do Arq. de dados
a
! ser lido
NmArqSol := NmArqDados-".dat"+" .sol" ! Nome do Arq. de saída,
para
! gravação da solução
final
NmArqTxt := NmArqDados-".dat"+" .txt" ! Nome do Arq. exportação
Defig7

```

```

! Leitura dos dados iniciais
initializations from NmArqDados
nr_Itens; L; W; Serra
end-initializations

```

```

! Inicialização dos vetores com os dados dos itens
forall(i in 1..nr_Itens)do
It_Lenght(i) := 0
It_Width(i) := 0
It_Demand(i) := 0
It_Weigth(i) := 0
end-do

```

```

! Leitura dos itens
initializations from NmArqDados
[It_Lenght, It_Width, It_Demand, It_Weigth] as "vt_Itens"
end-initializations

```

```

!### Gravação em arquivo dos itens lidos
fopen(NmArqSol, F_OUTPUT)
writeln("Arquivo: ", NmArqDados, "\n\n")
writeln("Relação de Itens - Originais")
forall(i in I)
writeln("Item ", strfmt(i, 2), ": ", strfmt(It_Width(i), 4), " x ",
strfmt(It_Lenght(i), 4), " - ", strfmt(It_Demand(i), 4))
fclose(F_OUTPUT)

```

```

end-procedure
#####
Procedure Rot_Itens
-----

  Cria os itens rotacionados

#####

procedure Rot_Itens

  nr_ItRot := 2*nr_Itens ! Total de Itens, considerando Rotação

  ! Registra nos vetores ItR os itens originais e seus pares
  ! rotaciona dos
  forall(i in I)do

    ! Itens Originais
    ItR_Lenght(i) := It_Lenght(i)
    ItR_Width(i) := It_Width(i)
    ItR_Demand(i) := It_Demand(i)
    ItR_Weigth(i) := It_Weigth(i)
    ItR_ParRot(i) := i+nr_Itens
    ItR_ItOrig(i) := i

    ! Pares rotacionados
    ItR_Lenght(i+nr_Itens) := It_Width(i)
    ItR_Width(i+nr_Itens) := It_Lenght(i)
    ItR_Demand(i+nr_Itens) := 0
    ItR_Weigth(i+nr_Itens) := 0
    ItR_ParRot(i+nr_Itens) := i
    ItR_ItOrig(i+nr_Itens) := i
  end-do

  !## Compensação do desgaste da serra
  L += Serra
  W += Serra

  forall(i in II)do
    ItR_Width(i) += Serra
    ItR_Lenght(i) += Serra
  end-do

end-procedure

#####
Procedure Ordena_Itens
-----

  Ordena os vetores ItR

#####

procedure Ordena_Itens

  declarations
    vt_AuxOrder:array(1..6) of integer ! Vetor Aux. P/ ordenação
                                         ! dos Itens
  end-declarations

  ! Processo de ordenação
  forall(ii in II) do

    ! A comparação será feita do último item até o item corrente
    n := nr_ItRot
    while (n > ii) do

```

```

! Critério para ordenação dos itens
!(1º menor largura; 2º menor comprimento)
if((ItR_Width(ii)<ItR_Width(n)) or
  ((ItR_Width(ii)=ItR_Width(n)) and
  (ItR_Lenght(ii)<ItR_Lenght(n))))
then

    ! Vt_AuxOrder recebe os dados do item corrente (ii)
    vt_AuxOrder(1) := ItR_Width(ii)
    vt_AuxOrder(2) := ItR_Lenght(ii)
    vt_AuxOrder(3) := ItR_Demand(ii)
    vt_AuxOrder(4) := ItR_Weigth(ii)
    vt_AuxOrder(5) := ItR_ParRot(ii)
    vt_AuxOrder(6) := ItR_ItOrig(ii)

    !Efetua a troca dos ponteiros para o par rotacionado
    if(ItR_ParRot(ii)<>n)then
        tR_ParRot(ItR_ParRot(ii)):= n           I-
        tR_ParRot(ItR_ParRot(n)):= ii          I-
        tR_ParRot(ii):=ItR_ParRot(n)
        ItR_ParRot(n):= vt_AuxOrder(5)
    end-if

    ! O item corrente passa a ser o n-ésimo item
    ItR_Width(ii) := ItR_Width(n)
    ItR_Lenght(ii) := ItR_Lenght(n)
    ItR_Demand(ii) := ItR_Demand(n)
    ItR_Weigth(ii) := ItR_Weigth(n)
    ItR_ItOrig(ii) := ItR_ItOrig(n)

