

UM ALGORITMO PARA GERAÇÃO DE PADRÕES DE CORTE OTIMIZADOS PARA O CORTE RADIAL DE TORAS EM SERRARIAS

Gilberto Vinicius Paulino Nunes

Departamento de Informática, Universidade Federal de Viçosa
Campus Univesitário da UFV, Av. PH Holfs , s/n. 36570-000 Centro Viçosa. MG, Brasil
gilbertovpn@gmail.com

José Elias Claudio Arroyo

Departamento de Informática - UFV
Campus Univesitário da UFV, Av. PH Holfs , s/n. 36570-000 Centro Viçosa. MG, Brasil
jose.e.arroyo@gmail.com

Helio Garcia Leite

Departamento de Engenharia Florestal - UFV
Campus Univesitário da UFV, Av. PH Holfs , s/n. 36570-000 Centro Viçosa. MG, Brasil
hglete@gmail.com

RESUMO

Neste artigo é considerado o problema de geração de padrões otimizados de corte para os tipos radial e paralelo no desdobro de toras em serrarias. Desenvolveu-se um algoritmo baseado em técnicas de otimização utilizadas por softwares que lidam com o corte do tipo tangencial. O algoritmo proposto foi avaliado com dados reais da indústria madeireira, comparado com os softwares Sawmodel e Maxitora e mostrou-se eficiente para o corte tangencial paralelo. No caso do corte radial, foi obtido um aproveitamento médio de 46,95% das toras levadas em consideração.

PALAVRAS CHAVE. Otimização Combinatória, Programação Dinâmica, Problema de Corte de Toras, Corte Radial, Corte Paralelo.

OC – Otimização Combinatória, PM – Programação Matemática

ABSTRACT

In this paper the optimized cutting patterns generation problem is considered to the quartersawn and plainsawn. We developed an algorithm based on optimizing techniques used by softwares to solve the plainsawn problem. The proposed algorithm was evaluated with real industry data, compared with Sawmodel and Maxitora systems and it was efficient to plainsawn problem. The adapted algorithm that solve the quartersawn problem reached a yield of 46.95% of the considered logs.

KEYWORDS. Combinatorial Optimization, Dynamic Programming, Cutting Log Problem, Quartersawn, Plainsawn.

Combinatorial Optimization, Mathematical Programming

1. Introdução

Em praticamente todos os processos industriais existe a necessidade de se tomar uma decisão. Seja para realizar cálculos de viabilidade de uma ideia de produto, para decidir a melhor forma de produzi-lo ou sobre como vendê-lo após a conclusão do processo produtivo.

Na prática, a maioria dos ramos da indústria lança mão de algum método eficiente (geralmente computacional) para obter auxílio nas principais decisões a serem tomadas. Entre tantos ramos da indústria, é destacada aqui a indústria madeireira.

O processo de condução da madeira, desde sua forma bruta na natureza até o produto final que chega ao consumidor, passa por vários estágios: plantio, condução, colheita e transporte de toras, classificação das toras, processamento, secagem, classificação e comercialização. O produto final desta atividade pode ser a fabricação de móveis, material de construção ou a obtenção de celulose para a fabricação de papel, entre outros derivados da madeira. Analisando o estágio chamado aqui de “processamento”, tem-se o desdobro de toras em madeira serrada.

As principais formas de processamento de toras são o Corte Paralelo (ou Tangencial) e o Corte Radial. A mais importante diferença entre estas formas de corte está na direção que os anéis de crescimento (veias) da árvore tomam na peça serrada, o que influencia diretamente a estabilidade, propensão a curvatura e torção do produto obtido.

O Corte Paralelo é o método mais simples e direto de corte de toras e consiste na realização de uma série de cortes paralelos ao longo da face da tora. (veja Figura 1A).

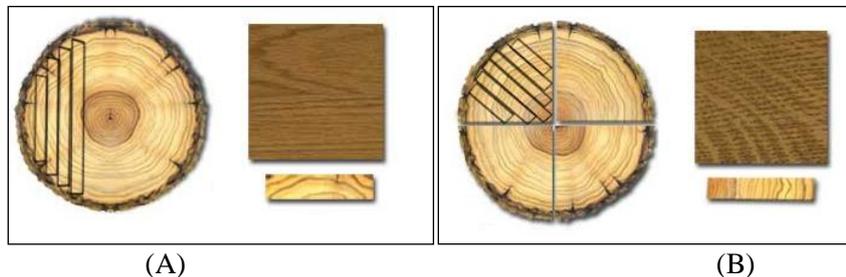


Figura 1 – (A) Corte Paralelo (B) Corte Radial (Fonte: amishhandcraftedheirlooms.com)

Devido às múltiplas direções dos anéis de crescimento, à medida que a peça seca, envelhece ou é utilizada, existe grande possibilidade de curvatura, torção e abaulamento da mesma. Em compensação, a aparência dos desenhos criados pelos anéis de crescimento estampada na peça gerada por este corte é bastante admirada.

