

# Um sistema computacional para auxiliar no planejamento de cortes em bobinas de aço

## A computer system to aid in the planning of steel rolls cuts

Robinson Samuel Hoto<sup>1</sup>; Fernando Spolador<sup>2</sup>; Nelson Maculan<sup>3</sup>

---

### Resumo

O planejamento de cortes em bobinas de aço é um problema de otimização combinatória. Algumas empresas do setor de metalurgia efetuam a laminação a frio do aço destas bobinas para que ele adquira propriedades físicas desejáveis. Nesse caso, os padrões de corte devem ser compostos por compartimentos de itens compatíveis para o processo de laminação, dificultando a tarefa do planejamento dos cortes. Um compartimento representa uma bobina intermediária a ser laminada, de modo que é possível combinar bobinas intermediárias com diferentes necessidades de laminação em uma mesma bobina do estoque. Neste trabalho, é apresentado o sistema RollCut, que auxilia o planejamento dos cortes de bobinas.

**Palavras-chave:** Problema de corte. Duas fases. Padrões compartimentados.

---

### Abstract

The planning of cuts in steel rolls is a combinatory optimization problem. Some companies of the metallurgical industry use the steel cold lamination process so that it acquires the necessary physical properties. In this case, the cutting patterns should consist of compartments of items compatible with the lamination process, hindering the task of cuts planning. A compartment represents an intermediate roll to be laminated, so that it is possible to combine intermediate rolls with different lamination needs in the same roll of the stock. In this work the prototype of the RollCut System will be presented to aid with the cuts planning.

**Key words:** Cutting problem. Two phases. Compartmentalized patterns.

---

<sup>1</sup> Docente do Departamento Matemática da Universidade Estadual de Londrina, Londrina/PR. E-mail: hoto@uel.br

<sup>2</sup> Aluno do Programa de Mestrado do Departamento de Engenharia de Sistemas da Universidade Estadual de Campinas. E-mail: spolferrando@yahoo.com.br.

<sup>3</sup> Docente do Programa de Engenharia de Sistemas e Computação da Universidade Federal do Rio de Janeiro. E-mail: maculan@cos.ufrj.br.

## Introdução

Padrões de corte compartimentados (HOTO, 2001) são comuns em empresas do ramo da metalurgia que necessitam efetuar laminação a frio do aço acondicionado em bobinas. Estas bobinas do estoque são identificadas pelo seu “peso” (entre de 12.000 e 13.500 kg), sua largura (entre 1000 mm e 1300 mm), espessura do aço (entre de 0,90 mm e 6,30 mm) e pelo teor de carbono na liga, que determinará a dureza do aço. A demanda é composta por fitas de especificações determinadas (largura, espessura e tipo de aço) que devem ser cortadas de uma bobina do estoque e, devido a razões técnicas, a maioria dessas fitas precisam ter a espessura do aço reduzida por meio de laminação. A laminação é feita com “cilindros de laminação” que efetuam pressão sobre a lâmina de aço em temperatura ambiente (laminação a frio). Devido a limitações técnicas, a máquina de laminação não pode processar a largura total de uma bobina do estoque e além do mais, existem vários conjuntos de fitas com reduções de espessura de aço distintas, de modo que devem ser definidas bobinas intermediárias com larguras admissíveis pelo laminador e compostas de fitas compatíveis entre si. Essas bobinas intermediárias são cortadas de uma bobina em estoque e, delas são recortadas as fitas, de modo que, os padrões de corte devem ser estruturados em compartimentos.

Em trabalhos anteriores, desenvolvemos um sistema computacional denominado RollCut, capaz de planejar os cortes de bobinas de aço sujeitas a laminação a frio. Neste trabalho, acrescentamos ao sistema RollCut um módulo no qual os padrões de corte são visualizados graficamente. Do mesmo modo, efetuamos testes da utilização de memória, após as otimizações em estruturas originais do código.

A abordagem adotada na resolução do Problema de Corte de Bobinas de Aço (HOTO et al., 2002) é a Técnica de Geração de Colunas de Gilmore- Gomory (GILMORE; GOMORY, 1961; GILMORE; GOMORY, 1963). A modelagem matemática do problema, cujos detalhes podem ser encontrados em

(HOTO, 2001), tem por objetivo a minimização de custos associados às perdas de aço numa bobina, e custos por utilização de compartimentos (com ou sem laminação), visando ao cumprimento de uma demanda, e respeitando a disponibilidade de estoque. Para isso consideramos:

- $c_{aço}^r$  custo linear do aço da bobina (objeto)  $r = 1, \dots, m$  (**\$/unidade de comprimento**);
- $T_j^r$  perda linear de aço na bobina (objeto)  $r = 1, \dots, m$ , segundo o padrão compartimentado  $j$  (**unidades de comprimentos**);
- $c_{hj}^r$  custo operacional da bobina intermediária (compartimento)  $h$  na bobina  $r$ , segundo o padrão compartimentado  $j$  (**\$/unidade do compartimento  $h$** );
- $H_{jr}$  número total de bobinas intermediárias (compartimentos) na bobina  $r$ , segundo o padrão compartimentado  $j$  (**unidades de compartimentos**);
- $y_{hj}^r$  número de bobinas intermediárias (compartimentos)  $h$  na bobina  $r$ , segundo o padrão compartimentado  $j$  (**unidades de compartimentos**);
- $x_j^r$  número de bobinas utilizadas do tipo  $r$ , segundo o padrão compartimentado  $j$  (**unidades de bobinas**).