    ! O n-ésimo item passa a ser o item corrente
    ItR_Width(n) := vt_AuxOrder(1)
    ItR_Lenght(n) := vt_AuxOrder(2)
    ItR_Demand(n) := vt_AuxOrder(3)
    ItR_Weigth(n) := vt_AuxOrder(4)
    ItR_ItOrig(n) := vt_AuxOrder(6)

end-if
n -= 1

end-do

end-do

!## ! Gravação em arquivo dos itens com rotação, ordenados
fopen(NmArqSol,F_APPEND)
writeln("\nRelação de Itens c/ Rotação - Pós-Ordenação")
forall(ii in II)
    writeln("Iem ",strfmt(ii,2),": ",strfmt(ItR_Width(ii),4),
    " x ",strfmt(ItR_Lenght(ii),4), " - ",
    strfmt(ItR_Demand(ii),4)," - ",
    strfmt(ItR_ParRot(ii),2)," - ",
    strfmt(ItR_ItOrig(ii),2))
fclose(F_APPEND)

end-procedure

#####
Procedure Gera_Faixas
-----

Cria até duas faixas para cada item informado no arquivo de dados
- Uma com a quantidade máxima do item considerado
- Outra com apenas um item

#####

```

```

procedure Gera_Faixas

vf:=0 ! Inicialização do ponteiro de faixa

forall(v in 1..2)do ! Controla a rotação do objeto
  if(v=2)then ! v=1 --> Cria faixas para o objeto LxW
    vAux := W ! v=2 --> Cria faixas para o objeto WxL
    W := L
    L := vAux
  end-if

  ! Construção de faixas com itens de mesma largura
  forall(ii in II) do

    ! Constroi uma faixa com a qtde máxima do item ii
    ! {floor(L/ItR_Lenght(ii))}, complementando a sobra com
    ! outros itens, se possível
    Compl_Faixas_1(ItR_Width(ii),L,ii,floor(L/ItR_Lenght(ii)),v)

    ! Se a faixa anterior possuir mais de um item ii, contruimos
    ! uma nova faixa com apenas um item ii, complementando a
    ! sobra com outros itens
    if(floor(L/ItR_Lenght(ii))>1 and
      ItR_Width(ii)=ItR_Width(ii+1))then

      Compl_Faixas_1(ItR_Width(ii),L,ii,1,v)

    end-if

  end-do

  ! Desfaz a rotação do objeto
  if(v=2)then
    vAux := W
    W := L
    L := vAux
  end-if

end-do

forall(ff in RFx)do

  !Calculo das dimensoes da Sobra da Faixa
  Mt_SbFaixas(ff,1):=Mt_Faixas(ff,1)
  Mt_SbFaixas(ff,2):=Mt_Faixas(ff,2)
  -sum(i in II)(Mt_ItFaixas(ff,i)*ItR_Lenght(i))

  ! Complemento das faixas com itens de larguras menores
  Compl_Faixas_2(ff)

  !Calculo da Perda total da Faixa
  Mt_Faixas(ff,3):=Mt_Faixas(ff,1)*Mt_Faixas(ff,2)-sum(i in II)
  (Mt_ItFaixas(ff,i)
  *(It_Lenght(ItR_ItOrig(i))-Serra)
  *(It_Width(ItR_ItOrig(i))-Serra))

  ! Calculo da Perda p/ sobra da Faixa
  Mt_Faixas(ff,4):=Mt_SbFaixas(ff,1)*Mt_SbFaixas(ff,2)

end-do

end-procedure

#####
Procedure Compl_Faixas_1
-----

```

Efetua o complemento exato da faixa, ou seja, com outros itens de mesma largura

Parâmetros:

VFw -->Largura da Faixa  
 vL -->Comprimento da Faixa  
 vItem -->Item pivo da faixa, ou item base para a formação da faixa  
 Qtd\_Item--> Quantidade do item pivo na faixa  
 v--> Indica para que tipo de objeto (LxW ou WxL) a faixa pertence

#####

```
procedure
  Compl_Faixas_1(vFw:integer,vL:integer,vItem:integer,Qtd_Item:integer,
    v:integer)
```

!### Declaração das variáveis

```
declarations
  vX:array(II) of mpvar ! Variável indicando a quantidade
dos
  vY:array(II) of mpvar ! Variável binária indicando se o
item
  vZ:lincstr ! Função objetivo
end-declarations
```

!### Determinação do tipo da Variável #####