O Corte Radial consiste em, primeiramente, dividir a face da tora em quatro partes iguais e a partir de cada uma destas, realizar cortes paralelos ao raio da tora (veja Figura 1B). Este método, gera peças mais estáveis, menos propensas aos problemas de curvatura, torção e abaulamento, além de ter baixa permeabilidade. O uso de madeira serrada desta forma é indicado para situações de uso extremas, como pisos de alto tráfego em restaurantes, bares ou cozinhas e para situações de alta tensão como é o caso de instrumentos de corda. Sua aparência estética ímpar também é muito admirada por atesãos e consumidores finais.

Ambos tipos de corte necessitam atenção especial do operador da serra a fim de reduzir desperdícios. Esta redução permite que a face da tora seja melhor aproveitada diminuindo a quantidade de toras e, conseqüentemente, de árvores abatidas para a produção de uma quantidade demandada de peças. Para isto, é fundamental que o operador seja guiado por algum Diagrama ou Padrão de Corte.

Tais diagramas são representações gráficas que especificam quais as peças e de que pontos da face de tora as peças devem ser retiradas. Na figura 2 ilustra-se um exemplo de diagrama de corte de tora. Utilizando-se um diagrama de corte otimizado, o operador da serraria é guiado na realização do corte com melhor aproveitamento da tora. A busca por estes diagramas pode ser estudada como um Problema de Corte e Empacotamento (PCE).

Neste trabalho é considerado o problema de geração de diagramas de corte otimizados

para desdobro de toras através do Corte Radial.

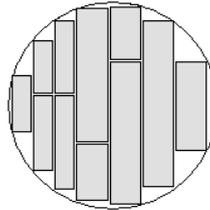


Figura 2 – Exemplo de um Diagrama de Corte. Fonte: Sawmodel.

2. Revisão de Literatura

Problemas de Corte e Empacotamento, PCE's, são amplamente estudados devido ao grande interesse prático e teórico. Algumas abordagens podem ser encontradas em Dyckhoff (1990), Dyckhoff e Finke (1992), Downsland e Downsland (1992), Sweeney e Paternoster (1992) e Morabito e Arenales (1992). Dyckhoff et al. (1997) e Lodi et al. (2002) apresentam extensa biografia sobre o tema. Na literatura, o PCE tem sido resolvido através de métodos exatos e metaheurísticas (Puchinger, 2011; Andrade 2006).

Geralmente, o estudo de soluções para o PCE se baseia na obtenção de peças retangulares obtidas a partir de objetos retangulares maiores (Problemas de Corte e Empacotamento Retangulares). Pode-se destacar exceções como o trabalho de Birgin et al. (2005) onde são encontradas soluções para o corte ou empacotamento de peças retangulares em regiões de formatos arbitrários e o de Birgin et al. (2003), onde foram obtidas peças circulares (ou cilíndricas) através de regiões retangulares.

Duas restrições específicas para o desenho de diagramas de corte para serrarias devem ser respeitadas. A mais simples e direta é a de que sejam empacotadas peças retangulares em regiões circulares (peças de madeira em face de tora). Além disto, devido ao fato de as toras serem geralmente desdobradas através de serras de fita ou serras circulares, os diagramas de corte devem obedecer uma organização baseada em estágios. Esta organização, por vezes, é também de chamada de organização por níveis ou por faixas. Stoyan e Chugay (2009) estudaram a primeira restrição citada ao tratar do problema do empacotamento de cilindros e paralelepípedos retangulares em regiões arbitrárias. A segunda restrição foi analisada por Ortmann et al. (2010), que apresentam novas e melhoradas heurísticas para corte e empacotamento retangular baseados em estágios.

Algoritmos e sistemas para a obtenção de diagramas de corte de toras otimizados são encontrados em Leite (1994) e Leite et. al. (2000) que utilizam Programação Dinâmica para gerar soluções. Na literatura, foram encontrados dois exemplos de softwares desenvolvidos no Brasil para a otimização na conversão de toras em madeira serrada. São eles o *Sawmodel* (Leite et al., 2000), e o *MaxiTora* (<http://www.optimber.com.br>). Nos dois sistemas, o PCE é dividido em fases resolvidas como Problema da Mochila e solucionado através do método de Programação Dinâmica, otimizando o corte em duas dimensões distintas, vertical e horizontal.

Não foram encontrados trabalhos específicos para a geração de diagramas otimizados para o Corte Radial. Desta forma, a seguir detalham-se as técnicas usadas nos dois sistemas nacionais citados (que lidam com o Corte Paralelo), que serão usadas para desenvolver o algoritmo proposto neste trabalho.

Maxitora

Por ser um software comercial, o modelo utilizado no Maxitora não está completamente disponível na literatura. Uma pequena descrição do processo de otimização empregado é feita por Novak (2007), como se segue.

Baseado em Programação Dinâmica, o sistema faz a análise de estágios e utiliza-se de uma equação de recorrência. Os estágios são posições que variam de 1 em 1 milímetro, a partir de uma posição zero e alcançam o diâmetro da face da tora (Figura 3).

