Contribuições para o Problema de Redução de Ciclos de Serra

Rodolfo Ranck Junior, Horacio Hideki Yanasse, José Carlos Becceneri

Instituto Nacional de Pesquisas Espaciais – INPE
Caixa Postal 515 – 12.227-010 – São José dos Campos – SP – Brazil

rodolforanck@yahoo.com.br, {horacio, becce}@lac.inpe.br

Abstract. *In this work we propose a heuristic to solve the problem of reducing saw cycles. This problem arises in cutting settings where the time used by the saw machine contributes significantly to the production costs, and/or in a situation with high demands where the saw machine has to be used efficiently increasing the number of items produced per unit of time. The heuristic is based on the Yanasse et al (1993) idea of producing cutting patterns that can be repeated many times, preferably in quantities that are a multiple of the saw machine capacity and in a linear integer programming model based on Mosquera (2007).*

Resumo. *Neste trabalho, propomos uma heurística para resolver o Problema de Redução de Ciclos de Serra. Este problema surge em ambientes de corte cujo tempo de funcionamento da máquina de corte contribui significativamente para os custos de produção, e/ou em uma situação de alta demanda em que é preciso utilizar a serra de maneira eficiente aumentando o número de itens produzidos por unidade de tempo. A heurística é baseada na idéia de Yanasse et al (1993) de produzir padrões que podem ser repetidos várias vezes, de preferência em quantidades múltiplas da capacidade da serra e em um modelo de programação linear inteira baseado em Mosquera (2007).*

1. Introdução

Considere uma máquina de corte que pode cortar um ou mais objetos juntos de cada vez. Este objeto, ou conjunto de objetos, é cortado na máquina segundo um determinado padrão, todos juntos, de uma vez, conforme um plano de corte previamente estabelecido. Assim, se um padrão de corte é repetido diversas vezes em um plano de corte, podem-se cortar nesta máquina até um máximo de Cap objetos de cada vez, todos eles segundo um mesmo padrão. Cap é a capacidade da serra, ou seja, o número máximo de objetos que podem ser cortados de uma vez pela serra.

Um ciclo de serra corresponde ao tempo gasto pela máquina de corte para cortar completamente um ou mais objetos juntos segundo um padrão. Admite-se que o ciclo de serra não varie muito com a quantidade de objetos sendo cortados juntos ou com o padrão sendo cortado. Desta forma, para reduzir o tempo total utilizado pela máquina de corte basta minimizar o número de ciclos de serra necessário para atender à demanda.

No problema de corte de estoque deseja-se minimizar custos de produção. Uma das parcelas deste custo é dada pelo número de objetos utilizados para atender a uma carteira de pedidos; outra parcela é dada pela operação da máquina de corte, cujo custo, estamos admitindo ser proporcional ao tempo de uso. O problema de redução de ciclos de serra (PRCS) foi abordado em Yanasse et al (1993) que sugeriu uma heurística que gera, inicialmente, padrões em quantidades múltiplas da capacidade da serra e, caso isto não seja mais possível, estas quantidades são reduzidas até que toda a demanda seja satisfeita.

Uma maneira indireta de minimizar o número de ciclos de serra é reduzir o número de padrões distintos em um plano de corte. Ao se reduzir padrões e ao mesmo tempo atender à demanda, vários objetos podem ser cortados juntos de uma vez dado que cada padrão passa a ser repetido mais vezes. O problema de redução de padrões foi estudado, por exemplo, por Haessler (1975), Foerster e Wäscher (2000), Vanderbeck (2000), Salles Neto (2005), Yanasse e Limeira (2006), entre outros.

Mosquera (2007) propôs um modelo matemático baseado na proposta de Yanasse et al (1993) para tentar resolver o PRCS. Neste modelo impõe-se que todos os padrões produzidos sejam cortados um número múltiplo da capacidade da serra:

$$\text{Min } z = \sum_{j=1}^{n} x_j \tag{1}$$

$$\text{s.a.} \quad \sum_{j=1}^{n} a_{ij}\, x_j \ \geq \ b_i, \quad i = 1, \dots, m \tag{2}$$

$$x_j = \ Cap \cdot y_j, \quad j = 1, \dots, n \tag{3}$$

$$x_j, y_j \ \in \ \mathbb{N}, \quad j = 1, \dots, n \tag{4}$$

em que:

b_i é a demanda do item i;
a_{ij} é a quantidade de itens do tipo *i* no padrão *j*;
x_j é a quantidade de vezes que o padrão *j* é repetido;
y_j é a quantidade de ciclos gastos pelo padrão *j*.

