

O PROBLEMA DE CORTE DE PLACAS DEFEITUOSAS

Andréa Carla Gonçalves Vianna

Departamento de Computação

Faculdade de Ciências

Universidade Estadual Paulista (UNESP)

Bauru – SP

vianna@fc.unesp.br

Marcos Nereu Arenales *

Departamento de Matemática Aplicada e Estatística

Instituto de Ciências Matemáticas e de Computação

Universidade de São Paulo (USP)

São Carlos – SP

arenales@icmc.usp.br

* *Corresponding author* / autor para quem as correspondências devem ser encaminhadas

Recebido em 12/2002; aceito em 02/2006 após 1 revisão

Received December 2002; accepted February 2006 after one revision

Resumo

O problema tratado neste trabalho consiste em cortar uma placa retangular em peças menores retangulares, de modo que a perda seja minimizada. A placa, entretanto, contém defeitos bem localizados. Propomos uma abordagem em *grafo E/OU* para representação das soluções possíveis e um método de enumeração implícita para determinar a solução ótima. Resultados computacionais demonstram a efetividade da abordagem.

Palavras-chave: problemas de corte e empacotamento; busca em grafo E/OU; otimização combinatória.

Abstract

This paper addresses to the two-dimensional cutting problem where a rectangular plate is to be cut into smaller rectangular pieces, in such a way as to minimizing the waste. However, the plate has a number of well-located defects. We propose an AND/OR-graph approach to represent the feasible solutions and a branch and bound method in order to search the graph. Computational experiments show that the approach is effective.

Keywords: cutting and packing problems; AND/OR-graph search; combinatorial optimization.

1. Introdução

O Problema de Corte, genericamente, consiste em cortar unidades maiores em unidades menores, otimizando uma determinada função (por exemplo, minimização da perda). Este tipo de problema aparece em diversos processos industriais, tais como no corte de bobinas de papel e alumínio, barras de aço, chapas metálicas e de madeira, placas de circuito impresso, caixas de papelão, rolos de tecido, entre outros. Em geral, os objetos são perfeitos e homogêneos, ou seja, é possível cortar itens em qualquer posição do objeto. Há casos em que isto não ocorre, por exemplo, uma chapa de madeira reconstituída com pequenos defeitos de fabricação.

Os problemas de corte com placas defeituosas foram pouco estudados na literatura. Carnieri *et al.* (1993) apresentaram um procedimento heurístico para a resolução de problemas com objeto defeituoso (veja apêndice).

Neste trabalho apresentamos a resolução do problema de corte bidimensional com placa defeituosa utilizando uma abordagem de solução baseada numa busca em grafo E/OU (Morabito, 1989; Morabito, 1992; Morabito *et al.*, 1992; Morabito & Arenales, 1994, 1996; Arenales & Morabito, 1995). Vianna (2000) estendeu a abordagem em grafo E/OU para tratar outros casos não analisados por trabalhos anteriores, envolvendo diferentes processos de corte. Esta abordagem utiliza uma técnica de busca híbrida (semi-informada), na qual se combina a busca em profundidade limitada com limite e a busca *hill-climbing*, utilizando também, heurísticas baseadas nos limitantes superiores e inferiores. Na seção 2 definimos o problema a ser estudado. Na seção 3 revisamos a representação das soluções (padrões de corte) como caminhos completos num grafo E/OU e na seção 4 apresentamos métodos de enumeração implícita baseados nesta representação. Experimentos computacionais são relatados na seção 5.

2. Problema de Corte Bidimensional com Placa Defeituosa

Considere uma placa retangular de tamanhos (L,W) , com L sendo o comprimento e W a largura e um conjunto de m peças retangulares de tamanhos (ℓ_i,w_i) , com ℓ_i sendo comprimento e w_i a largura da peça i , e um valor de utilidade v_i é associado a cada peça i , $i=1,\dots,m$ (Figura 1). Um problema de corte bidimensional consiste em cortar a placa retangular em peças menores, de forma a maximizar o valor de utilidade total (se $v_i=\ell_iw_i$, isto é, a área da peça i , então o objetivo é equivalente a minimizar a perda).

O modo como as peças estão arranjadas na placa é chamado de padrão de corte e limitaremos este trabalho aos padrões de corte guilhotinados, isto é, padrões que são obtidos por uma seqüência de cortes que produzem sempre dois novos retângulos (Figura 2). Note que é possível alocar peças em qualquer posição da placa, pois não há defeitos.

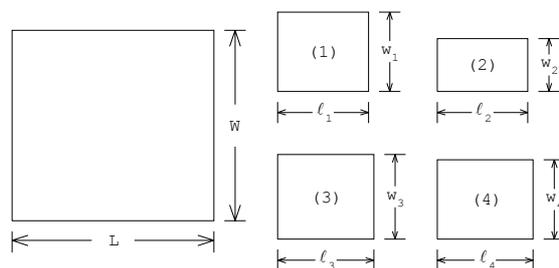


Figura 1 – Problema de Corte Bidimensional.

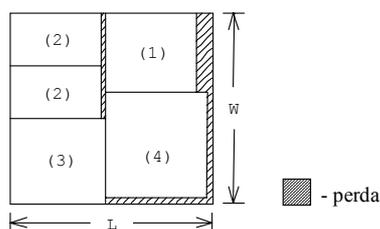


Figura 2 – Padrão de corte bidimensional.

Suponha agora que a placa retangular contenha um ou mais defeitos, ou seja, os padrões de corte devem ser gerados de forma a não alocar peças sobre o(s) defeito(s) (Figura 3). Neste trabalho, são tratadas as áreas defeituosas de maneira retangular (Figura 4).