Limitações no estoque também são admitidas, de modo que uma disponibilidade  $e_r$  para cada tipo de bobina  $r = 1, \dots, m$  é considerada e ainda é introduzido:

- $p^r$  “peso” (a massa) da bobina (objeto)  $r = 1, \dots, m$  (**Kg**);
- $L^r$  largura da bobina (objeto)  $r = 1, \dots, m$  (**mm**);
- $\ell_i$  largura da fita (item)  $i = 1, \dots, n$  (**mm**);
- $d_i$  demanda solicitada da fita (item)  $i = 1, \dots, n$  (**Kg**);
- $a_{ij}^r$  número de fitas (itens) tipo  $i$  na bobina  $r$ , segundo o padrão compartimentado  $j$  (**unidades de fitas**).

O modelo matemático para o qual o RollCut foi concebido é apresentado a seguir:

$$\text{minimizar } \sum_{r=1}^m \sum_{j=1}^{p_r} \left( c_{aço}^r T_j^r + \sum_{h=1}^{H_{jr}} (c_{hj}^r y_{hj}^r) \right) x_j^r$$

sujeito a:

$$\sum_{r=1}^m \sum_{j=1}^{p_r} \frac{P^r}{L^r} \ell_i a_{ij}^r x_j^r = d_i, \quad i=1, \dots, n$$

$$\begin{array}{ccc} \sum_{j=1}^{p_1} x_j^1 & & \leq e_1 \\ \vdots & \ddots & \vdots \\ \sum_{j=1}^{p_m} x_j^m & & \leq e_m \end{array}$$

$$x_j^r \geq 0, \text{ e inteiro } r=1, \dots, m, \quad j=1, \dots, p_r$$

Como mencionamos, a primeira parcela da função objetivo esta relacionada com a perda de aço durante o processo, já a segunda parcela representa os custos associados aos processos operacionais. O primeiro grupo de restrições (igualdade) refere-se ao atendimento das demandas, e o segundo grupo de restrições (desigualdades) modela a disponibilidade de bobinas em estoque.

O gerador de colunas do modelo anterior consiste numa Mochila Compartimentada (HOTO; ARENALES; MACULAN, 1999; HOTO et al., 2002), cujo modelo é descrito como:

maximizar

$$\begin{aligned} & \sum_{h \in V_1} \left( \left( \sum_{i \in N_1} (c_{aço}^r + \pi_i \frac{P^r}{L^r}) \ell_i a_{ih}^r \right) - (c_{hj}^r - c_{aço}^r S_2) \right) y_h + \dots + \\ & + \sum_{h \in V_k} \left( \left( \sum_{i \in N_k} (c_{aço}^r + \pi_i \frac{P^r}{L^r}) \ell_i a_{ih}^r \right) - (c_{hj}^r - c_{aço}^r S_2) \right) y_h \end{aligned}$$

sujeito a:

$$\sum_{h \in V_1} (S_2 + \sum_{i \in N_1} \ell_i a_{ih}^r) y_h + \dots + \sum_{h \in V_k} (S_2 + \sum_{i \in N_k} \ell_i a_{ih}^r) y_h \leq L^r - S_1$$

$$\left\lfloor \frac{y_k}{y_k + 1} \right\rfloor L_{\min} \leq S_2 + \sum_{i \in N_s} \ell_i a_{ih}^r \leq \left\lceil \frac{y_k}{y_k + 1} \right\rceil L_{\max}, \quad h \in V_s, \quad s=1, \dots, k$$

$$a_{ih}^r, y_h \geq 0 \text{ e inteiros, } i \in N_1 \cup \dots \cup N_k, \quad h \in V_1 \cup \dots \cup V_k$$

No modelo do gerador de colunas  $L_{\min}$  e  $L_{\max}$  são respectivamente as larguras mínima e máxima que uma bobina intermediária pode apresentar, em razão do processo de laminação. Os parâmetros  $S_1$  e  $S_2$  são respectivamente os tamanhos dos refilos nas bordas das bobinas do estoque e das bobinas intermediárias. A variável  $y_h$  determina o número de vezes que o compartimento  $h$  aparece na mochila e  $\pi_i$  é o multiplicador Simplex associado ao item  $i$ , por fim, cada  $V_s$  representa o conjunto de índices dos compartimentos viáveis construídos com itens indexados por  $N_s$ .

Na seção 1, descreveremos as funções básicas do RollCut; na seção 2, apresentaremos diagramas de fluxos de dados para um melhor entendimento do sistema e, discutiremos as alterações efetuadas nas estruturas de dados; na seção 3 apresentaremos os resultados numéricos que obtivemos com exemplos simulados e as conclusões. Já na seção 4, discutiremos uma nova abordagem, na qual estamos trabalhando para gerar padrões compartimentados restritos.