2. O método proposto para resolver o PRCS

O PRCS pode ser formulado da seguinte forma:

$$\text{Min } z = \sum_{j=1}^{n} \sum_{\alpha=0}^{cap-1} x_{\alpha j} \tag{5}$$

$$\text{s.a.} \quad \sum_{j=1}^{n} \sum_{\alpha=0}^{cap-1} a_{ij} x_{\alpha j} \,(cap - \alpha) = \ b_i, \quad i = 1, \dots, m \tag{6}$$

$$x_{\alpha j} \ \in \ \mathbb{N}, \ \ j = 1, \dots, n; \ \ \alpha = 0, \dots, cap - 1 \tag{7}$$

em que:

$x_{\alpha j}$ é a quantidade de ciclos gastos pelo padrão *j* com (*cap* - α) objetos cortados simultaneamente.

Neste modelo minimiza-se o número de ciclos permitindo que 1, 2,..., *Cap* objetos sejam cortados juntos em um ciclo da serra, o que não era permitido no modelo de Mosquera (2007).

Neste modelo existem várias colunas linearmente dependentes e, portanto, se utilizarmos a relaxação do problema e resolvermos pelo método simplex com geração de colunas conforme Gilmore e Gomory (1963), estas colunas não poderão estar simultaneamente na base. Frente a isso, propomos neste trabalho uma heurística em que a cada iteração, um problema é resolvido para uma quantidade M de objetos sendo cortados juntos, a começar pelo valor de Cap. Conforme a demanda é atendida, a demanda residual é atualizada e um novo problema é considerado. A cada novo problema, a quantidade de objetos sendo cortados juntos é reduzida de uma unidade.

Cada um dos problemas é resolvido utilizando o método simplex com geração de colunas, portanto, não há garantias de que a solução obtida seja inteira. Assim, utilizamos a heurística residual proposta por Poldi (2003) para gerar uma solução inteira para o problema. Desta solução, padrões que não atingem um determinado nível de eficiência pré-estabelecido são excluídos.

Seja P(b') o problema de programação linear inteira dado por:

$$\text{Min} \quad \sum_{j=1}^{n} y_j \tag{8}$$

$$\text{s.a.} \quad \sum_{j=1}^{n} a_{ij}\, y_j = b'_i, \qquad i = 1, \dots, m \tag{9}$$

$$y_j \in \mathbb{N}, \; j = 1, \dots, n \tag{10}$$

A descrição do método proposto com todos os passos para a solução do PRCS é apresentada a seguir:

1.Inicialização:

$M = cap$;

$d_i = b_i$, $i = 1,..., m$.

Enquanto $d \neq 0$ faça:

Início_enquanto

2. $b'_i = \left\lfloor \frac{d_i}{M} \right\rfloor, i = 1, \dots, m$;
3. Resolva o problema P(b') relaxando a condição de integralidade e aplicando a técnica de geração de colunas;
4. Encontre uma solução inteira a partir da solução obtida conforme Poldi(2003);
5. Elimine os padrões da solução que não atingem o nível de aspiração pré-determinado;
6. Armazene os padrões restantes e o valor de M correspondente;
7. Atualize a demanda residual d (demanda ainda não atendida);
8. Reduza o valor de M em uma unidade.

Fim_enquanto

9. A solução final é dada pelos padrões armazenados e suas freqüências M correspondentes.

O nível de aspiração dos padrões

Seguindo o método de Yanasse et al (1993), o nível de aspiração (Passo 5) utilizado na heurística proposta é variável. Quando um padrão é utilizado muitas vezes, seu desperdício deve ser pequeno, já que será multiplicado pelo número de vezes que este padrão é empregado na solução. Por outro lado, se o padrão é utilizado poucas vezes, é possível ser mais tolerante, pois mesmo que o padrão apresente um desperdício grande, ele não contribui tanto para o desperdício global da solução.

São pré-estabelecidos dois níveis de aspiração para a heurística, o valor máximo e o valor mínimo. Valores maiores são utilizados para padrões que são repetidos mais vezes, valores menores para padrões que são repetidos menos vezes. O valor máximo para o nível de aspiração é utilizado para $M = Cap$. Para outros valores de M utiliza-se uma redução linear do valor máximo para o valor mínimo ($M = 2$). Quando $M = 1$, aceita-se todos os padrões gerados visto que a demanda precisa ser satisfeita.

4. Testes Computacionais

Os problemas testes foram gerados aleatoriamente utilizando-se o programa Cutgen1 de Gau e Wäscher (1995), com as características apresentadas na Tabela 1.