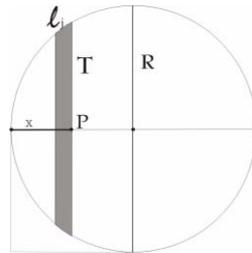


Figura 3. Esquema Maxitora. Fonte: Novak (2007)

Utiliza-se, como equação de recorrência, a expressão

$$f(x) = \max\{f(x - 1), f(x - l_j) + C_j\} \quad (1)$$

Onde a função $f(x)$ representa o ganho obtido com a porção da tora de tamanho x e C_j é o ganho obtido com uma fatia de espessura l_j . A obtenção de C_j é através da otimização unidimensional de um comprimento $2\sqrt{2Rx - x^2}$ (2), como descrito a seguir:

Analisando a Figura 3, temos que

$$\begin{aligned} (R - x)^2 + T^2 &= R^2 \\ T^2 &= R^2 - (R - x)^2 \\ T &= \sqrt{2Rx - x^2} \end{aligned}$$

Como T diz respeito a apenas metade da circunferência, a equação (2) é encontrada realizando a multiplicação $2T$. Ainda segundo Novak (2007), esta otimização é feita por Programação Dinâmica (PD) antecipadamente para as várias dimensões l_j e armazenada na memória.

Sawmodel

No sistema que possui o módulo Sawmodel, três níveis são considerados. O nível 1 contempla a transformação de troncos de árvores em toras, os níveis 2 e 3 tratam da conversão de toras em madeira serrada. Sendo assim, apenas os dois últimos são de interesse deste trabalho.

No Nível 2, as toras sofrem cortes paralelos e são transformadas em tábuas-mãe. O Nível 3 consiste em cortar as tábuas-mãe gerando o produto de interesse final. Desta forma, a PD funciona de forma análoga para os dois níveis em questão, bastando utilizar, no terceiro nível, as variáveis do algoritmo de forma perpendicular ao nível anterior. Em outras palavras, a PD é realizada considerando um eixo horizontal no Nível 2 e, em seguida, considerando um eixo vertical para o Nível 3.

Leite (1994) apresenta as funções de recorrência da seguinte forma:

Sendo,

- N_x = número de estágios no nível 2 (eixo x);
- k_x = intervalo de estágio no nível 2;
- $n_x k_x$ = variável de estado no nível 2, em mm;
- s_{kx} = espessura de corte no nível 2;
- $l_x[z]$ = variável decisória no nível 2, correspondente à espessura a ser cortada entre dois estágios consecutivos ou não, em mm;
- Z = número total de produtos;
- z = subgrupo de Z , com largura $\geq n_x k_x$ e/ou largura $\geq n_y k_y$;
- N_y = número de estágios no nível 3 (eixo y);
- k_y = intervalo de estágio no nível 3, em mm;
- $n_y k_y$ = variável de estado no nível 3, em mm;

- s_{ky} = espessura de corte no nível 3;
- $l_y[z]$ = variável decisória no nível 3, correspondente à largura a ser cortada entre dois estágios, consecutivos ou não, no nível 3, em mm;
- $F_{n_x}(n_x k_x)$ = valor máximo da seção da tora entre os estágios 0 e n_x , com intervalos k_x no nível 2;
- w = número do produto;
- $G_{n_y}(n_y k_y, n_x k_x, n_x k_x - l_x[z])$ = máximo valor da função para a espessura definida entre os estágios n_x e $n_x - \frac{l_x[z]}{k_x}$, como função da variável de estado $n_y k_y$, no nível 3;
- $G_{n_x}(n_y k_y, n_x k_x, n_x k_x - l_x[z])$ = máximo valor no estágio n_y da seção da tora, entre os estágios n_x e $n_x - \frac{l_x[z]}{k_x}$ no nível 2 e entre os estágios 0 e n_y no nível 3;
- $V(w, n_x k_x, n_y k_y, l_x[z], l_y[z])$ = valor do produto w , obtido nas coordenadas $n_x k_x$ e $n_y k_y$, de espessura $l_x[z]$ e de largura $l_y[z]$;

temos,

Nível 2 (Equação (3))

$$F_{n_x}(n_x k_x) = \max \left\{ G_{n_y}(n_y k_y, n_x k_x, n_x k_x - l_x[z]) + F_{n_x - \frac{l_x[z]}{k_x}}(n_x k_x - l_x[z] - k_x) \right\}$$

$$0 \leq n_x \leq N_x, z \in Z, f_0 = 0;$$

Nível 3 (Equação (4))

$$G_{n_y}(n_y k_y, n_x k_x, n_x k_x - l_x[z])$$

$$= \max \left\{ V(w, n_x k_x, n_y k_y, l_x[z], l_y[z]) + G_{n_y - \frac{l_y[z]}{k_x - s_{ky}}}(n_y k_y - l_y[z] - s_{ky}) \right\}$$

$$0 \leq n_y \leq N_y, G_0 = 0.$$

A avaliação dos produtos nos estágios é feita em ordem crescente do valor de espessura $l_x[z]$ e largura $l_y[z]$. Isto acontece pois, se num estágio não é possível inserir uma peça de uma determinada espessura ou largura, não será possível inserir nenhuma peça cujas dimensões sejam de tamanho maior que a peça citada anteriormente, sendo desnecessária a avaliação.

A maioria dos estudos sobre otimização no desdobro de toras consiste na implementação de um tipo de corte mais simples, que é o Corte Paralelo. Portanto, existem outros padrões que ainda não foram explorados no campo de programação matemática e informática. Assim, foi idealizado o presente estudo com o seguinte objetivo: considerando-se a união das técnicas empregadas nos sistemas Sawmodel e Maxitora, gerar um sistema capaz de obter diagramas de corte otimizados para Corte Radial.