## A Interface do Sistema RollCut

O RollCut foi implementado em Delphi para plataforma Windows, ocupa 5Mb em disco e é composto de um formulário principal que está ilustrado na figura 1, na qual descrevemos seus comandos de operação. No lado direito do formulário há um campo de *Status* que fornece informações em tempo de execução.

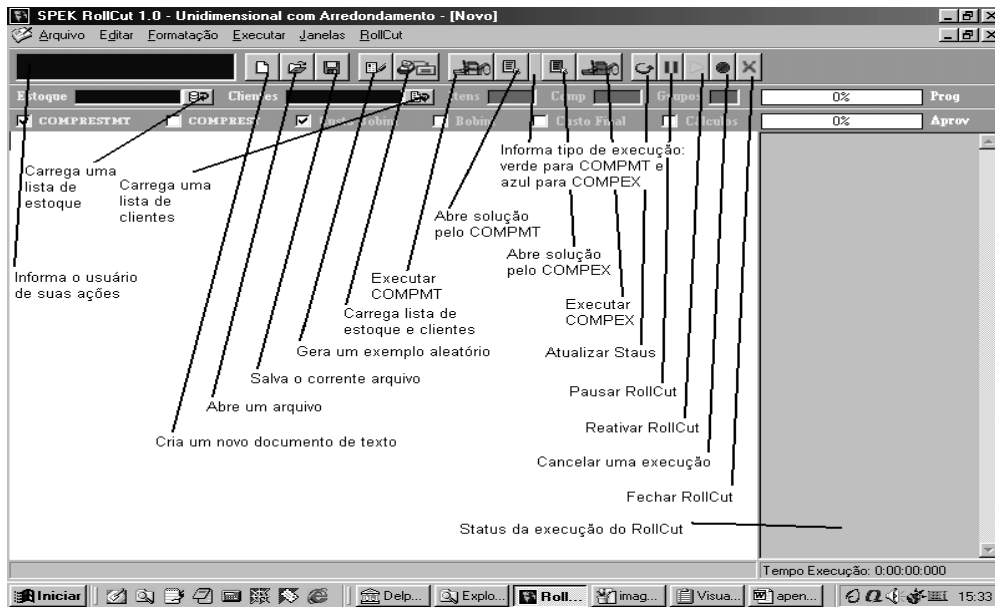


Figura 1. Tela de entrada do RollCut.

Nos campos **Estoque** e **Clientes**, o RollCut informa o nome dos arquivos de estoque e de clientes. Caso a lista de estoque seja insuficiente para atender a lista de clientes, o sistema solicita que uma nova lista seja carregada, não permitindo a execução enquanto esta tarefa não for completada.

O aplicativo possui um editor de texto integrado no qual podem ser visualizados os arquivos de estoque e de clientes (Figura 2). O editor de texto integrado ao RollCut também permite visualizar os relatórios com os resultados das soluções, ou qualquer arquivo que o usuário precise ler e editar.

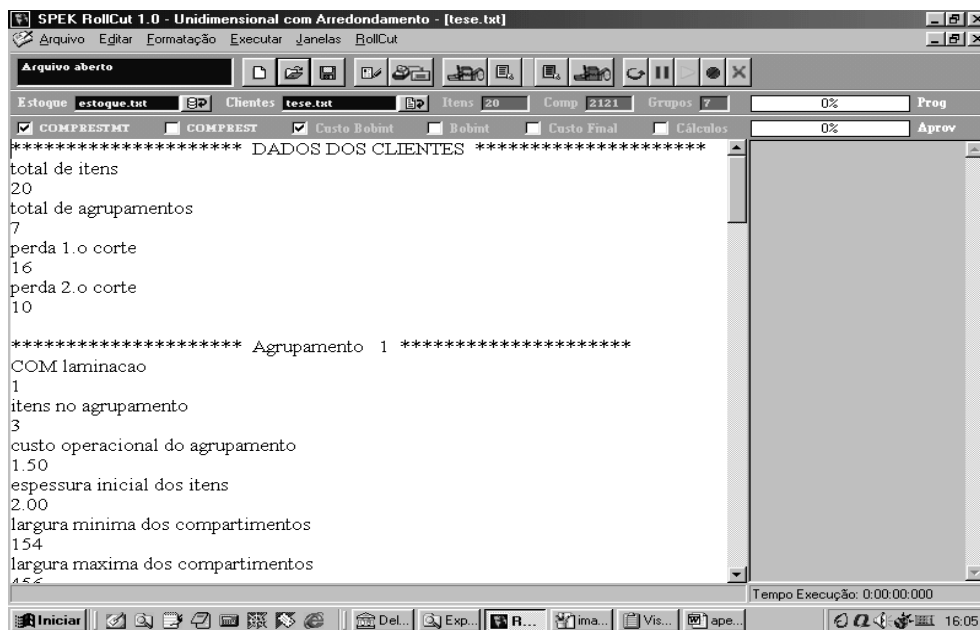


Figura 2. Editor de texto do RollCut.

Após o carregamento da lista de clientes, o sistema informa no campo **Itens** o número total de itens a serem processados, no campo **Comp** o número total de compartimentos a serem examinados e no campo **Grupos** o número de agrupamentos de itens sujeitos ao mesmo tipo de laminação.