Tabela 1. Parâmetros utilizados nos testes computacionais

Teste	*Número de itens*	*Demanda média*	*Capacidades de serra utilizadas (Cap)*
1	10	100	5,10,20,30,50
2	10	1000	5,10,20,30,50
3	40	100	5,10,20,30,50
4	40	1000	5,10,20,30,50
5	60	5000	5,10,20,30,50

Para todos os testes foram utilizados os seguintes parâmetros:

L : tamanho da barra = 1000;
V1 : limitante inferior do tamanho dos itens = 1% do objeto;
V2 : limitante superior do tamanho dos itens = 20% do objeto;
SEED : semente = 1994;
ICPRB: número de problemas gerados = 20.

Os resultados dos testes computacionais para cada uma das 5 classes estão apresentados nas Tabelas 2 a 6. Nestas tabelas, HRC é a heurística para redução de ciclos proposta, CSP é o problema de corte de estoque resolvido apenas considerando a redução de objetos utilizados no atendimento da demanda e Lbound é o limitante inferior para o número de ciclos de serra baseado na solução relaxada obtida em CSP e é dado por Lbound = $\lceil N_{obj}/cap \rceil$, em que Nobj é o número de objetos mínimo para atender a demanda do problema em questão.

Os resultados apresentados são baseados em uma média de 20 execuções. O código foi escrito em linguagem C e foi executado em um computador com processador Intel Centrino Duo T2400, 1,83 Ghz operando com 1GB de memória RAM do tipo DDR2. O Sistema Operacional utilizado foi o Microsoft Windows XP.

Tabela 2. Resultados do Teste 1

Cap	***Nº OBJETOS***		***Nº CICLOS***			***Nº PADRÕES***		***TEMPO (s)***	
	HRC	**CSP**	**Lbound**	**HRC**	**CSP**	**HRC**	**CSP**	**HRC**	**CSP**
5	108,8	108,55	22	23,7	28,45	11	12,15	0,439	0,054
10	108,95	108,55	11	13,2	19,15	9,75	12,15	0,439	0,054
20	109,35	108,55	6	8,45	14,3	8	12,15	0,761	0,054
30	109,9	108,55	4	6,9	13,15	6,9	12,15	0,894	0,054
50	111	108,55	3	6,05	12,3	6,05	12,15	1,160	0,054

Tabela 3. Resultados do Teste 2

Cap	Nº OBJETOS		Nº CICLOS			Nº PADRÕES		TEMPO (s)	
	HRC	CSP	Lbound	HRC	CSP	HRC	CSP	HRC	CSP
5	1081,2	1080,8	217	218,2	222,9	15	14,25	0,697	0,260
10	1081,25	1080,8	109	110,7	116,4	14,8	14,25	0,759	0,260
20	1081,75	1080,8	55	57	63,5	14,1	14,25	0,996	0,260
30	1082,4	1080,8	37	39,4	45,8	13,95	14,25	1,072	0,260
50	1083,35	1080,8	22	25,4	31,6	12,75	14,25	1,511	0,260

Tabela 4. Resultados do Teste 3

Cap	Nº OBJETOS		Nº CICLOS			Nº PADRÕES		TEMPO (s)	
	HRC	CSP	Lbound	HRC	CSP	HRC	CSP	HRC	CSP
5	417,7	417,55	84	86,8	108,8	39,85	47,6	2,605	1,230
10	417,85	417,55	42	46,4	71,5	33,85	47,6	3,209	1,230
20	418,5	417,55	21	26,75	54	25,05	47,6	3,524	1,230
30	419,25	417,55	14	20,55	48,75	20,45	47,6	3,392	1,230
50	420,45	417,55	9	15,9	46,45	15,9	47,6	3,759	1,230

Tabela 5. Resultados do Teste 4

Cap	Nº OBJETOS		Nº CICLOS			Nº PADRÕES		TEMPO (s)	
	HRC	CSP	Lbound	HRC	CSP	HRC	CSP	HRC	CSP
5	4171,65	4171,65	835	837,5	860,85	53,6	51,55	2,669	1,215
10	4171,85	4171,65	418	421,6	447,35	51,75	51,55	2,808	1,215
20	4172,55	4171,65	209	214,4	241,85	49,3	51,55	3,145	1,215
30	4173,2	4171,65	140	145,6	172,75	47,05	51,55	3,790	1,215
50	4174,4	4171,65	84	90,85	120,2	43,95	51,55	3,983	1,215