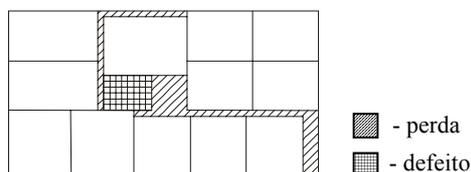


Figura 3 – Padrão de corte em uma placa defeituosa.



Figura 4 – Placa defeituosa e placa a ser tratada.

Note que o defeito fica bem localizado pelas coordenadas do canto inferior esquerdo (x_1, y_1) e do canto superior direito (x_2, y_2) .

A seguir apresentamos uma representação dos possíveis padrões de corte guilhotinados para uma placa com defeitos.

3. Representação dos Padrões de Corte em um Grafo E/OU

A abordagem em Grafo E/OU para a resolução de Problemas de Corte foi inicialmente proposta por Morabito (1989), para problemas de corte guilhotinado bidimensional irrestrito (o termo “irrestrito” é usado na literatura dos problemas de corte para designar que não há limitação sobre o número de itens num padrão de corte e “restrito” caso contrário) e não-estagiado (Morabito *et al.*, 1992). Morabito (1992) estendeu esta abordagem para problemas de corte guilhotinado irrestrito e restrito, considerando as dimensões unidimensional, bidimensional e tridimensional (Morabito & Arenales, 1994, 1996). Arenales (1993) generalizou a abordagem para problemas de corte diversos, incluindo cortes não-guilhotinados (Arenales & Morabito, 1995). Vianna (2000) estendeu a abordagem para diferentes processos de corte.

Um grafo E/OU pode ser definido para representar todos os possíveis padrões de corte, em que os nós representam retângulos (no caso do problema de corte bidimensional, com placa e peças retangulares) e os arcos representam cortes. Um arco (corte) estabelece uma relação entre um nó N do grafo (retângulo), com dois outros nós N_1 e N_2 (retângulos obtidos após o corte), portanto, um arco-E. Os nós N_1 e N_2 são chamados sucessores de N e, N predecessor de N_1 e N_2 . Os padrões de corte são gerados examinando-se todas as possibilidades alternativas de corte (daí, arcos-OU) e uma delas é reproduzir o próprio retângulo N (chamado de corte-0, lê-se corte zero), ao qual nenhum outro corte será feito, indicando o final do processo de corte. Um corte-0 é representado por um arco ordinário (isto é, aponta para um único nó). O nó inicial é representado pela placa (L,W) e os nós finais são aqueles originados de um corte-0 (sem perda de generalidade, associam-se aos retângulos finais um ou mais itens idênticos. Veja, por exemplo, o primeiro arco a esquerda emergindo da raiz na Figura 5). Os cortes (verticais ou horizontais) podem ser restritos, sem perda de generalidade, a um conjunto finito, chamado de “conjunto de discretização”, formado pelas combinações lineares não-negativas dos tamanhos dos itens (veja Herz (1972) e Morabito & Arenales (1995) para sua construção). A Figura 5 ilustra diversas seqüências de corte e, portanto, diversos padrões de corte. Por exemplo, três arcos-OU emergem da raiz, indicando alternativas diferentes para se obter um padrão de corte. Note que um padrão de corte é bem definido seguindo-se uma seqüência de arcos-E (cortes), a partir da raiz (placa inicial) até nós finais (nós após cortes-0). Esta seqüência é chamada de caminho completo e todo padrão de corte tem um caminho completo associado. Observe também que um mesmo nó pode pertencer a seqüências diferentes (isto é, retângulos de mesmos tamanhos podem ser obtidos por diferentes seqüências de corte), o que caracteriza um ciclo. Apesar disto, por simplicidade de implementação (limitação do uso de memória que pode ser explosivo), duplicamos tais nós (ou seja, nós diferentes podem representar retângulos idênticos, porém obtidos de seqüências de corte diferentes) e, portanto, trabalhamos com árvores (grafo conexo sem ciclos). O valor de utilidade do padrão de corte é a soma dos valores de utilidade dos nós finais do caminho completo associado.

Padrões de corte numa placa defeituosa são representados num grafo E/OU, análogo ao caso sem defeitos, com ligeiras modificações. Uma seqüência de cortes na placa defeituosa pode ser observada na Figura 6. Note que cada nó do grafo E/OU pode representar um retângulo defeituoso ou não e as coordenadas do defeito devem ser atualizadas após o corte. Por exemplo, considere que um retângulo (x,y) cujo defeito tem coordenadas (x_1,y_1) , canto inferior esquerdo, e (x_2,y_2) , canto superior direito, seja cortado verticalmente no comprimento x_1 . Assim, dois novos retângulos são obtidos, sendo que retângulo esquerdo não contém defeitos e o retângulo direito contém um defeito cujas coordenadas são: $(0,y_1)$ e (x_2-x_1, y_2) (veja o primeiro corte na Figura 6).



Figura 5 – (a) parte do grafo E/OU representando padrões de corte.
(b) padrão de corte representado em destaque em (a).

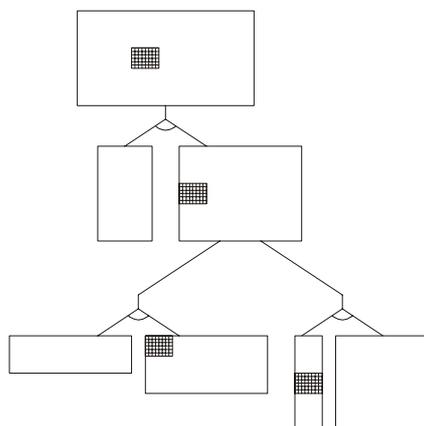


Figura 6 – Possível caminho (seqüência de corte) numa placa defeituosa.