Há duas formas básicas de gerar padrões compartimentados (resolver Mochilas Compartimentadas): uma faz uso do algoritmo COMPEX, e a outra, faz uso do algoritmo COMPMT, que é baseado no cálculo de limitantes superiores (resultados sobre o desempenho destes geradores estão detalhados em Hoto (2002)).

Para gerar padrões compartimentados com a restrição adicional  $\sum_{h \in V_1} a_{ih}^r y_h + \dots + \sum_{h \in V_k} a_{ih}^r y_h \leq d_i$ ,  $i \in N_1 \cup \dots \cup N_k$ , o usuário deve optar entre duas heurísticas: **COMPRESTMT** ou **COMPREST** [5, 7].

Após a execução do RollCut, o usuário pode visualizar o relatório do planejamento dos cortes pelo simples acionamento de um botão (Figura 1). Existem ainda algumas opções disponíveis no protótipo acadêmico, e elas servem para obter informações adicionais do desempenho das soluções: **Custo Bobint** (custos das bobinas intermediárias), **Bobint** (número de bobinas intermediárias), **Custo Final** (custos de matéria-prima e laminação) e **Cálculos** (registro dos cálculos das matrizes básicas e inversas, dos multiplicadores Simplex e das colunas geradas). Ainda existem duas barras de progresso **Prog** (progresso de uma execução) e **Aprov** (progresso do aproveitamento alcançado com os padrões de corte). Na próxima seção, faremos um resumo dos fluxos das informações no RollCut e também discutiremos algumas alterações que foram efetuadas no sistema.

## Fluxos de Dados e Alterações no Sistema Original

Na figura 3, ilustramos o funcionamento dos fluxos de dados do sistema RollCut. Os procedimentos 01

e 02 consistem no Simplex com Geração de Colunas.

Os procedimentos 03 e 04 efetuam o arredondamento da solução e permitem tomar decisão acerca da existência ou não de um problema residual. Esse problema será definido pelo procedimento 05 e resolvido através dos procedimentos 01 e 02, cuja solução será agregada à solução arredondada (STADTLER, 1990; WÄSCHER; GAU, 1996). Caso não exista o problema residual os procedimentos 06 e 07 são acionados para garantir o cumprimento da demanda.

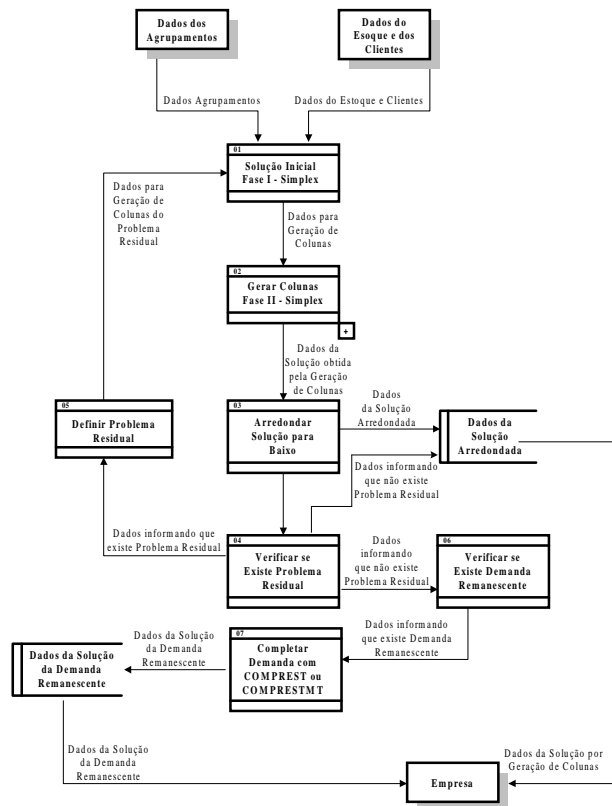


Figura 3. DFD nível 1 (Procedimento Executar RollCut).

Na figura 4, apresentamos a “explosão” do procedimento 02 do DFD nível 1 (Procedimento Gerar Colunas).