```
forall(i in II)do
  vX(i) is_integer
  vY(i) is_binary
  if(i=vItem)then
    vX(i)<=Qtd_Item
    vX(i)>=Qtd_Item
    vY(i)>=1
    vY(i)<=1
  elif(ItR_Width(i)<>vFw)then
    vX(i)<=0
    vX(i)>=0
    vY(i)<=0
    vY(i)>=0
  else
    vX(i)>=0
    vX(i)<=It_Demand(ItR_ItOrig(i))
  end-if
end-do
```

!### Declarando a função objetivo #####  
 vZ := sum(i in II)(vX(i)\*ItR\_Lenght(i)\*ItR\_Width(i));

!### Declarando as restrições #####

! 1ª Restrição: Restrição física da faixa  
 restr1 := sum(i in II)(vX(i)\*ItR\_Lenght(i)) <= vL;

! 2ª Restrição: Limita o nº de itens diferentes na faixa a 2  
 restr2 := sum(i in II)(vY(i))<=2

! 3ª Restrição: Impõe que a variável vX seja Zero,  
 ! se o item não for utilizado na faixa (vY=0)  
 forall(i in II)  
 restr3(i) := vX(i) <= vL\*vY(i)

!### Resolução do problema:

! Max vZ  
 ! S.a: restr1

```

!      restr2
!      restr3
maximize(XPRS_DUAL+XPRS_PRI,vZ);

!### Atualização a Matriz de Faixas
vf+=1
Mt_Faixas(vf,1) := vFw      ! Largura da Faixa
Mt_Faixas(vf,2) := L        ! Comprimento da Faixa
Mt_Faixas(vf,3) := 0        ! Perda total da Faixa \ (será calculado
Mt_Faixas(vf,4) := 0        ! Sobra da Faixa / depois)
Mt_Faixas(vf,5) := vItem    ! Item Pivo da faixa
Mt_Faixas(vf,6) := v        ! 1 - Obj Nao Rotacionado;
                                ! 2 - Obj Rotacionado
forall(i3 in II)
    Mt_ItFaixas(vf,i3) := integer(getsol(vX(i3)))
end-procedure

#####
Procedure Compl_Faixas_2
-----

Efetua o complemento não-exato da faixa, ou seja, com outros itens de larguras menores que a largura da faixa. A rotina é executada para a área de sobra da faixa, registrada na matriz Mt_SbFaixas

Parâmetros:
    vf --> Ponteiro indicando a faixa corrente na matriz Mt_SbFaixas

#####

procedure Compl_Faixas_2(vf:integer)

!### Declaração das variáveis
declarations
    vX:array(II) of mpvar
    vY:array(II) of mpvar
    vZ:linctr
end-declarations

!### Determinação do tipo da Variável #####
forall(i in II)do
    vX(i) is_integer
    vX(i) >= 0
    vY(i) is_binary
end-do

!### Declarando a função objetivo #####
vZ := sum(i in II)(vX(i)*ItR_Lenght(i)*ItR_Width(i));

!### Declarando as restrições #####

! 1ª Restrição: Restrição física da faixa
restr_01:=sum(i in
II)(vX(i)*ItR_Lenght(i))<=Mt_SbFaixas(vf,2);

forall(i in II)do

! 2ª Restrição: Somente os itens com larguras menores, ou
! igual, a largura da faixa de sobra podem ser utilizados
restr_02(i) := vY(i)*ItR_Width(i)<= Mt_SbFaixas(vf,1)

! 3ª Restrição: Impõe que a variável vX seja Zero,
! se o item não for utilizado na faixa (vY=0)
restr_03(i) := vX(i)<= vY(i)*Mt_SbFaixas(vf,2)

```

```

end-do

!### Critério de otimização #####
maximize(XPRS_DUAL+XPRS_PRI,vZ);

!### Atualização a Matriz de Faixas
forall(i in II)do
    Mt_ItFaixas(vf,i)+= integer(getsol(vX(i)))
end-do

!Calculo das dimensoes da Sobra da Faixa
Mt_SbFaixas(vf,2):=Mt_Faixas(vf,2)
                    -sum(i in
II)(Mt_ItFaixas(vf,i)*ItR_Lenght(i))

!Calculo da Perda total da Faixa
Mt_Faixas(vf,3):=Mt_Faixas(vf,1)*Mt_Faixas(vf,2)
                    -sum(i in II)(Mt_ItFaixas(vf,i)
                    *(It_Lenght(ItR_ItOrig(i))-Serra)
                    *(It_Width(ItR_ItOrig(i))-Serra))

!Calculo da Perda p/ sobra da Faixa
Mt_Faixas(vf,4):=Mt_SbFaixas(vf,1)*Mt_SbFaixas(vf,2)

end-procedure

#####
Procedure Gera_Padroes
-----

Cria até cinco padrões para cada faixa registrada na matriz
Mt_Faixas
- O primeiro toma por base um padrão tabuleiro gerado para a
  faixa selecionada;
- Os demais são padrões tomam por base padrões derivados do
  padrão tabuleiro, pela exclusão de algumas faixas