É possível que se tenha mais do que um defeito numa placa e as atualizações de suas coordenadas são feitas de maneira análoga (Figura 7).

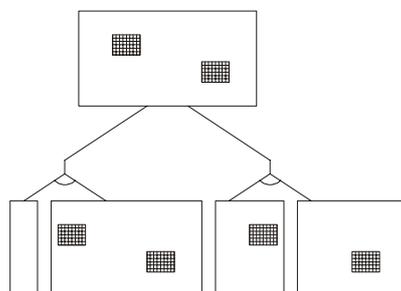


Figura 7 – Possíveis cortes numa placa com mais de um defeito.

O número de possíveis caminhos completos no grafo E/OU (ou seja, o número de padrões de corte) pode ser enorme e computacionalmente intratável. Na seção seguinte mostramos como enumerar implicitamente caminhos que não levam a uma solução ótima e como desenvolver buscas heurísticas baseadas em limitantes superiores e inferiores.

4. Método de Enumeração Implícita

Para a resolução do problema de corte com placa defeituosa é utilizado um método de enumeração implícita, baseado numa busca no grafo E/OU que descreve o espaço de soluções possíveis. A estratégia de busca combina busca em profundidade com busca informada (*hill-climbing*). Durante o processo de busca é possível reduzir o número de nós explicitamente gerados a um conjunto finito formado pelas combinações lineares inteiras não-negativas das dimensões das peças (Christofides & Whitlock, 1977). Enumerar explicitamente todos os caminhos do grafo, durante o processo de busca, é, na maioria dos problemas práticos, inviável. As soluções podem ser enumeradas implicitamente, ou seja, é possível descartar a expansão (i.e., o exame de seus sucessores) de um nó sem perder a solução ótima, usando limitantes. A seguir, apresentamos alguns limitantes para um retângulo qualquer representado num nó do grafo.

4.1 Limitante inferior

É usual em problemas de otimização (maximização), a determinação de limitantes inferiores pela obtenção de soluções viáveis. Nos problemas de corte, soluções viáveis são facilmente determinadas. Uma solução trivial para o subproblema do nó (x,y) pode ser obtida por preenchê-lo somente com peças iguais, conforme o padrão da Figura 8. Um padrão de corte que contém peças todas iguais é denominado padrão de corte homogêneo.

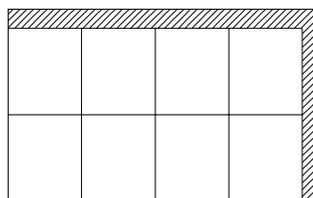


Figura 8 – Padrão de corte homogêneo.

Escolhendo-se o melhor dentre estes padrões, tem-se o limitante inferior dado por:

$$LI(x, y) = \max_{1 \leq i \leq m} \left\{ v_i \cdot \left\lfloor \frac{x}{\ell_i} \right\rfloor \cdot \left\lfloor \frac{y}{w_i} \right\rfloor \right\}$$

Quando se tem um problema com um nó representando um retângulo defeituoso de dimensão (x,y) e um defeito nas coordenadas (x_1,y_1) e (x_2,y_2) , podemos definir soluções viáveis e triviais evitando-se o defeito.

Soluções viáveis associadas a um retângulo com defeito podem ser obtidas preenchendo-se os retângulos formados ao redor do defeito, como, por exemplo, ilustrado na Figura 9. Há várias outras maneiras de definir os retângulos ao redor do defeito, mas o objetivo aqui é

simplesmente obter um limitante inferior trivial. As outras maneiras surgem durante a busca, após uma seqüência de cortes (da mesma forma que o próprio desenho na Figura 9a surgiria numa seqüência de dois cortes verticais).

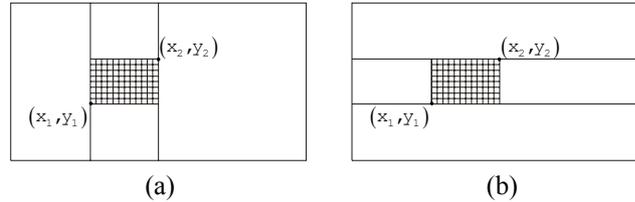


Figura 9 – Partições simples de uma peça com defeito obtendo retângulos sem defeitos.

Cada retângulo sem defeito da Figura 9 é preenchido com o maior número de peças iguais (solução homogênea), o que define uma solução homogênea composta (i.e., uma composição de várias soluções homogêneas). Assim sendo, um limitante inferior de um problema com placa defeituosa é dado por:

$$LI_D = \max \left\{ \begin{array}{l} LI(x_1, y) + LI(x_2 - x_1, y - y_2) + LI(x_2 - x_1, y_1) + LI(x - x_2, y), \\ LI(x, y_1) + LI(x_1, y_2 - y_1) + LI(x - x_2, y_2 - y_1) + LI(x, y - y_2) \end{array} \right\}$$

O procedimento é análogo quando se têm mais defeitos sobre o retângulo, definindo-se faixas horizontais ou verticais em torno dos defeitos.

4.2 Limitante superior

Como é usual em problemas de otimização (maximização), um limitante superior é obtido pela relaxação do problema.