3. Metodologia

Para alcançar o objetivo deste trabalho, primeiramente foi desenvolvido um algoritmo que gera diagramas de corte otimizados para o Corte Paralelo a fim de realizar uma avaliação comparativa com os sistemas Sawmodel e Maxitora (que também resolvem o problema para o Corte Paralelo) e em seguida, o mesmo algoritmo foi adaptado de forma a gerar as soluções para o Corte Radial. A seguir apresenta-se os detalhes do algoritmo para o Corte Paralelo de toras.

Como dados de entrada, são informados, em milímetros, o raio r da tora e as dimensões (altura h e largura l) de cada peça que se deseja obter a partir do corte. A partir destes, é função do algoritmo informar, como dados de saída, o diagrama de corte (similar ao apresentado na Figura 2), o valor de área total aproveitada da face da tora (soma da área de todas as peças do diagrama), o volume recuperado e o rendimento percentual obtido em relação ao volume total da tora.

Assim como acontece nos modelos Sawmodel e Maxitora, o algoritmo proposto neste trabalho é dividido em duas fases, vertical e horizontal. Na primeira fase, é realizada uma otimização unidimensional para uma barra de comprimento igual ao diâmetro D da face da tora. Os passos da Programação Dinâmica percorrem esta barra em variações de 1 em 1 milímetro e, para cada uma destas posições, é feita a otimização unidimensional levando em conta a altura h de cada uma das peças a serem produzidas.

A equação de recorrência utilizada, é a mesma do software Maxitora (Equação 1), ou seja:

$$f(x) = \max\{f(x - 1), f(x - l_j) + C_j\}$$

e esta equação garante o preenchimento da tabela da técnica PD.

Na tabela da PD, as linhas variam de zero a D , e em cada uma é armazenado o valor do ganho obtido e as peças que geraram tal ganho. Como a otimização visa o maior aproveitamento da área da face da tora, o valor de uma peça é obtido multiplicando sua altura pela sua largura, ou seja, a área da peça. Desta forma, o ganho obtido em cada estágio da otimização é calculado somando o valor (área) de cada peça escolhida no estágio em questão. É importante notar que, mesmo sendo esta fase de otimização unidimensional, o valor de cada peça é considerado em duas dimensões (área da peça). Isto é feito para evitar que peças de largura exageradas sejam consideradas boas opções para a segunda fase do algoritmo.

Uma vez completa, a tabela da PD é facilmente interpretada: para obter a informação de quais peças otimizam uma posição x do diâmetro, tal que $0 \leq x \leq D$, basta ler a linha x da tabela, onde estão armazenadas as informações necessárias.

Como já mencionado, as linhas armazenam as peças que geram o ganho obtido. Considerando essas peças dispostas uma acima da outra, temos que cada linha da tabela possui o que será chamado de “coluna de peças”.

Cada coluna de peças é considerada como uma nova peça criada de altura igual a soma das alturas das peças inseridas na coluna e de largura igual a maior largura dentre as peças inseridas em tal estágio. Estas peças, e não as peças informadas como dados de entrada, serão as consideradas pela fase 2 do algoritmo.

Neste ponto da fase 1, nenhuma tentativa de inserção de peça é descartada, ou seja, a cada tentativa de inserção, uma nova peça é criada, inserida em uma lista que armazena as peças que serão consideradas para a segunda fase e somente então são feitos os testes de otimização da equação de recorrência. Desta forma, sem comprometer a eficiência computacional do sistema na primeira fase, a segunda fase passa a ter um maior leque de escolha entre peças do que teria se apenas as peças ótimas da solução unidimensional fossem consideradas.

A segunda fase do algoritmo, a exemplo do algoritmo desenvolvido por Leite (1994), é feita considerando um giro de 90° na face da tora, ou seja, é feita a otimização perpendicular à primeira fase. Considerando apenas as peças novas criadas na primeira fase, usando a Equação (1) de recorrência e percorrendo-se o diâmetro da tora em estágios de 1 mm, uma nova tabela de PD é criada levando em conta algumas restrições.

Assim como feito pelos algoritmos usados como base, cada estágio tem uma altura T calculada através do Teorema de Pitágoras que gera a Equação (2) $T = 2\sqrt{2Rx - x^2}$ já detalhada neste trabalho. A cada iteração da PD, verifica-se se a peça cabe, em altura, no valor T daquele estágio e evita-se cálculos desnecessários para peças impossíveis de serem encaixadas no determinado ponto.

A nova tabela gerada nesta segunda fase traz, em cada linha x ($0 \leq x \leq D$), a coluna de peças mais interessantes a serem encaixadas no ponto x . Quando $x = D$, a linha da tabela informa o valor final da otimização, as peças que melhor aproveitam a área total da tora e em que ponto do diâmetro as peças devem ser encaixadas, gerando, assim, a solução do problema.

Uma importante diferença entre o algoritmo desenvolvido neste trabalho e os algoritmos dos softwares Maxitora e Sawmodel está no fato de o algoritmo deste trabalho não possuir a restrição de se encaixar, em uma mesma coluna de peças, apenas peças de larguras iguais. Testes foram realizados adicionando esta restrição, mas os resultados obtidos não foram tão satisfatórios quanto os obtidos quando permitido o encaixe de peças de diferentes larguras

numa mesma coluna de peças.