#####

procedure Gera_Padroes

declarations
    vAux:integer          ! Var. Auxiliar
    vWfx_Min1: integer    ! Largura da menor faixa
    vWfx_Min2: integer    ! Largura da menor faixa
    vIt_Pivo: integer     ! Item Pivo: 1º Item da faixa
    A: string
end-declarations

vp:=0                    ! Inicialização do ponteiro de padrão
forall(f in Rfx)do
    ! Testa a demanda do Item pivo da faixa, só construirá padrões
    ! de corte para itens com demanda maiores que zero
    if(It_Demand(ItR_ItOrig(Mt_Faixas(f,5)))>0)then

        !Verifica se a faixa é para um objeto LxW ou WxL
        if(Mt_Faixas(f,6)=1)then

            vQtd_Fx:=integer(W/Mt_Faixas(f,1)) ! Nr. de faixas
            vW := W
            vL := L
        else

```

```

vQtd_Fx:=integer(L/Mt_Faixas(f,1)) ! Nr. de faixas
vW := L
vL := W
end-if

! Identificação do Item Pivo, item base da faixa
vIt_Pivo := Mt_ItFaixas(f,1)

!Determinação da faixa de largura mínima
! Considerando apenas as faixas com pivo com demanda >0
vWfx_Min1 := min(ff in Rfx | Mt_Faixas(ff,5)<>vIt_Pivo and
                    (It_Demand(ItR_ItOrig(Mt_Faixas(ff,5)))>0))
                    (Mt_Faixas(ff,1))

! Considerando todas as faixas
vWfx_Min2 := min(ff in Rfx | Mt_Faixas(ff,5)<>vIt_Pivo)
                    (Mt_Faixas(ff,1))

!Determinação do passo de variação das faixas
passo := ceil(vQtd_Fx/5)
while(vQtd_Fx >0)do

    !### Atualização a Matriz de Padrões de Corte
    vp+=1
    Mt_Padroes(vp,1):=vW ! Largura da Faixa
    Mt_Padroes(vp,2):=vL ! Comprimento da
    Faixa
    Mt_Padroes(vp,5):=Mt_Faixas(f,5) ! Item Pivo do Pa-
    drao
    Mt_Padroes(vp,6):=Mt_Faixas(f,6) !\1-Obj Nao Rota-
    cionado
    !/2-Obj Rotacionado

    ! Montagem da Matriz de Faixas do Padrao (Mt_FxPadroes)
    ! e da Matriz de Itens dos Padroes (Mt_ItPadroes)
    forall(vNr_Fx in 1..vQtd_Fx)
        forall(i in II)do
            Mt_FxPadroes(vp,vNr_Fx,i) += Mt_ItFaixas(f,i)
            Mt_ItRPadroes(vp,i) += Mt_ItFaixas(f,i)
            Mt_ItPadroes(vp,ItR_ItOrig(i)) +=
            Mt_ItFaixas(f,i)
        end-do

        Mt_SbPadroes(vp,1):=Mt_Padroes(vp,1)-
        vQtd_Fx*Mt_Faixas(f,1)
        Mt_SbPadroes(vp,2) := Mt_Padroes(vp,2)

        vf := 0 ! inicializa o contador para novas faixas

        if(Mt_SbPadroes(vp,1)>= vWfx_Min1 )then
            Compl_Padroes(f,vQtd_Fx,1)
        end-if

        if(Mt_SbPadroes(vp,1)>= vWfx_Min2 )then
            Compl_Padroes(f,vQtd_Fx,2)
        end-if

        vQtd_Fx -= passo
    end-do
end-if
end-do

forall(p in RPd)do
    !Calculo da Perda total do Padrao
    Mt_Padroes(p,3):=integer((1-(sum(i in I)(Mt_ItPadroes(p,i)
        *(It_Lenght(i)-Serra)
        *(It_Width(i)-Serra))
        /((Mt_Padroes(p,1)-Serra)

```

```

*(Mt_Padros(p,2)-Serra))) *10000)

!Calculo da Perda p/ sobra do Padrao
Mt_Padros(p,4):=integer((1-(sum(i in I)(Mt_ItPadros(p,i)
*(It_Lenght(i))*It_Width(i))))
/(Mt_Padros(p,1)*Mt_Padros(p,2)))
*10000)

end-do

end-procedure

#####
Procedure Compl_Padros
-----

Efetua o complemento dos padrões dos padrões de corte criados,
reduzindo a área de sobra

#####
procedure Compl_Padros(vFcx:integer,vQtd_vFcx:integer,vFlag:integer)

!### Declaração das variáveis
declarations
  vAux: integer ! Var. Auxiliar
  vX:array(RFcx) of mpvar ! Var. indicando a qtde de vezes
que a
  vY:array(RFcx) of mpvar ! faixa aparece no padrão
! Var. binária indicando se a faixa foi
! utilizada no padrão
  vZ:linctr ! Função objetivo
end-declarations