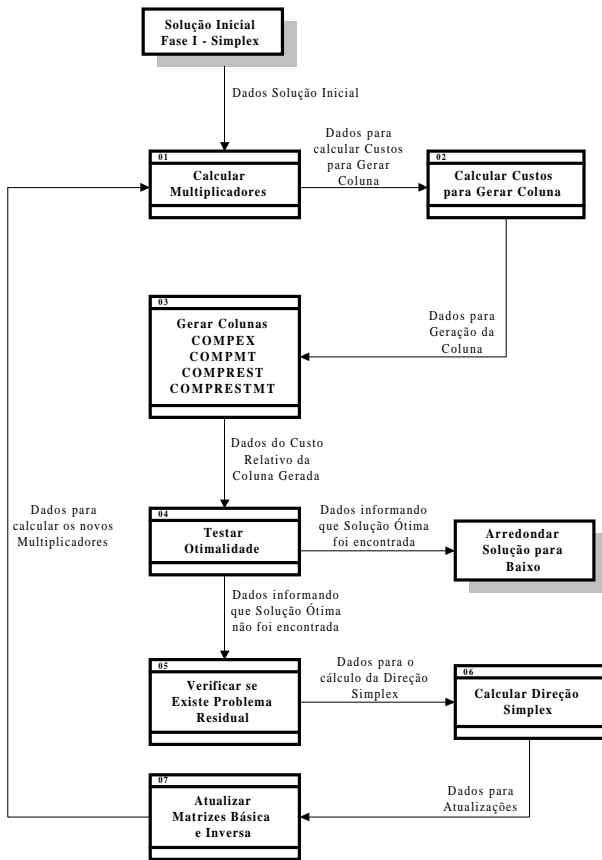


Figura 4. DFD nível 2 (Procedimento Gerar Colunas).

No sistema original, os pormenores de cada padrão de corte eram disponibilizados num relatório. Já na nova versão do RollCut, os padrões podem também ser visualizados graficamente (Figura 5), facilitando a compreensão do usuário.

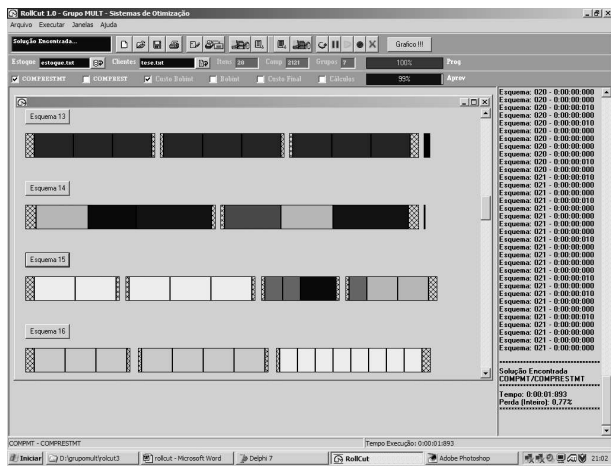


Figura 5. RollCut com painel gráfico de padrões.

Para cada padrão existe um botão que quando acionado pelo usuário abre um formulário com informações resumidas (Figura 6). As informações detalhadas podem ser lidas no relatório.

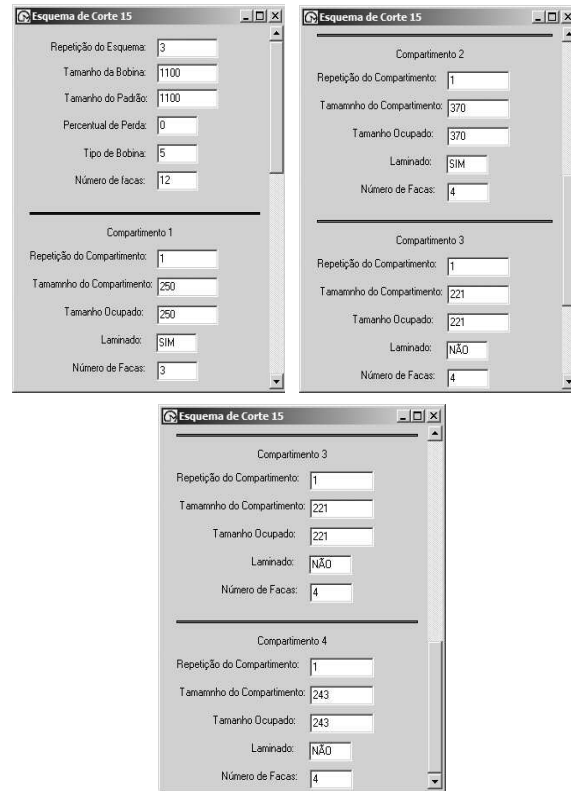


Figura 6. Formulário com os dados dos padrões.

O RollCut foi originalmente implementado com listas estáticas (vetores e matrizes superdimensionadas), alocando 98872 kilobytes numa máquina com processador AMD Duron 850 MHz e 128 Mb de RAM ao ser inicializado. Com o intuito de melhorar seu desempenho, efetuamos alterações significativas em suas estruturas de dados, substituindo as listas estáticas por dinâmicas. Assim o sistema passou a alocar 26636 kilobytes (cerca de 27% da memória alocada com estruturas estáticas).

## Resultados Numéricos e Conclusões

Na tabela 1, apresentamos os resultados que obtivemos com 21 exemplos, utilizando o algoritmo COMPMT para gerar os padrões compartimentados. A tabela 2 contém os resultados dos mesmos exemplos, porém, os padrões foram gerados com o COMPEX.