Na segunda fase, é realizada uma validação de largura das colunas de peças que serão encaixadas. Se uma coluna de peças possui largura l , altura h e está sendo avaliada a viabilidade de encaixá-la na posição x do diâmetro da tora, verifica-se se a faixa $x - l$ possui altura maior ou igual a h . Se isto não for verdadeiro, esta coluna de peças é descartada para o estágio x . Esta avaliação evita cálculos desnecessários durante o processo. Porém, é possível que sejam encaixadas colunas de peças que não aproveitem ao máximo, na segunda fase, a área disponível, gerando espaços vazios no diagrama de corte. Por isto, uma terceira fase foi considerada para o algoritmo desenvolvido neste trabalho, uma fase de pós-otimização.

Nesta terceira fase, é considerado o retângulo cuja altura é o valor que vai da parte superior de uma coluna de peças até tocar o limite da circunferência da face da tora e largura igual a largura da coluna de peças. Neste retângulo, é feita uma otimização semelhante a feita nas fases 1 e 2, considerando as peças disponibilizadas como dados de entrada e, se possível, uma ou mais peças são encaixadas neste espaço.

De posse do diagrama de corte final, calcula-se o volume recuperado (soma dos volumes das peças encaixadas) e calcula-se, como razão entre volume recuperado e volume total da tora, o rendimento obtido na otimização.

A partir do algoritmo gerado para resolver o problema do Corte Paralelo, foi desenvolvido também um algoritmo que gere soluções para o problema de Corte Radial. Este problema consiste em cortar a face em quatro partes iguais e retirar os subprodutos de madeira da peça obtida, o quarto de face de tora.

Para a obtenção de tal objetivo, foram realizadas adaptações geométricas no algoritmo apresentado acima a fim de garantir a consistência das soluções.

Primeiramente, o quarto de círculo é considerado na posição apresentada na figura abaixo, com o raio da tora na vertical.

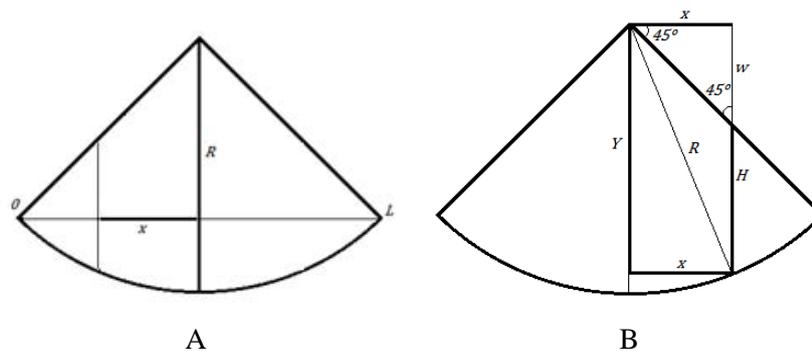


Figura 4– (A) Representação de um Quarto de Tora. (B) Cálculo da altura de uma faixa.

A tora é então dividida em estágios de 1 mm perpendiculares ao raio da tora, começando do ponto 0 (zero), chegando até o ponto L , especificados na figura 4A.

A exemplo do algoritmo que resolve o problema do Corte Paralelo, a primeira fase consiste na otimização unidimensional através da Equação (1) de recorrência. Porém, para o caso do Corte de Quarto de Tora, a barra unidimensional passa a ter o comprimento igual ao raio da tora e a partir da tabela de PD encontrada, é possível avaliar a otimização para cada faixa da tora a ser analisada na próxima fase do algoritmo.

Na segunda fase, a tora é percorrida na horizontal do ponto 0 até o ponto L e é avaliada como na segunda fase do algoritmo proposto para o Corte Paralelo.

Outra adaptação importante é a do valor T da altura de cada faixa encontrado pela Equação (2), que passa a ser referido por H e calculado da seguinte forma:

Considere a figura 4B, onde são especificados o valor x da posição da faixa que se deseja saber a altura H , R é o valor do raio da tora e Y é um cateto do triângulo retângulo de hipotenusa R .

Pelo Teorema de Pitágoras,

$$\begin{aligned} R^2 &= x^2 + Y^2 \\ Y &= \sqrt{R^2 - x^2} \quad (5) \end{aligned}$$

Note que, $Y = H + w$ e que o triângulo retângulo de catetos w e x é isósceles (ângulos internos de 45°), logo seus catetos são iguais, ou seja, $w = x$.

Assim, $Y = H + x$ e pela Equação (5) temos que a medida da altura H da faixa que corta a tora numa posição x é dada por

$$H = \sqrt{R^2 - x^2} - x \quad (6)$$

Consideremos agora $x = L$. Neste caso, podemos calcular o valor de L através do cosseno do ângulo de 45° . Seja,

$$\begin{aligned} \cos 45 &= \frac{L}{R} \\ L &= \frac{R\sqrt{2}}{2} \quad (7) \end{aligned}$$

Sendo assim, temos que o valor de x que percorre o quarto de tora em questão, varia entre 0 e $\frac{R\sqrt{2}}{2}$, ou seja, $0 \leq x \leq \frac{R\sqrt{2}}{2}$.