Um limitante superior para o nó (x, y) é definido relaxando-se o problema de corte, considerando-se apenas que a área dos itens alocados não exceda a área do retângulo, ou seja,

$$\begin{aligned} LS(x, y) = \text{maximizar} \quad & \sum_{i \in A(x, y)} v_i \cdot a_i \\ \text{sujeito a:} \quad & \sum_{i \in A(x, y)} (\ell_i \cdot w_i) \cdot a_i \leq (x \cdot y) \\ & a_i \geq 0, \quad i \in A(x, y) \end{aligned}$$

em que $A(x, y) = \{i \mid \ell_i \leq x \text{ e } w_i \leq y\}$, com a seguinte solução trivial

$$LS(x, y) = \begin{cases} \max_{i \in A(x, y)} \left\{ v_i \cdot \left(\frac{x}{\ell_i} \right) \cdot \left(\frac{y}{w_i} \right) \right\} \\ 0, \text{ se } A(x, y) = \emptyset \end{cases}$$

Este limitante foi inicialmente utilizado por Herz (1972) e mais tarde por Morabito *et al.* (1992).

Para o problema de corte com placa defeituosa, o limitante superior de um nó (x, y) é calculado pela sua área, ou seja, a área total do nó (x, y) , decrementada das áreas defeituosas.

4.3 Estratégia de busca

Uma estratégia de busca utilizada por Morabito (1989, 1992) para resolver problemas de corte e empacotamento, consistiu na combinação de duas estratégias básicas: *backtracking* (BT) e *hill-climbing* (HC). *Backtracking* é uma variação da estratégia de busca em profundidade (*depth-first*), que consiste em ramificar primeiro o nó gerado mais recentemente, de modo que todos os caminhos do grafo sejam percorridos. *Hill-climbing*, por outro lado, é uma estratégia de busca miope, baseada em otimização local, que escolhe a melhor solução encontrada entre os sucessores de um nó, descartando os demais. Esta escolha é baseada numa avaliação dos limitantes inferior e superior da solução.

Estas duas estratégias básicas podem ser combinadas, buscando uma solução mais promissora dentre descendentes de um nó, um pouco além de seus sucessores imediatos. Para isto, é necessária a noção de profundidade de um nó no grafo. A profundidade de um nó no grafo é definida tal que, o nó raiz tem profundidade zero e, a profundidade de qualquer outro nó é igual à profundidade de seu predecessor imediato, acrescido de 1. As estratégias podem, então, ser combinadas, de modo que BT investiga todos os possíveis caminhos até uma profundidade máxima e, HC escolhe o mais promissor, descartando-se os demais (se a profundidade máxima for igual a 1, então tem-se HC pura, se ilimitado, tem-se BT puro). Cada nó no caminho mais promissor com profundidade máxima é novamente investigado por BT até a profundidade máxima, e assim por diante.

Considere um nó N , seus nós sucessores N_1 e N_2 . Seja $v(N)$ o valor do nó N dado por uma solução viável (por exemplo, a solução homogênea, ou outra melhor quando disponível) e, $LI(N)$ e $LS(N)$ o limitante inferior e superior do nó N , respectivamente. Durante a expansão deste nó, são utilizados os limitantes inferiores e superiores para evitar a geração de nós desnecessários, descartando caminhos não promissores. Assim,

- se $v(N)=LS(N)$, então a solução que fornece $v(N)$ é solução ótima para o nó N .
- se $v(N)<LI(N_1)+LI(N_2)$, então o valor do nó N é atualizado por $v(N)=LI(N_1)+LI(N_2)$ (uma nova solução viável é fornecida pela composição das soluções de nós sucessores).
- se $v(N)\geq LS(N_1)+LS(N_2)$, ou seja, se o valor de uma solução viável no nó N é melhor que a soma dos limitantes superiores de seus sucessores, então não é necessário investigar os nós N_1 e N_2 .

Observe que o valor de um caminho completo é a soma dos valores dos nós finais e, o melhor caminho a partir do nó N é determinado por aquele que apresenta o melhor valor $v(N)$.

Note que, com o limitante na profundidade do grafo e a estratégia *hill-climbing*, a otimalidade do problema de corte pode ser perdida.

A seguir, é apresentado o algoritmo para esta estratégia de busca híbrida.

4.4 Algoritmo BT-HC

Passo 0

Considere a placa (L,W) , as peças (ℓ_i,w_i) , $i=1,\dots,m$, e MP , a profundidade máxima permitida na estratégia de busca.

Considere o nó raiz como o nó que contém as informações da placa inicial.

Passo 1 – Backtracking

Aplique a estratégia de busca *backtracking* com profundidade MP a partir do nó raiz, armazenando o caminho que apresenta o melhor valor para este nó.

Passo 2 – Hill-Climbing

Para cada nó final (profundidade MP) do caminho gerado pela estratégia *backtracking*, verifique se é possível expandi-lo. Se for, retorne ao Passo 1, considerando-o como nó raiz.

4.5 Heurísticas

Para reduzir o espaço de busca, recorre-se a algumas heurísticas que podem ser aplicadas no algoritmo, de forma a diminuir o espaço de busca (Morabito *et al.*, 1992; Arenales & Morabito, 1995; Morabito & Arenales, 1996, Vianna, 2000). A seguir são apresentadas duas heurísticas utilizadas no algoritmo do Problema de Corte Bidimensional com Placa Defeituosa.

Considere um nó qualquer N e seus sucessores N_1 e N_2 .

- H1 (Morabito *et al.*, 1992)

Se $(1 + \lambda_1) \cdot v(N) \geq LS(N_1) + LS(N_2)$ com $\lambda_1 > 0$, os nós N_1 e N_2 são descartados como não-promissores.

Esta heurística faz uma aposta de que se os limitantes superiores para os nós sucessores (que não é necessariamente um limitante superior para o nó N) não forem muito melhores que o valor da solução viável já disponível no nó, então isto é uma indicação de que este não é um bom caminho a ser investigado.