!### Determinação do tipo da Variável #####
forall(ff in RFcx)do
  vX(ff) is_integer
  vY(ff) is_integer
  if(Mt_Faixas(ff,6)<>Mt_Faixas(vFcx,6) or ff=vFcx)then
    vX(ff)>=0
    vX(ff)<=0
    vY(ff)>=0
    vY(ff)<=0
  else
    if(vFlag=1 and
It_Demand(ItR_ItOrig(Mt_Faixas(ff,5)))=0)then
      vX(ff)>=0
      vX(ff)<=0
      vY(ff)>=0
      vY(ff)<=0
    else
      vX(ff)>=0
      vY(ff)>=0
      vY(ff)<=1
    end-if
  end-if
end-do

!### Declarando a função objetivo #####
vZ:=sum(ff in RFcx)(vX(ff)*sum(i in II)(Mt_ItFaixas(ff,i)
*ItR_Lenght(i)*ItR_Width(i)))

!### Declarando as restrições #####

! 1ª Restrição: Restrição física do padrão
restr1:=sum(ff in
RFcx)(vX(ff)*Mt_Faixas(ff,1))<=Mt_SbPadros(vp,1)

! 2ª Restrição: Só permite o uso de uma faixa
restr2:=sum(ff in RFcx)(vY(ff))<=1

```

```

! 3ª Restrição: Zera a quantidade das faixas não utilizadas
forall(ff in RFx)
  restre3(ff) := vX(ff) <= 1000*vY(ff)

!### Resolução do Problema
tempo_Proc:= gettime ! Tempo de início
maximize(XPRS_DUAL+XPRS_PRI,vZ) !
tempo_Proc:= gettime-tempo_Proc ! Tempo final

!### Atualização da Matriz de Padroes

forall(ff in RFx)do
  forall(k in 1..integer(getsol(vX(ff))))do
    Mt_SbPadroes(vp,1) -= Mt_Faixas(ff,1)
    vf += 1
    forall(i in II)do
      Mt_FxPadroes(vp,vQtd_vFx+vf,i) := Mt_ItFaixas(ff,i)
      Mt_ItRPadroes(vp,i) += Mt_ItFaixas(ff,i)
      Mt_ItPadroes(vp,ItR_ItOrig(i)) += Mt_ItFaixas(ff,i)
    end-do
  end-do
end-do

! Tempo de processamento do padrão
Mt_Padroes(vp,7) := integer(tempo_Proc)

end-procedure

#####
Procedure Atend_Demanda
-----

#####

procedure Atend_Demanda

!## Declaração das variáveis
declarations
  Xm:array(RPd) of mpvar
  Z:linctr
end-declarations

forall(p in RPd)
  Xm(p) is_integer

!## Define a função objetivo
Z := sum(p in RPd)( Xm(p) )

!## Restrições
forall(i in I)
  restr_Dem(i):=sum(p in
  RPd)(Xm(p)*Mt_ItPadroes(p,i))>=It_Demand(i)

forall(p in RPd)
  Restr_NNeg(p) := Xm(p) >=0

!### Processo #####

!## Parametrização
setparam("XPRS_presolve",0) ! desliga o presolver
setparam("XPRS_MAXTIME",TMPMAXPROC_2) ! Parametriza o tempo max.
! de processamento

!## Resolve o Problema relaxado
minimize(XPRS_TOP,Z)

```

```

savebasis(1)

!# Saídas do Resultado da relaxação para Arquivo
fopen(NmArqSol,F_APPEND)
  writeln("\n\n","Sol. da relaxação")
  writeln("\n","Função Objetivo =", strfmt(getobjval,8),"\n")

  forall(p in RPd | getsol(Xm(p))>0)
    writeln("Xm(",strfmt(p,2),")= ",strfmt((getsol(Xm(p))),7,2))
    writeln( "      -----")
    if(sum(p in RPd)(round(getsol(Xm(p))))<>0) then
      vTotPadroes :=sum(p in RPd)((getsol(Xm(p))))
      vPerda_Sobra:=(sum(p in RPd)((getsol(Xm(p)))
        *Mt_Padroes(p,4)/100))
        /(sum(p in RPd)((getsol(Xm(p))))))
      vPerda_Total:=(sum(p in RPd)(Mt_Padroes(p,3)/100
        *getsol(Xm(p)))/(sum(p in RPd)
        (getsol(Xm(p))))))
      vPerda_Serra:= vPerda_Total-vPerda_Sobra
      writeln("TOTAL = ",strfmt(vTotPadroes,7,2)," Padroes")

      writeln("\nPerdas: ")
      writeln("      Sobra= ",strfmt(vPerda_Sobra,4,1))
      writeln("      Serra= ",strfmt(vPerda_Serra,4,1))
      writeln("      ----")
      writeln("      TOTAL= ",strfmt(vPerda_Total,4,1))
    end-if
  fclose(F_APPEND)