A terceira fase, pós-otimização é semelhante à apresentada para o algoritmo do Corte Paralelo, bastando apenas atentar para o fato de que o retângulo considerado passa a ter o valor de sua altura como sendo a dimensão que vai do ponto mais alto da coluna de peças e alcança a linha onde a tora sofreu o corte que a transformou em quarto de tora.

4. Resultados e Discussão

Em seu trabalho, Novak (2007) testou o sistema implementado utilizando dados reais de estoque e demandas obtidos de uma empresa especializada. Estes dados são usados para testar o algoritmo implementado neste trabalho.

Na ocasião, o estoque da empresa era composto de toras separadas por classes diamétricas, como é usualmente utilizado pela indústria. Essas classes foram divididas em diâmetros de 30, 32, 34, 36, 38 e 40 centímetros e possuíam um custo, segundo tabela abaixo.

Diâmetro (cm) d_k	Número de toras no estoque (unidade) E_k	Volume (m^3) V_k	Custo (tora) R\$ C_k
30	200	0,2119	12,70
32	150	0,2411	14,50
34	130	0,2725	16,40
36	120	0,3052	18,40
38	180	0,3400	20,40
40	170	0,3768	22,50
Total	950 unidades	-	-

Tabela 1: Toras disponíveis e custos (Fonte: Novak, 2007)

Quanto à demanda, quatro peças foram consideradas em quantidades reais de pedidos. Sendo N o número de peças de diferentes dimensões existentes no pedido, $w_i \times l_i$, as dimensões das peças a serem cortadas e q_N a quantidade de peças do tipo $w_i \times l_i$ a ser cortada para atender a demanda, a Tabela 2 mostra os valores referentes aos pedidos feitos à tal empresa.

Peças (cm) $w_i \times l_i$	Quantidade de peças q_N
2,5 x 12,5	1000
7,5 x 7,5	1000
5 x 6	1000
5 x 15	500

Tabela 2: Demanda de peças (Fonte: Novak, 2007)

De posse destes dados, Novak (2007) gerou dezesseis cenários distintos que serviram de base para a geração de possíveis padrões de corte, utilizando o software Maxitora. A sistemática utilizada consistiu em combinações dos quatro tipos de peças demandadas. No primeiro cenário, todas as quatro peças foram consideradas, ou seja, nos dados de entrada para o Maxitora foram inseridas as medidas de todas as peças e, então, gerado os padrões de corte. No segundo cenário, a peça 2,5 x 12,5 foi desconsiderada, e o software gerou os padrões de corte usando, como dados de entrada, as dimensões das peças 7,5 x 7,5, 5 x 6 e 5 x 15. No terceiro cenário, a única peça desconsiderada foi a de dimensões 7,5 x 7,5. Desta forma, foram gerados $2^N = 16$ cenários, como apresentados na Tabela 3, onde o símbolo “X” representa uma peça desconsiderada e o símbolo “x”, uma peça considerada.

1ª	2ª	3ª	4ª	5ª	6ª	7ª	8ª	9ª	10ª	11ª	12ª	13ª	14ª	15ª	16ª
x	X	x	x	x	X	X	X	x	x	x	x	X	X	X	X
x	x	X	x	x	X	x	x	X	X	x	X	x	X	X	X
x	x	x	X	x	x	X	x	X	x	X	X	X	x	X	X
x	x	x	x	X	x	x	X	x	X	X	X	X	X	x	X

Tabela 3. Cenários de geração de padrões de corte. (Fonte: Novak, 2007)

Para cada um destes cenários, foi gerado um padrão de corte para cada classe diamétrica. Logo, desconsiderando o décimo sexto cenário no qual nenhuma peça é utilizada, temos 15 cenários x 6 classes diamétricas = 90 padrões.

É importante saber que os padrões foram obtidos considerando que as toras possuem 2 cm de casca e as serras possuem espessura de 5cm vertical e horizontalmente.

A partir disto, um modelo matemático foi criado a fim de minimizar o custo da produção das peças demandadas usando padrões gerados por software.

Considerando k os tipos de toras separados em classes diamétricas, c_k representa o custo de toras do tipo k , x_j^k é a variável de decisão e representa o número de toras do tipo k que serão cortadas segundo o padrão de corte j .

Uma restrição do modelo diz respeito ao atendimento da demanda. Sendo a_{ij}^k o número de peças do tipo i produzidas pelo padrão de corte j gerado para toras do tipo k e q_i a quantidade de peças do tipo i demandada. Outra restrição visa atender ao estoque, ou seja, a quantidade de toras disponíveis.

O modelo completo é descrito a seguir

$$\text{Min } F = \sum_{k=1}^K \sum_{j=1}^{NP_k} c_k x_j^k$$

s.a.

$$\sum_k \sum_j a_{ij}^k x_j^k = q_i, \quad i = 1, 2, \dots, N$$

$$\sum_j x_j^k \leq E_k, \quad k = 1, 2, \dots, K$$

onde:

a_{ij}^k são os padrões de corte j da tora de diâmetro d_k ;
 x_j^k é o número de toras tipo k que serão cortadas segundo o padrão j ;
 q_i é a quantidade da demanda do tipo i , (matriz $m \times I$);
 E_k é a quantidade em estoque da matéria prima do tipo k (matriz $m \times I$);
 c_k é o custo da tora do tipo k , (matriz $I \times n$);
 $NP_k = n^\circ$ de padrões de corte para a tora tipo k , (índice).