**Tabela 1.** Resultados Numéricos do RollCut com COMPMT.

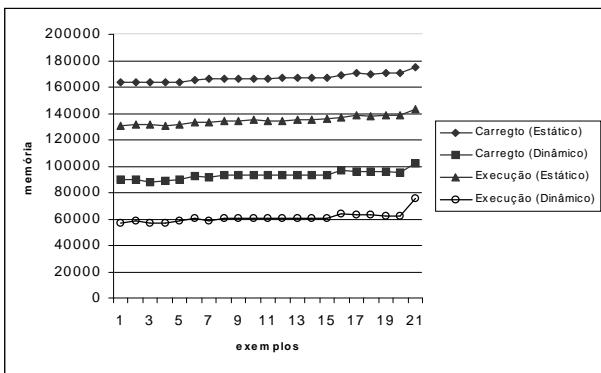
Exemplo			Estático			Dinâmico		
Agrup	Comp	Itens	Carregto	Execução	Tempo	Carregto	Execução	Tempo
5	1515	32	163554	131074	8680	89542	57040	4740
		32	163592	131112	16810	89968	58620	7750
		27	163598	131113	14390	88440	57076	6100
		32	163592	130912	15050	88476	57140	6260
		35	163592	131112	19320	89984	58620	9120
10	3030	62	165628	133340	38280	92140	60392	24380
		62	166144	133340	110340	91660	58804	63000
		52	166612	134324	34170	92904	60088	20540
		60	166612	134324	53330	93104	60312	29830
		57	166472	134756	17800	93004	60205	11090
15	4545	67	166460	134388	41520	93324	60068	28340
		85	167416	134388	85840	93620	60667	57780
		82	167268	135188	65420	93492	60484	42130
		82	167276	135200	63710	93500	60480	43010
		75	166996	136308	54430	93280	60340	34000
20	6060	95	168556	136664	95570	96728	63644	69750
		110	170904	139012	167690	95916	62840	119570
		120	170012	138120	142920	96416	63348	105230
		102	170468	138576	80900	95572	62496	60090
		100	170348	138464	193010	95479	62400	136930
30	9090	142	174696	143224	416280	102472	75859	333340

**Tabela 2.** Resultados Numéricos do RollCut com COMPEX.

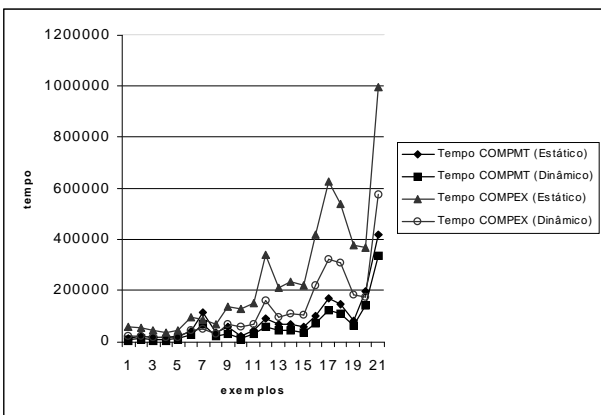
Exemplo			Estático			Dinâmico		
Agrup	Comp	Itens	Carregto	Execução	Tempo	Carregto	Execução	Tempo
5	1515	32	163554	131074	53770	89542	57040	20600
		32	163592	131112	51630	89968	58620	19450
		27	163598	131113	40100	88440	57076	13460
		32	163592	130912	34380	88476	57140	12190
		35	163592	131112	41520	89984	58620	15710
10	3030	62	165628	133340	91340	92140	60392	42350
		62	166144	133340	83740	91660	58804	45970
		52	166612	134324	65250	92904	60088	28070
		60	166612	134324	136660	93104	60312	66190
		57	166472	134756	126220	93004	60205	56350
15	4545	67	166460	134388	145990	93324	60068	67060
		85	167416	134388	337680	93620	60667	156600
		82	167268	135188	210320	93492	60484	92760
		82	167276	135200	231130	93500	60480	108420
		75	166996	136308	216950	93280	60340	102000
20	6060	95	168556	136664	416340	96728	63644	216570
		110	170904	139012	626480	95916	62840	321640
		120	170012	138120	539030	96416	63348	303740
		102	170468	138576	375640	95572	62496	181040
		100	170348	138464	366046	95479	62400	170490
30	9090	142	174696	143224	995850	102472	75859	573800

Nas tabelas 1 e 2 cada coluna informa respectivamente:

- **Agrup:** total de agrupamentos de itens com as mesmas necessidades de laminação.
- **Comp:** total de compartimentos envolvidos para a determinação dos padrões compartimentados.
- **Itens:** total de itens do exemplo.
- **Carregto:** valor em *kilobytes* da memória utilizada após o carregamento das listas de estoque e clientes.
- **Execução:** valor máximo em *kilobytes* da memória utilizada na execução do exemplo.
- **Tempo:** tempo gasto em *milisegundos* para a execução do exemplo.



**Figura 7.** Memória usada com estrutura Estática e Dinâmica no RollCut.



**Figura 8.** Tempo usado com estrutura Estática e Dinâmica no RollCut.

Na figura 7, podemos visualizar a utilização de memória do RollCut para o carregamento de dados e a execução dos exemplos das tabelas 1 e 2. Fica

bastante evidente que as estruturas dinâmicas proporcionaram uma utilização mais racional da memória alocada pelo RollCut. Na figura 8, podemos visualizar o tempo de execução do RollCut com os procedimentos COMPMT e COMPEX na estrutura estática original do sistema, e com as novas estruturas dinâmicas e códigos otimizados, nos quais também é nítido o melhor desempenho. Usando o COMPMT, a maior diferença em tempo de execução foi observada para o exemplo 4 (a estrutura estática gastou 140,4% a mais do tempo gasto pela estrutura dinâmica), a menor diferença foi observada para o exemplo 21 (24,9% a mais). Usando o COMPEX, a maior diferença em tempo de execução foi observada para o exemplo 3 (a estrutura estática gastou 197,9% a mais do tempo gasto pela estrutura dinâmica), a menor diferença foi observada para o exemplo 21 (73,6% a mais). Com esses resultados, entendemos que houve um ganho efetivo com as estruturas dinâmicas que foram inseridas no RollCut, o que de fato era esperado.