- H2 (Morabito *et al.*, 1992)

Se $\lambda_2 \cdot LI(N) \geq LI(N_1) + LI(N_2)$ com $0 < \lambda_2 < 1$, os nós N_1 e N_2 são descartados.

Esta heurística faz uma aposta de que se a solução composta pelas soluções homogêneas nos nós sucessores não for razoavelmente melhor que a solução homogênea do nó N , então este caminho não é promissor.

5. Experimentos Computacionais

A implementação computacional foi desenvolvida na linguagem Pascal, utilizando o compilador Borland Pascal versão 7.0, em um microcomputador Pentium 4, 1.70 GHz, 512 Mbytes de RAM.

Na implementação deste problema não foi considerado o caso restrito (i.e., uma quantidade máxima de cada tipo de peças), mas a implementação da abordagem em grafo E/OU permite este ajuste incluindo informações adicionais em cada nó (Morabito & Arenales, 1995; Vianna *et al.*, 2000).

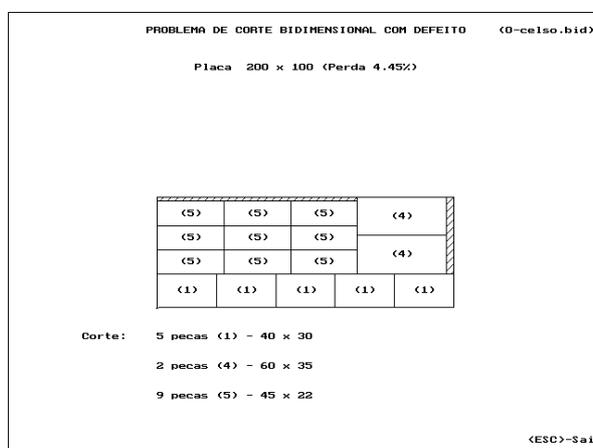
Durante a implementação levou-se em conta a orientação das peças na placa, de modo a não permitir rotações de peças.

A seguir, são mostrados os resultados obtidos com o exemplo apresentado por Carnieri *et al.* (1993), descrito no apêndice, em uma placa de dimensões (200,100) e 5 peças conforme os dados da Tabela 1.

Tabela 1 – Exemplo de Carnieri *et al.* (1993).

I	ℓ_i	w_i	v_i
1	40	30	10
2	68	26	12
3	50	20	8
4	60	35	18
5	45	22	9

O padrão ótimo de corte obtido para este exemplo sem defeito é apresentado na Figura 10 e o valor de utilidade total é 167. Utilizando a abordagem descrita neste trabalho, este exemplo foi executado em 0.22s e foram gerados 2649 nós.

**Figura 10** – Padrão ótimo para o problema sem defeito com valor de utilidade igual a 167.

Foram realizados 8 testes com este exemplo, mudando a posição do defeito. A posição dos defeitos, assim como os resultados obtidos por Carnieri *et al.* (1993), são apresentados na Tabela 2.

Tabela 2 – Defeitos e soluções obtidas por Carnieri *et al.* (1993).

Exemplo	Coordenadas do defeito	Solução
1	(100,50) (105,54)	166
2	(100,40) (105,44)	160
3	(100,60) (105,66)	162
4	(125,20) (132,30)	158
5	(125,71) (132,79)	164
6	(30,30) (40,40)	164
7	(80,40) (110,58)	157
8	(80,40) (118,58)	154

A abordagem em grafo E/OU, com os parâmetros $\lambda_1=0.01$ e $\lambda_2=0.95$, obteve os resultados apresentados na Tabela 3; o número de nós percorridos e o tempo computacional também são apresentados. Note que, em todos os exemplos o resultado obtido com a abordagem em grafo E/OU foi maior ou igual aos resultados apresentados pela Tabela 2.

Tabela 3 – Soluções obtidas por grafo E/OU.

Exemplo	Número de nós	Solução	Tempo (s)
1	9137	166	4.61
2	7591	160	3.57
3	7713	162	4.40
4	13197	160	6.15
5	21985	164	13.51
6	3239	164	1.32
7	14947	158	12.47
8	10075	154	8.07

As Figuras 11, 12 e 13, apresentam os melhores padrões de corte obtidos em alguns exemplos da Tabela 3.

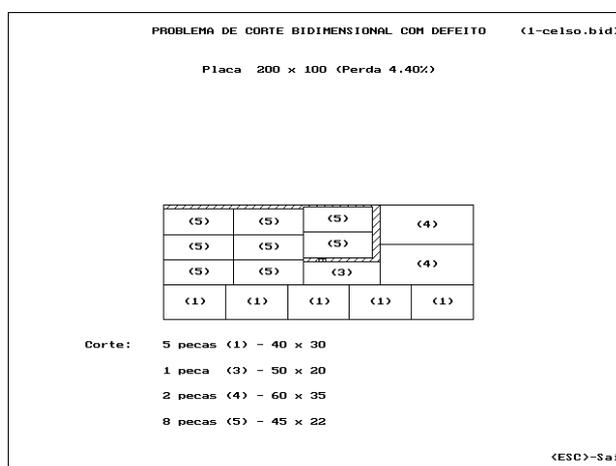


Figura 11 – Exemplo 1 (valor de utilidade=166).

Observe que a solução obtida para o exemplo 1, embora tenha um valor de utilidade menor que a solução sem defeito (Figura 10), apresenta uma perda menor, pois o objetivo consiste em maximizar o valor de utilidade.