Para fins de comparação, foram gerados os 90 padrões de corte referentes às peças demandas utilizando o software Sawmodel e utilizado o algoritmo proposto neste trabalho. Os dados obtidos foram submetidos à mesma otimização representada pelo modelo matemático discutido acima usando o software XPress. Os resultados são mostrados a seguir.

Para os dados utilizados por Novak (2007), não considerando a restrição dos valores de x_j^k serem inteiros:

- 164, 474 toras de 30 cm de diâmetro utilizando o padrão de corte 1;
 - 13,1579 toras de 36 cm de diâmetro cortadas de acordo com o padrão de corte 8;
 - 62,5 toras de 38 cm de diâmetro cortadas de acordo com o padrão de corte 1;
 - 46,0526 toras de 40 cm de diâmetro utilizando o padrão de corte 9;
- o que gera um custo de R\$ 4553,62.

Para os dados obtidos pelo software Sawmodel, temos:

- 6,85714 toras de 32 cm de diâmetro utilizando o padrão de corte 12;
 - 20 toras de 36 cm de diâmetro utilizando o padrão de corte 1;
 - 36 toras de 38 cm de diâmetro utilizando o padrão de corte 11;
 - 140 toras de 40 cm de diâmetro utilizando o padrão de corte 1;
- o que gera um custo de R\$ 4349,83.

Para os dados obtidos pelo algoritmo apresentado neste trabalho, temos:

- 45, 4545 toras de 32 cm de diâmetro utilizando o padrão de corte 12;
 - 15, 1515 toras de 36 cm de diâmetro utilizando o padrão de corte 9;
 - 60,6061 toras de 40 cm de diâmetro utilizando o padrão de corte 3;
 - 90,9091 toras de 40 cm de diâmetro utilizando o padrão de corte 5;
- o que gera um custo de R\$ 4345, 45.

Podemos notar que o algoritmo proposto gerou um custo 0,1% menor que o gerado pelo Sawmodel e 4,6% menor que o gerado pelo Maxitora.

Além desta comparação, foi ainda realizada uma comparação entre o rendimento obtido pelo software Sawmodel e pelo algoritmo deste trabalho. O Software Maxitora foi desconsiderado desta comparação pois o cálculo de rendimento é feito de maneira distinta.

O rendimento é a porcentagem do volume recuperado pelo padrão de corte em relação ao volume total da tora. Para saber o volume recuperado pelo padrão de corte, basta multiplicar a altura e a largura de cada peça obtida e multiplicar, ainda, o comprimento da tora. A soma dos resultados destas multiplicações de cada peça é o volume recuperado.

Usando os 90 padrões de corte já gerados, foi feita uma média do rendimento obtido para cada classe diamétrica por cada algoritmo. Os valores são apresentados na Tabela 4 onde cada célula de valor representa a porcentagem de rendimento média obtida.

Analisando os resultados, temos que o software Sawmodel apresentou um rendimento médio de 64,98% para todas as toras e o algoritmo proposto apresentou 65,72% de rendimento, ou seja, uma melhora de 1,14%.

	30 cm	32 cm	34 cm	36 cm	38 cm	40 cm
SAWMODEL	59,8802	66,3915	64,3163	62,5755	67,1782	69,5396
Algoritmo Proposto	60,8914	66,9237	64,0882	66,5217	66,7450	69,1578

Tabela 4: Rendimento Médio (%)

A partir dos resultado apresentados acima, atestamos a eficácia do algoritmo proposto para o Corte Paralelo. Sendo assim, geramos os resultados para o problema de Corte de Quarto de Tora, que passamos a apresentar.

Como no Corte de Quarto de Tora apenas a quarta parte da face circular da tora é utilizada, tem-se que uma peça cuja dimensão supere o valor do raio da tora é impossível de ser produzida. No intuito de utilizar dados reais, continuaremos com os mesmos valores de diâmetro de tora utilizados anteriormente, bem como o custo de cada uma. Porém, como algumas peças demandadas possuem dimensões grandes o suficiente para gerar problemas para serem obtidas por meio de um Corte de Quarto de Tora, realizaremos uma substituição nas dimensões destas peças e a tabela de demandas passa a ser:

Peças (cm) wi x li	Quantidade de peças Qn
3 x 6	1000
6 x 6.5	1000
7.5 x 7.5	1000
3 x 10	500

Tabela 5: Demanda de peças para Corte de Quarto de Tora

Tais dimensões foram geradas de acordo com a a norma NBR 7203 (1982) da ABNT. Foram escolhidas peças possíveis de ser obtidas com as toras em questão e também considerada a diversidade de produtos. De posse destes dados, o mesmo procedimento de avaliação usando no Corte Paralelo foi usando para o Corte de Quarto de Tora.