Tabela 6. Resultados do Teste 5

Cap	Nº OBJETOS		Nº CICLOS			Nº PADRÕES		TEMPO (s)	
	HRC	CSP	Lbound	HRC	CSP	HRC	CSP	HRC	CSP
5	31578,55	31578,35	6316	6319,5	6353,5	79,7	75,8	5,563	2,637
10	31578,65	31578,35	3158	3163,1	3200,95	78,6	75,8	5,563	2,637
20	31579,6	31578,35	1579	1585,6	1625,8	78,7	75,8	6,620	2,637
30	31580,65	31578,35	1053	1060,5	1100,45	78,25	75,8	7,511	2,637
50	31582,8	31578,35	632	640,75	680	76,25	75,8	7,711	2,637

Das tabelas, observa-se, como esperado, que o número de ciclos de serra é reduzido gradativamente com o aumento de Cap. Observa-se que a redução de ciclos foi acompanhada de um ligeiro crescimento no número de objetos, (menos de 1% nos problemas grandes). A HRC gerou soluções com um número reduzido de ciclos, comparado com as soluções dadas pelo CSP. Além disso, comparando os valores com os limitantes inferiores obtidos, observa-se que a diferença não é grande (inferior a 10%).

Das tabelas verifica-se a correlação entre número de ciclos e número de padrões: ambos decrescem com o aumento da capacidade. No entanto, o número de padrões não necessariamente diminui com o aumento de Cap ou com a redução do número de ciclos (Tabela 6). Portanto, ao se minimizar o número de padrões distintos não necessariamente se minimiza o número de ciclos

Os tempos computacionais observados também não foram grandes. Para os problemas maiores (Teste 5) os maiores tempos foram inferiores a 8,0 segundos.

5. Considerações Finais

A heurística proposta apresentou um bom desempenho no conjunto de testes computacionais realizados, reduzindo, para todos estes casos, o número de ciclos de serra obtido pelo CSP. Os resultados, em média, estão bem próximos ao limitante inferior para o número de ciclos e foram obtidos com um baixo tempo computacional.

Dos resultados computacionais observou-se ainda que o número de padrões não necessariamente diminui quando o número de ciclos é reduzido. Portanto, minimizar o número de padrões não implica em minimizar o número de ciclos. O conjunto de testes pode ser ampliado para se ter uma avaliação mais robusta da heurística proposta.

Esta é uma primeira abordagem para resolução deste problema. Outras abordagens podem ser empregadas, por exemplo, tentar resolver o modelo proposto por meio de um procedimento tipo *branch and bound* ou variações. Isto, entretanto, será objeto de desenvolvimentos futuros.

6. Referências

Foerster, H. e Wäscher, G. (2000), Pattern reduction in one-dimensional cutting stock problems.*International Journal of Production Research 38*, 1657–1676.

Gau, T. e Wäscher, G. (1995), CUTGEN1: a problem generator for the standard one-dimensional cutting stock problem. *European Journal of Operational Research*, v. 84, n. 3, 572-579,

Gilmore, P. C. e Gomory, R. E. (1961), A linear programming approach to the cutting-stock problem. *Operations Research 9*, 849–859.

Gilmore, P. C. e Gomory, R. E. (1963), A linear programming approach to the cutting-stock problem - Parte II. *Operations Research 11*, 863–888.

Haessler, R. W. (1975), Controlling cutting pattern changes in one-dimensional trim problems. *Operational Research 23*, 483–493.

Mosquera, G. P. *Contribuições para o Problema de Corte de Estoque Bidimensional na Indústria Moveleira*. 146 p. Dissertação de Mestrado, IBILCE – UNESP – São José do Rio Preto, 2007.

Poldi, K. C. *Algumas extensões do problema de corte de estoque*. Dissertação de Mestrado, ICMC-USP, 2003.

Salles Neto, L. L. *Modelo Não-Linear para Minimizar o Número de Objetos Processados e o Setup num Problema de Corte Unidimensional*. Tese de Doutorado, UNICAMP, 2005.

Vanderbeck, F. (2000), Exact algorithm for minimizing the number os setups in the one-dimensionalcutting stock problem. *Operations Research, work paper version 48*, 915–926.

Yanasse, H.H., Harris, R.G., e Zinober, A.S.I. (1993), Uma heurística para redução do número de ciclos da serra no corte de chapas. *In Anais do XIII ENEGEP*, vol. 2, XIII ENEGEP/ I Congresso Latino Americano de Engenharia Industrial, 879-885.

Yanasse, H. H. e Limeira, M. S. (2006), A hybrid heuristic to reduce the number of different patterns in cutting stock problems. *Computers and Operations Research 33*, 2744–2756.