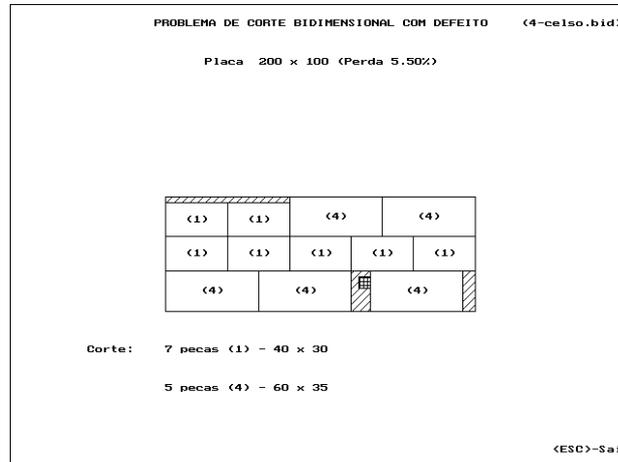


Figura 12 – Exemplo 4 (valor de utilidade=160).

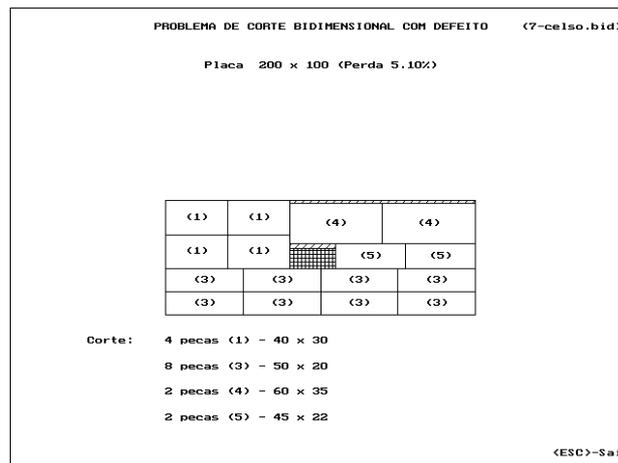


Figura 13 – Exemplo 7 (valor de utilidade=158).

A seguir apresentamos outros exemplos gerados com as informações da Tabela 1, contendo vários defeitos na placa. A Tabela 4 mostra seis novos exemplos, com as coordenadas dos erros (cantos inferior esquerdo e superior direito), bem como a solução encontrada pela abordagem proposta.

Tabela 4 – Exemplos com vários defeitos e soluções.

Exemplo	Coordenadas dos defeitos	Solução
9	(0,0) (5,5) (0,95) (5,100) (194,0) (200,10) (194,90) (200,100)	153
10	(95,0) (105,10) (95,90) (105,100) (0,45) (10,55) (190,45) (200,55)	148
11	(95,0) (105,10) (95,90) (105,100) (0,45) (10,55) (190,45) (200,55) (120,60) (125,65)	143
12	(0,0) (5,5) (30,15) (33,19) (60,47) (66,50) (198,45) (200,55)	150
13	(15,10) (20,15) (15,85) (20,90) (185,10) (190,15) (185,85) (190,90)	142
14	(0,45) (10,55) (190,45) (200,55)	160

As Figuras 14 – 19 ilustram os padrões de corte obtidos. Observe como as perdas crescem quando os defeitos são espalhados pela placa.

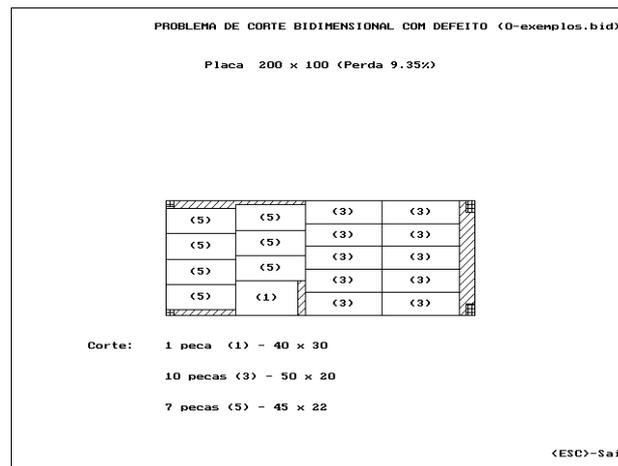


Figura 14 – Exemplo 9 (valor de utilidade=153).

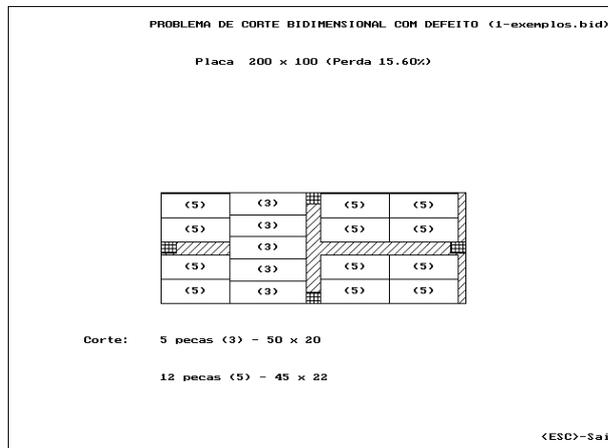


Figura 15 – Exemplo 10 (valor de utilidade=148).