!## Resolve o Problema inteiro
loadprob(Z) ! Reload the problem
loadbasis(1)
minimize(XPRS_PRI+XPRS_DUAL,Z)

!## Removendo a compensação do desgaste da serra
L -= Serra
W -= Serra

forall(i in II)do
  ItR_Width(i) -= Serra
  ItR_Lenght(i)-= Serra
end-do

!## Término da geração da solução
Tempo_proc2 := gettime
!### Exporta Padrões para a rotina Defig7 (CorteBi)
Exporta_Padrões

!### Saídas do Resultado para Arquivo #####
fopen(NmArqSol,F_APPEND)
  writeln("\n\n","Sol. inteira")
  writeln("\n","Função Objetivo =", strfmt(getobjval,8),"\n")

  forall(p in RPd | getsol(Xm(p))>0)
    writeln( "Xm(", strfmt(p,2),")= ",strfmt((getsol(Xm(p))),3,0))
    writeln( "      ---")
    if(sum(p in RPd)(round(getsol(Xm(p))))<>0) then
      vTotPadroes :=sum(p in RPd)(round(getsol(Xm(p))))
      vPerda_Sobra:=(sum(p in RPd)(round(getsol(Xm(p)))
        *Mt_Padroes(p,4)/100))
        /(sum(p in RPd)(round(getsol(Xm(p))))))
      vPerda_Total:=(sum(p in RPd)(Mt_Padroes(p,3)/100
        *round(getsol(Xm(p))))
        /(sum(p in RPd)(round(getsol(Xm(p))))))
      vPerda_Serra:= vPerda_Total-vPerda_Sobra
      writeln( "TOTAL = ",strfmt(vTotPadroes,3)," Padroes")

      writeln("\nPerdas: ")

```

```

        writeln("      Sobra= ",strfmt(vPerda_Sobra,4,1))
        writeln("      Serra= ",strfmt(vPerda_Serra,4,1))
        writeln("      ----")
        writeln("      TOTAL= ",strfmt(vPerda_Total,4,1))
    end-if

    writeln("\n\nQtde de Itens Gerados\n")

    forall(i in I)
        writeln("Item ", strfmt(i,2),"(",strfmt(It_Width(i),4),
            "x",strfmt(It_Lenght(i),4),")= ",
            strfmt(sum(p in RPd)(Mt_ItPadroes(p,i)
            *round (getsol(Xm(p))))),4),
            "      (" ,strfmt(It_Demand(i),4),")")

    writeln("\n\nPadroes utilizados:")

    forall(p in RPd)do
        if(getsol(Xm(p))>0)then
            writeln("\n", "      Padrão: ",strfmt(p,2,0)," W x L: ",
                strfmt(Mt_Padroes(p,1),1,0)," x ",
                strfmt(Mt_Padroes(p,2),1,0),
                "      Perda Total / Sobra: ",
                strfmt(Mt_Padroes(p,3)/100,3,1),
                " / ",strfmt(Mt_Padroes(p,4)/100,3,1)," \n")
            forall(i in I)
                writeln("      Item ", strfmt(i,2),"(",
                    strfmt(It_Width(i),4),"x",
                    strfmt(It_Lenght(i),4),")= ",
                    strfmt(Mt_ItPadroes(p,i),2))
            end-if
        end-do

    writeln("\n\nFim do Processo \n")
    writeln("Tempo Total: ",strfmt(Tempo_proc2-Tempo_procl,1),
        " s\n")
    fclose(F_APPEND)

```

end-procedure

```

#####
Procedure Exporta_Padroes
-----

```

Realiza a exportação dos dados dos padrões para que possa ser desenhado no programa Defig7, desenvolvido por Tiago ????? (2004)

```

#####

```

procedure Exporta\_Padroes

```

forall(p in RPd)do
    if(p<10) then
        flag := "00"
    elif(p<100)then
        flag := "0"
    else
        flag := ""
    end-if

    fopen(NmArqTxt-" .txt"+"-"+flag+strfmt(p,1)+".txt",F_OUTPUT)
    writeln(NmArqTxt-" .txt"+"-"+flag+strfmt(p,1))
    if(Mt_Padroes(p,6)=1)then
        writeln(L," ",W)
    else
        writeln(W," ",L)
    end-if
end-do

```

```

writeln(Serra)
writeln(strfmt(Mt_Padros(p,3)/100,6,2))
writeln(2*nr_Itens)

forall(i in II)
  writeln(ItR_Lenght(i)," ",ItR_Width(i))

vNrFaixas:=0
forall(f in RPf)
  if(sum(i in II)(Mt_FxPadros(p,f,i ))>0) then
    vNrFaixas += 1
  end-if

writeln(vNrFaixas)
forall(f in RPf)do
  if(sum(i in II)(Mt_FxPadros(p,vNrFaixas-f+1,i))>0) then
    write(1," ")
    forall(i in II)
      write(strfmt(Mt_FxPadros(p,vNrFaixas-f+1,i),3)," ")
    write("\n")
  end-if
end-do
fclose(F_OUTPUT)
end-do

```