A tabela 6 apresenta o rendimento médio obtido pelo algoritmo que, após processado no modelo matemático acima apresentado, também no software Xpress, considerou a otimização para obter a demanda de peças utilizando:

- 500 quartos de toras de 30 cm de diâmetro utilizando o padrão de corte 1;
 - 250 quartos de toras de 38 cm de diâmetro cortadas de acordo com o padrão de corte 14;
 - 250 quartos de toras de 40 cm de diâmetro cortadas de acordo com o padrão de corte 1;
 - 125 quartos de toras de 40 cm de diâmetro utilizando o padrão de corte 13;
- o que gera um custo de R\$ 4905,63.

	30 cm	32 cm	34 cm	36 cm	38 cm	40 cm
Algoritmo Proposto	41,89667	44,30667	45,67933	46,98333	50,16267	52,692

Tabela 6: Rendimento Médio de quartos de toras (%)

5. Conclusões

Neste trabalho foi proposto um algoritmo baseado na técnica Programação Dinâmica para geração de digramas de corte otimizados a serem utilizados no desdobro de toras utilizando o Corte Radial. O objetivo deste problema é gerar representações gráficas que auxiliem um operador de serraria na criação de peças de madeira de forma a aproveitar ao máximo a tora e diminuir o desperdício.

O algoritmo proposto combina técnicas de otimização usadas nos sistemas nacionais Sawmodel e Maxitora que também são utilizados como parâmetro de comparação. Primeiramente, o sistema proposto foi criado para gerar diagramas para o Corte Paralelo e comparados com os sistemas já citados. Após, o sistema é adaptado para gerar diagramas otimizados para o Corte Radial.

Os testes computacionais foram realizados com dados reais de uma serraria, considerando-se demanda de peças, custo de matéria-prima (toras) e com o objetivo de minimizar o custo da produção. Pode-se concluir que o algoritmo proposto gera diagramas que encontram custos equivalentes ao software Sawmodel (custo 0,1% menor) e 4,6% menor que o gerado pelo Maxitora para o caso do Corte Paralelo, além de um rendimento 1.14% maior que o obtido pelo Sawmodel. No caso do Corte Radial, o rendimento médio das toras usadas como parâmetro foi de 46.95%.

6. Referências Bibliográficas

- Andrade, C.E.** (2006), Um algoritmo exato para o Problema de Empacotamento Bidimensional em Faixas, *Dissertação de Mestrado*, UNICAMP.
- Birgin, E.G., Martínez, J.M., Nishihara, F.H., Ronconi, D.P.** (2005) Orthogonal Packing of Rectangular items within Arbitrary Convex Regions by Nonlinear Optimization, *European Journal of Operational Research*, 33, 3535-3548.
- Birgin, E.G., Martínez, J.M., Ronconi, D.P.** (2003), Optimizing the Packing of Cylinders into a Rectangular Container: A Nonlinear Approach, *European Journal of Operational Research*, 160, 19-33.
- Dowsland, K. e Dowsland, W.** (1992). Packing problems. *European Journal of Operational Research*. 56, 2-14.
- Dyckhoff, H, Scheithauer, G., Terno, J.,** (1997), Cutting and Packing. Annotated Bibliographies in Combinatorial Optimization [edited by M. Amico, F.Maffioli and S. Martello], *John Wiley & Sons*, New York, 393-414.
- Dyckhoff, H.** (1990). A typology of cutting and packing problems. *European Journal of Operational Research*, 44, 145-159.
- Dyckhoff, H. e Finke, U.** (1992). Cutting and packing in production and distribution: Typology and bibliography, *Springler-Verlag Co*, Heildelberg.
- European Journal of Operational Research*, 141:241–252.
- Leite, H.G.,** (1994) Conversão de Troncos em Multiprodutos da Madeira Utilizando Programação Dinâmica. *Tese Doutorado*, Universidade Federal de Viçosa.
- Leite, H.G., Campos, J.C.C., Paula Junior, G.G.,** (2000), Dynatree: Modelo de Programação Dinâmica Para Avaliação de Multiprodutos de Povoamentos, *Software sem registro de patente*.
- Lodi, A., Martello, S., Monaci, M.,** (2002), Two-dimensional packing problems: A survey.
- Morabito, R. e Arenales, M. N.** (1992). Um exame dos problemas de corte e empacotamento. *Pesquisa operacional* 12(1), 1-20.
- Novak, R.S.,** (2007), Um Sistema de Otimização Aplicada ao Desdobro de Madeira, *Dissertação de Mestrado*, Universidade Federal do Paraná.
- Ortmann, F.G., Nthabiseng, N., van Vuuren, J.H.,** (2010), New and Improvised Level Heuristics for the Rectangular Strip Packing and Variable-sized Bin Packing Problems, *European Journal of Operational Research*, 203, 306-315.
- Puchinger, J.** (2011), Combining Metaheuristics and Integer Programming for Solving Cutting and Packing Problems, *Tese de Doutorado*, Viena University of Technology, 2011.
- Stoyan, Y., Chugay, A.** (2009) Packing Cylinders and Rectangular Parallelepipeds with Distances Between them into a Given Region, *European Journal of Operational Research*, 197, 446-455.
- Sweeney, P. e Paternoster, E.** (1992). Cutting and packing problems: A categorized, applicationoriented research bibliography. *The Journal of the Operational Research Society*. 43, 691-706.