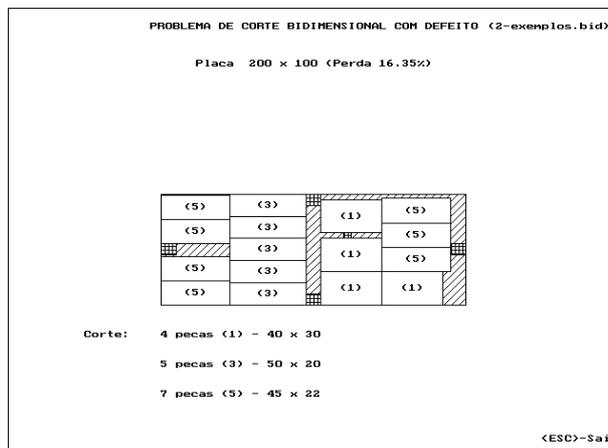


Figura 16 – Exemplo 11 (valor de utilidade=143).

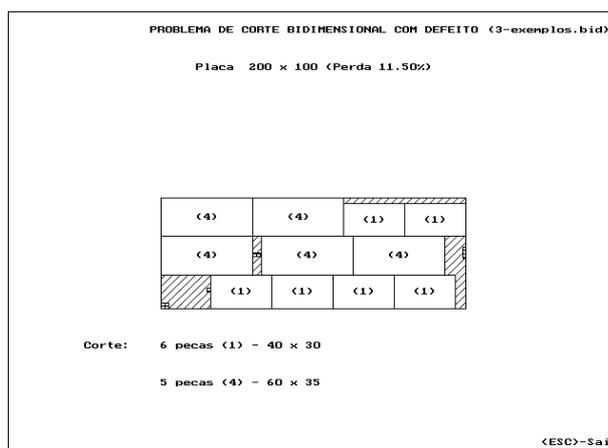


Figura 17 – Exemplo 12 (valor de utilidade=150).

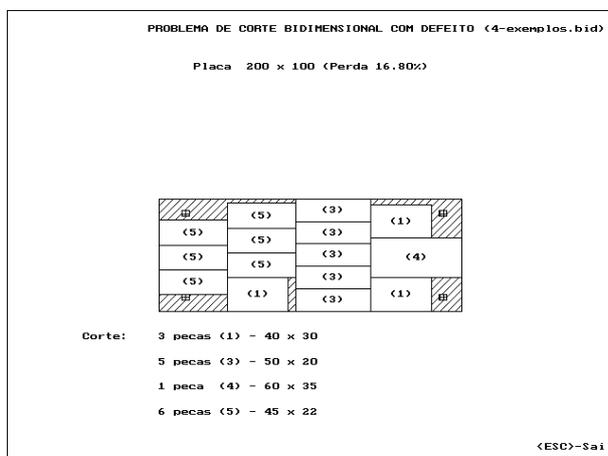


Figura 18 – Exemplo 13 (valor de utilidade=142).

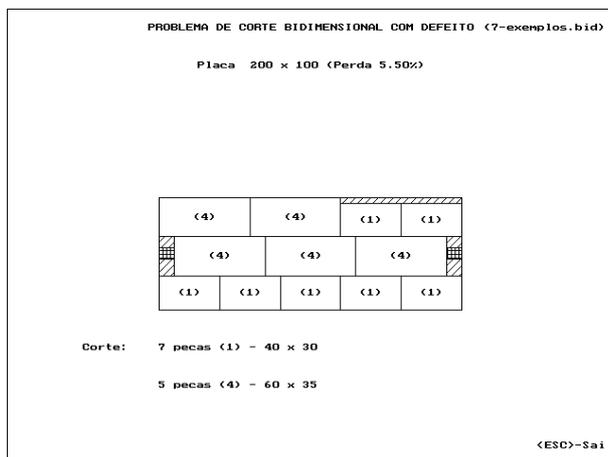


Figura 19 – Exemplo 14 (valor de utilidade=160).

6. Conclusões

Apresentamos neste trabalho um método de solução para o Problema de Corte Bidimensional com placa defeituosa (problema pouco estudado na literatura), estendendo a abordagem em grafo E/OU, a qual foi anteriormente desenvolvida para problemas retangulares, sem a presença de defeitos. Mostramos que a extensão é obtida sem muitas modificações e os experimentos computacionais demonstram ser esta uma ferramenta adequada de resolução de problemas de corte, bem flexível, permitindo que restrições adicionais (neste trabalho, defeitos) possam ser incluídas no processo de busca. Extensões da abordagem em grafo E/OU a outros problemas de corte (com diferentes formas, restrições de processos, por exemplo) tem sido objeto de estudo dos autores.

Reconhecimento

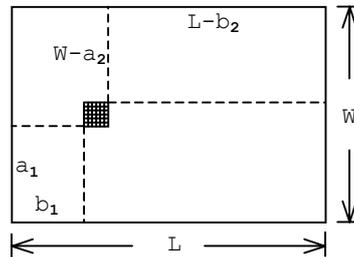
Este trabalho teve apoio da FAPESP e CNPq. Os autores agradecem aos três revisores anônimos da revista Pesquisa Operacional, cujas sugestões incluídas e melhoraram o texto final.