```
end-procedure
```

```
end-model
```

## Referências Bibliográficas

---

ASSOCIAÇÃO BRASILEIRA DAS INDÚSTRIAS DO MOBILIÁRIO. **Panorama da Indústria Moveleira no Brasil: Informações Gerais**. Outubro/05, 55 p. São Paulo, 2005.

ARENALES, M. N.; MORABITO, R.; YANASSE, H. H. Problemas de Corte e Empacotamento. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 36., 2004, São João Del Rei. **Mini curso...**, São João Del Rei: SOBRAPO, 2004. p. 2690 - 2769. CD-ROM

AZEVEDO, A. B. A. **As implicações da difusão de normas técnicas para o aperfeiçoamento tecnológico da indústria moveleira**. 2003. 89 f. Dissertação (Mestrado), Curso de Pós-graduação em Política Científica e Tecnológica, Unicamp, Campinas/SP, 2003.

BARBOSA, T. H. Interface gráfica para a visualização de padrões de corte bidimensionais. São José do Rio Preto, 2005. **Relatório de Estágio Básico do DCCE/IBILCE/UNESP**, São José do Rio Preto, 2005. Orientadora: Prof<sup>a</sup> Dr<sup>a</sup> Socorro Rangel.

BARNHART, C.; et al. Branch-and-price: Column generation for solving huge integer programs, **Operations Research**, 46(3), p. 316-329, 1998

BAZARAA, M. S.; JARVIS, J. J.; HANIF, D. S. **Linear programming and network flows, 2.ed.**, 1990. New York: John Wiley & Sons, 684p.

BELOV, G.; SCHEITHAUER, G. A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting. **European Journal of Operational Research**, 171, 85–106, 2006

CACCETA, L.; HILL, S. P. Branch-and-Bound Methods for Network Optimization. **Mathematical and Computer Modelling**, 33, 517-532, 2001

CAVALI, R. **Problemas de Corte e Empacotamento na Indústria de Móveis: Um Estudo de Caso**, 2004, 87 f. Dissertação (Mestrado), Pós-Graduação em Matemática Aplicada, UNESP, São José do Rio Preto/SP, 2004

CAVALI, R; RANGEL, S. Production Planning: A Cutting Stock Problem In The Furniture Industry. In: CONGRESO LATINO-IBEROAMERICANO DE INVESTIGACIÓN DE OPERACIONES Y SISTEMAS, 12., 2004, **Anais...**, Havana/cuba, 2004. v. único , T137.

CINTRA, G. F. **Algoritmos para Problemas de Corte de Guilhotina Bidimensional**, 2004, 145 f. Tese (Doutorado), Pós-Graduação em Ciencia da Computacao) - USP, São Paulo/SP, 2004

COLOMBANI, Y.; HEIPCKE. S. **Mosel: An Overview**, Blisworth, 2004. Blisworth: Dash Associates, october/2004. Disponível em <<http://www.dashoptimization.com>>. Acesso em: 20/12/2005

DASH OPTIMIZATION. **Modeling with Xpress-MP**, Blisworth, 2004. Blisworth: Dash Associates. Disponível em <<http://www.dashoptimization.com>>. Acesso em: 20/12/2005

DYCKHOFF, H. A typology of cutting and packing problems. **European Journal Operational Research**, v.44, p.145-159, 1990

GILMORE, P. C.; GOMORY, R. E. A linear programming approach to the cutting-stock problem. **Operations Research**, 9, 849-859, 1962

GILMORE, P. C.; GOMORY, R. E. A linear programming approach to the cutting-stock problem – Part II, **Operations Research**, 11, 863-888, 1963

GILMORE, P. C.; GOMORY, R. E. Multistage cutting stock problems of two and more dimensional. **Operations Research**, 14, 1045-1074, 1965

GORINI, A P F. Panorama do setor moveleiro no Brasil com ênfase na competitividade externa a partir do desenvolvimento da cadeia externa de produtos sólidos de madeira. Rio de Janeiro, 1998. Rio de Janeiro: **BNDES Setorial 8**, setembro/1998. Disponível em: <<http://www.bndes.gov.br/conhecimento/Bnset/set801.pdf>>. Acesso em: 31/03/1963.

HEIPCKE, S. **Embedding Optimization Algorithms**, Blisworth, 2004. Blisworth: Dash Associates, october/2004. disponível em <<http://www.dashoptimization.com>>. Acesso em: 20/12/2005

JUSTINA, M. D. **Restrições e estratégias na determinação do preço no mercado moveleiro de São Bento do Sul**, 2004, 144 f. Programa de Pós-Graduação em Informática, PUC, Rio de Janeiro, 2004