Na próxima seção, descreveremos uma nova abordagem que estamos propondo para resolver o caso restrito de uma mochila compartimentada, responsável pela construção dos padrões compartimentados gerados pelo RollCut.

### Padrões Compartimentados Restritos gerados com Simplex

Na prática, é preciso considerar duas restrições adicionais, além da que foi citada na seção 1, durante a geração de um padrão compartimentado, são elas:

- A quantidade de compartimentos na mochila deve ser limitada por  $F_1 - 1$ , onde  $F_1$  é o número de facas que podem ser ajustadas durante a primeira etapa de corte (duas delas são usadas para o corte de refilos laterais da bobina):

$$\sum_{h \in V_1} y_h + \dots + \sum_{h \in V_k} y_h \leq F_1 - 1$$



• A quantidade de itens em cada compartimento  $h$  deve ser limitada por  $F_2 - 1$ , onde  $F_2$  é o número de facas que podem ser ajustadas durante a segunda etapa de corte (duas delas são usadas para o corte de refilos laterais da bobina intermediária):

$$\sum_{i \in N_s} a_{ih}^r \leq F_2 - 1, \quad h \in V_1 \cup \dots \cup V_k, \quad s = 1, \dots, k;$$

Assim, o modelo para gerar um padrão compartimentado tem de fato o seguinte aspecto:

$$z = \sum_{h \in V_1} \left( \sum_{i \in N_1} (c_{aço}^r + \pi_i \frac{P^r}{L^r}) \ell_i a_{ih}^r - (c_{hj}^r - c_{aço}^r S_2) \right) y_h + \dots$$

$$\dots + \sum_{h \in V_k} \left( \sum_{i \in N_k} (c_{aço}^r + \pi_i \frac{P^r}{L^r}) \ell_i a_{ih}^r - (c_{hj}^r - c_{aço}^r S_2) \right) y_h$$

sujeito a:

$$\sum_{h \in V_1} (S_2 + \sum_{i \in N_1} \ell_i a_{ih}^r) y_h + \dots + \sum_{h \in V_k} (S_2 + \sum_{i \in N_k} \ell_i a_{ih}^r) y_h \leq L^r - S_1$$

$$\sum_{h \in V_1} y_h + \dots + \sum_{h \in V_k} y_h \leq F_1 - 1$$

$$\left[ \frac{y_k}{y_k + 1} \right] L_{\min} \leq S_2 + \sum_{i \in N_s} \ell_i a_{ih}^r \leq \left[ \frac{y_k}{y_k + 1} \right] L_{\max}, \quad h \in V_s, \quad s = 1, \dots, k$$

$$\sum_{i \in N_s} a_{ih}^r \leq F_2 - 1, \quad h \in V_1 \cup \dots \cup V_k, \quad s = 1, \dots, k;$$

$$\sum_{h \in V_1} a_{ih}^r y_h + \dots + \sum_{h \in V_k} a_{ih}^r y_h \leq d_i, \quad i \in N_1 \cup \dots \cup N_k$$

$$a_{ih}^r, y_h \geq 0 \text{ e inteiros}, \quad i \in N_1 \cup \dots \cup N_k, \quad h \in V_1 \cup \dots \cup V_k$$

Digamos que o número total de compartimentos que envolvem a resolução de uma mochila compartimentada seja computável para o Simplex, isto é, o total de variáveis  $y_h$  permite o emprego do Simplex em tempo exequível. A abordagem que sugerimos para obter uma solução viável da compartimentação restrita consiste em:

### Algoritmo para Compartimentação Restrita com Simplex - COMPREST com Simplex

1. Para cada compartimento viável de capacidade  $L_h \in \{L_{\min}, L_{\min} + 1, \dots, L_{\max}\}$ ,  $h \in V_s$  com  $s = 1, \dots, k$ , resolva a seguinte mochila que fornecerá o maior valor de utilidade do compartimento  $h$ , com a restrição adicional  $\sum_{i \in N_s} a_{ih}^r \leq F_2 - 1$ :

$$\text{maximizar } v_h = \sum_{i \in N_s} (c_{aço}^r + \pi_i \frac{P^r}{L^r}) \ell_i a_{ih}^r$$

sujeito a:

$$\sum_{i \in N_s} \ell_i a_{ih}^r = L_h - S_2$$

$$\sum_{i \in N_s} a_{ih}^r \leq F_2 - 1$$

$$a_{ih}^r \geq 0 \text{ e inteiros}, \quad i \in N_1 \cup \dots \cup N_k$$

2. Utilize o Simplex para obter uma solução contínua do seguinte Programa Linear Inteiro, obtendo um padrão compartimentado viável:

maximizar

$$\sum_{h \in V_1} (v_h - (c_{hj}^r - c_{aço}^r S_2)) y_h + \dots + \sum_{h \in V_k} (v_h - (c_{hj}^r - c_{aço}^r S_2)) y_h$$

sujeito a:

$$\sum_{h \in V_1} a_{ih}^r y_h + \dots + \sum_{h \in V_k} a_{ih}^r y_h \leq d_i,$$

$$\sum_{h \in V_1} L_h y_k + \dots + \sum_{h \in V_k} L_h y_h \leq L^r - S_1$$

$$\sum_{h \in V_1} y_h + \dots + \sum_{h \in V_k} y_h \leq F_1 - 1$$

$$y_h \geq 0 \text{ e inteiros}, \quad h \in V_1 \cup \dots \cup V_k$$

**Observação 1** No passo 1 do algoritmo, para cada bobina  $r$  do estoque serão calculadas

$k(L_{\max} - L_{\min} + 1)$  mochilas, de modo que, no pior caso,  $rk(L_{\max} - L_{\min} + 1)$  mochilas serão calculadas. O cálculo destas mochilas é necessário para obtermos as utilidades  $v_h$  dos compartimentos, porém, um limitante superior destas utilidades poderia ser calculado rapidamente sem o esforço de se resolver mochilas.

**Observação 2** Para obtermos uma solução inteira viável do Programa Linear Inteiro do passo 2, necessitaremos um procedimento de arredondamento, por exemplo (STADTLER, 1990; WÄSCHER; GAU, 1996).

**Observação 3** Cada coluna  $a_h^r$  da matriz do Programa Linear Inteiro do passo 2 tem a seguinte configuração  $a_h^r = (a_{1h}^r \cdots a_{nh}^r L_h 1)^T$ , onde cada uma delas refere-se a um compartimento, e elas informam o quanto de cada um dos  $n$  tipos de itens ocorrem no compartimento ao qual ela faz referência. Além disso, elas trazem a capacidade do compartimento (tamanho da bobina intermediária) e a contribuição que será dada no preenchimento da mochila compartimentada.

Atualmente estamos trabalhando na implementação do algoritmo descrito para avaliarmos seu desempenho em relação às heurísticas já desenvolvidas (HOTO, 2001; HOTO et al., 2002; MARQUES, 2000; MARQUES; ARENALES; HOTO, 2002; MARQUES; ARENALES, 2002; ZAK, 2002a; ZAK, 2002b).

## Agradecimentos

Este trabalho contou com apoio financeiro do Mestrado em Engenharia Elétrica da Universidade Estadual de Londrina (UEL), da Fundação Araucária e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

## Referências

- GILMORE, P. C.; GOMORY, R. E. A linear programming approach to the cutting stock problem. *Operations Research*, Baltimore, v.9, n.6, p.849-859, 1961.
- \_\_\_\_\_. A Linear Programming Approach to the cutting stock problem, part II. *Operations Research*, Baltimore, v.11, n.6, p.863-888, 1963.
- HOTO, R. *O problema da mochila compartimentada aplicado no corte de bobinas de aço*. 2001. Tese (Doutorado em Engenharia de Sistema e Computação) – Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- HOTO, R.; ARENALES, M. N.; MACULAN, N. *O Problema da mochila compartimentada*. Londrina: UEL, 1999. Relatório Técnico, Departamento de Matemática.
- HOTO, R.; MACULAN, N.; ARENALES, M. N.; MARQUES, F. P. Um novo procedimento para o cálculo de mochilas compartimentadas. *Pesquisa Operacional*, Rio de Janeiro, v.22, p.213-234, 2002.
- MARQUES, F. P. *O problema da mochila compartimentada*. 2000. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação/USP, São Carlos.
- MARQUES, F. P.; ARENALES, M. N. O problema da mochila compartimentada e aplicações. *Pesquisa Operacional*, Rio de Janeiro, v.22, n.3, p.285-304, 2002.
- MARQUES, F. P.; ARENALES, M. N.; HOTO, R. O problema da mochila compartimentada: uma revisão dos casos irrestrito e restrito. In: OFICINA NACIONAL DE PROBLEMAS DE CORTE E EMPACOTAMENTO, 6., 2002, Campinas. *Anais...* Campinas, 2002. p.ini-final
- STADTLER, H. A one dimensional cutting-stock problem in the aluminium industry and its solution. *European Journal of Operational Research*, Amsterdam, v.44, n.2, p.209-223, jan. 1990.
- WÄSCHER, G.; GAU, T. Heuristics for the integer one dimensional cutting-stock problem: a computational study. *OR Spektrum*, New York, v.18, n.3, p.131-144, jul. 1996
- ZAK, E. J. Modeling multistage cutting stock problems. *European Journal of Operational Research*, Amsterdam, v.141, n.2, p.313-327, sep. 2002a.
- \_\_\_\_\_. Row and column generation technique for a multistage cutting stock problem. *Computers & Operations Research*, New York, v.29, n.9, p.1143-1156, 2002b.