Referências Bibliográficas

- (1) Arenales, M. (1993). Uma Teoria para o Problema de Corte. São Carlos: USP, 1993. Tese (Livre-docência) – Instituto de Ciências Matemáticas de São Carlos, Universidade de São Paulo.
- (2) Arenales, M. & Morabito, R. (1995). An AND/OR-graph approach to the solution of two-dimensional non-guillotine cutting problems. *European Journal of Operational Research*, **84**, 599-617.
- (3) Carnieri, C.; Mendoza, G.A. & Lupold, W.G. (1993). Optimal Cutting of Dimension Parts from Lumber with defect: a Heuristic Solution Procedure. *Forest Products Journal*, 66-72.
- (4) Christofides, N. & Whitlock, C. (1977). An algorithm for two-dimensional cutting problems. *Operations Research*, **25**, 30-44.
- (5) Gilmore, P.C. & Gomory, R.E. (1965). Multi-stage cutting stock problems of two or more dimensions. *Operations Research*, **13**, 94-120.
- (6) Herz, J. (1972). Recursive Computational Procedure for Two Dimensional Stock Cutting. *IBM Journal of Research and Development*, **16**, 462-469.
- (7) Morabito, R. (1989). Corte de Estoque Bidimensional. São Carlos: USP, 1989. Dissertação (Mestrado) – Instituto de Ciências Matemáticas de São Carlos, Universidade de São Paulo.
- (8) Morabito, R. (1992). Uma Abordagem em Grafo E/OU para o Problema do Empacotamento: Aplicação ao Carregamento de Paletes e Contêineres. São Carlos: USP, 1992. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo.
- (9) Morabito, R.; Arenales, M. & Arcaro, V.F. (1992). An AND-OR-graph approach for two-dimensional cutting problems. *European Journal of Operational Research*, **58**, 263-271.
- (10) Morabito, R. & Arenales, M. (1994). An AND/OR-Graph Approach to the Container Packing Problem. *International Transactions on Operational Research*, **1**(1), 59-73.
- (11) Morabito, R. & Arenales, M. (1996). Staged and constrained two-dimensional guillotine cutting problems: An AND/OR-graph approach. *European Journal of Operational Research*, **94**, 548-560.
- (12) Vianna, A.C.G. (2000). Problema de Corte e Empacotamento: uma Abordagem em Grafo E/OU. São Carlos: USP, 2000. Tese (Doutorado) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo.
- (13) Vianna, A.C.G.; Arenales, M.N. & Morabito, R. (2000). Estrutura de Dados utilizada na Abordagem em Grafo E/OU para a Resolução de Problemas de Corte. Relatórios Técnicos nº 126, ICMC-USP.

Apêndice – Método de Carnieri *et al.* (1993)

O método apresentado por Carnieri *et al.* (1993) para a resolução do Problema de Corte com Placa Defeituosa considera uma placa de comprimento L e largura W e um conjunto de n peças retangulares de tamanhos (ℓ_i, w_i) , sendo ℓ_i o comprimento e w_i a largura da peça i . O valor de utilidade de cada peça i é dado por v_i . A placa contém um único defeito (Figura 20) e o problema consiste em alocar as peças na placa retangular de forma a maximizar o valor de utilidade total sem sobrepor o defeito de coordenadas (b_1, a_1) do canto inferior esquerdo e (b_2, a_2) do canto superior direito.

**Figura 20** – Placa defeituosa.

Inicialmente o algoritmo proposto determina a tabela $G(x,y)$, que contém os valores ótimos para retângulos de tamanho (x,y) , sendo x e y a largura e o comprimento, respectivamente. A tabela $G(x,y)$ é determinada recursivamente (programação dinâmica, Gilmore & Gomory, 1965) e, é dada por:

$$G(x, y) = \text{Max} \{ H(x, y), G(x_0, y) + G(x - x_0, y), G(x, y_0) + G(x, y - y_0) \}$$

$$0 \leq x \leq L$$

$$0 \leq y \leq W$$

e, $H(x,y) = \max \{ v_i, w_i \leq x, \ell_i \leq y \}$

Os valores x e y são tomados de um conjunto de discretização (combinações lineares inteiras não-negativas das dimensões das peças), sem perda de generalidade. Observe que, a tabela $G(x,y)$ gera valores ótimos de retângulos (x,y) sem defeitos. Esta tabela também é utilizada para gerar limitantes na busca.

O próximo passo consiste em calcular uma boa solução inicial viável para o problema com defeito, usando o melhor valor entre quatro possíveis padrões de corte (Figura 21), dados por:

$$m_1 = G(W, b_1) + G(W, L - b_2)$$

$$m_2 = G(a_1, L) + G(W - a_2, L)$$

$$m_3 = G(W, b_1) + G(a_1, L - b_1) + G(W - a_1, L - b_2)$$

$$m_4 = G(W, b_1) + G(W - a_2, L - b_1) + G(a_2, L - b_2)$$

Portanto, a solução inicial será dada por $M = \max \{ m_1, m_2, m_3, m_4 \}$.

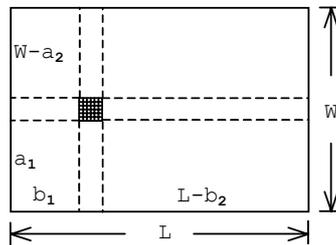


Figura 21 – Retângulos envolvidos nos quatro possíveis padrões de corte iniciais.

Na abordagem Grafo E/OU utilizamos soluções viáveis mais simples (seção 4.1).

O algoritmo de Carnieri *et al.* utiliza a solução inicial gerada como um limitante inferior e aplica um procedimento de enumeração implícita, restrita a soluções do tipo apresentada pela Figura 22, onde no máximo seis retângulos são gerados e a solução ótima para cada um desses retângulos é obtida pela fórmula recursiva de Gilmore e Gomory. O algoritmo escolhe a melhor solução entre todas as soluções geradas.

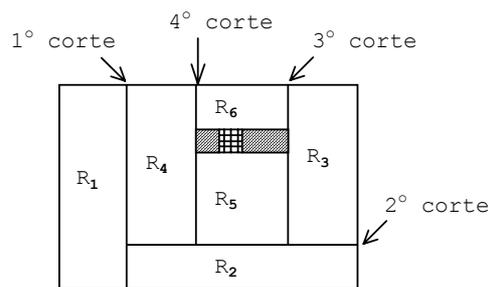


Figura 22 – Ilustração dos seis retângulos resultantes dos seis cortes.