KATSURAYAMA, D. **Estudo sobre o Efeito da Utilização de Padrões Tabuleiros na Produtividade do Equipamento de Cortes**, 2002, 121 f. Dissertação (Mestrado), Pós-Graduação em Computação Aplicada, INPE, São José dos Campos, 2002

MORABITO, R. ; ARENALES, M. N. Optimizing the cutting of stock plates in a furniture company. **International Journal Of Production Research**, Inglaterra, v. 38, n. 12, p. 2725-2742, 2000.

NEMHAUSER, G. L. e WOLSEY, L. A. **Integer and Combinatorial Optimization**. ISBN 0-471-8219-X, 1988

PERIN, C.; RANGEL, S. O problema do corte bidimensional. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 12., 1989, São José do Rio Preto. **Anais...**, São Carlos: SBMAC, 1989. v. 1.

FIGUEIREDO, A.; RANGEL, S. Aplicação de modelos 2-estágios e 1-grupo na geração de padrões de corte na indústria moveleira. In: CONGRESSO NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 27., 2005, São Paulo. **Anais...**, São Carlos: SBMAC, 2005. v. único. CD-ROM

RANGEL, S. **O problema do corte bidimensional**, 1989, 97 f. Dissertação (Mestrado), Pós-Graduação em Matemática Aplicada, UNICAMP, Campinas, 1989

SILVA, E. M. **Alinhamento das Estratégias competitivas com as estratégias de produção: Estudo de casos no Polo Moveleiro de Votuporanga/SP**, 2003, 163 f. Dissertação (Mestrado), Pós-Graduação em Engenharia de Produção, USP, São Carlos, 2003

STIPP, M.S. **Cluster Industrial: O Pólo moveleiro de Votuporanga-SP (1962-2001)**, 2002. Dissertação (Mestrado), Pós-Graduação em Economia, UNESP, Araraquara, 2002

VANCE, P. H. Branch-and-Price Algorithms for the One-Dimensional Cutting Stock Problem. **Computational Optimization and Applications**, 9, 211-228, 1998

WOLSEY, L. A. **Integer Programming. Wiley-Interscience publication.** ISBN 0-471-28366-5, 1998

WÄSCHER, G.; HAUBNER, H.; SCHUMANN, H., **An Improved Typology of Cutting and Packing Problems**, Working Paper No. 24, December 2004, Faculty of Economics and Management Magdeburg, 2004

YANASSE, H. H.; MORABITO, R. Linear models for one-group two-dimensional guillotine cutting problems. **International Journal of Production Research**, 2006.

YANASSE, H. H. ; KATSURAYAMA, D. M. An enumeration scheme to generate constrained exact checkerboard patterns. In: EURO CONFERENCE, 20., 2004, Rhodes. **Book of Abstracts...**, 2004. v. 1. p. 36-36.

YANASSE, H. H. ; KATSURAYAMA, D. M. Uma análise de produtividade do equipamento de cortes utilizando-se padrões tabuleiro. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 21. / INTERNATIONAL CONFERENCE ON INDUSTRIAL ENGINEERING AND OPERATIONS MANAGEMENT, 7., 2001, Salvador. 2001. **Anais...**, Porto Alegre: ABEPRO, 2001. v. 1. CD-ROM

YANASSE, H.H.; HARRIS, R.G.; ZINOBER, A.S.I. Uma Heurística para redução do número de ciclos da serra no corte de chapas. In: XIII ENEGEP - Encontro Nacional de Engenharia de Produção/ I Congresso Latino Americano de Engenharia Industrial, Florianópolis, SC, 05 a 08 de outubro de 1993. Publicado nos **Anais** do XIII ENEGEP, Florianópolis, Universidade Federal de Santa Catarina, 1993. Vol.II, p.879-885.

# Livros Grátis

( <http://www.livrosgratis.com.br> )

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)  
[Baixar livros de Literatura de Cordel](#)  
[Baixar livros de Literatura Infantil](#)  
[Baixar livros de Matemática](#)  
[Baixar livros de Medicina](#)  
[Baixar livros de Medicina Veterinária](#)  
[Baixar livros de Meio Ambiente](#)  
[Baixar livros de Meteorologia](#)  
[Baixar Monografias e TCC](#)  
[Baixar livros Multidisciplinar](#)  
[Baixar livros de Música](#)  
[Baixar livros de Psicologia](#)  
[Baixar livros de Química](#)  
[Baixar livros de Saúde Coletiva](#)  
[Baixar livros de Serviço Social](#)  
[Baixar livros de Sociologia](#)  
[Baixar livros de Teologia](#)  
[Baixar livros de Trabalho](#)  
[Baixar livros de Turismo](#)