

## Uma math-heurística para o problema de corte de peças irregulares

**Luiz Henrique Cherri**      **Franklina Maria Bragion Toledo**

Instituto de Ciências Matemáticas e de Computação

Universidade de São Paulo (USP)

{lhcherri , fran}@icmc.usp.br

**Maria Antónia Carravilla**

INESC TEC, Faculdade de Engenharia da Universidade do Porto

mac@fe.up.pt

### Resumo

O problema de corte de peças irregulares consiste em cortar peças convexas e não convexas a partir de um objeto. Apesar de diversas pesquisas serem desenvolvidas em torno deste problema, poucas têm sua base em modelos matemáticos. Neste artigo, abordamos o problema em que as peças são cortadas a partir de uma placa retangular de altura fixa e comprimento a ser minimizado. Propomos uma heurística baseada em um modelo matemático para resolver o problema. Este método é composto por fases de construção e de melhoria de solução. Os resultados mostram a eficiência da heurística, que apresenta soluções de boa qualidade em baixos tempos computacionais. Nossa proposta abre caminhos para a exploração de problemas de maior porte e para obtenção de soluções mais refinadas.

**PALAVRAS CHAVE:** *Nesting*, Math-heurísticas, Programação inteira.

**Área principal:** Programação matemática, Otimização Combinatória.

### Abstract

The nesting problem consists on cutting convex and non-convex pieces from an object. Although several studies have been developed around this problem, few have their basis on mathematical models. In this paper we approach the problem where the pieces are cut from a rectangular board with fixed height and a length to be minimized. We propose an heuristic based on a mathematical model to solve the problem. The method is composed by a construction phase followed by an improvement phase. The results show the efficiency of the heuristic, that presents good quality solutions in low computational time. Our approach opens new ways to solve larger problems and find more refined solutions.

**KEYWORDS:** *Nesting*, Math-heuristics, Integer programming.

**Main area:** Mathematical Programming, Combinatorial Optimization.

## 1 - Introdução

Os problemas de corte e empacotamento são estudados há mais de cinco décadas e aparecem com frequência no planejamento de produção de várias indústrias. Nestes problemas peças menores devem ser cortadas a partir de objetos maiores, definindo assim um plano de corte. Os principais objetivos do planejamento da produção destas indústrias são a redução dos custos e a melhoria da eficiência do processo de produção que estão associados a estratégias adequadas de corte.

O problema de corte de peças irregulares (*nesting problem*) é um caso especial dos problemas de corte e empacotamento. O problema estudado consiste em cortar peças irregulares a partir de uma placa retangular de altura fixa e comprimento ilimitado, de forma que as peças: a) estejam integralmente dentro da placa; b) não se sobreponham e c) sejam todas cortadas. O objetivo é de cunho econômico e visa minimizar o comprimento total utilizado da placa. Este problema é encontrado no planejamento de produção de diversas indústrias, tais como: confecções; metal mecânica; moveleira; calçadista; entre outras.

Boas revisões sobre o problema de corte de peças irregulares são encontradas em Bennell e Oliveira (2008) e Bennell e Oliveira (2009). Bennell e Oliveira (2008) apresentam as diferentes estruturas para a representação geométrica desse problema, ressaltando as vantagens e as desvantagens da utilização de cada tipo de representação. Entre estas estruturas estão: *raster points*, *direct trigonometry*, *nofit polygon (NFP)* e *phi function*. Bennell e Oliveira (2009) discutem diferentes abordagens heurísticas para a resolução do problema abrangendo heurísticas de construção e melhora de solução. Este problema pertence a classe de problemas NP-completos (Fowler et al., 1981) e devido à sua complexidade poucos trabalhos o abordam de forma exata. A primeira abordagem exata para a resolução do problema foi proposta em (Carravilla et al., 2003) em que é proposto um método de solução para o problema baseado em programação por restrições. Carravilla e Ribeiro (2005) e Gomes e Oliveira (2006) exploram um modelo que utiliza variáveis contínuas para definir o posicionamento das peças na placa. A avaliação do posicionamento das peças é linear, porém, a verificação de sobreposição entre elas é feita com o auxílio de variáveis binárias. Em Toledo et al. (2013) é proposto um modelo inteiro-misto no qual as variáveis de decisão são associadas a cada ponto do objeto e a cada tipo de peça. Com esta abordagem foi possível encontrar soluções ótimas para instâncias de médio porte da literatura, sujeito à precisão da discretização do objeto. Porém, os resultados obtidos pelos autores mostraram que o tempo computacional necessário para a resolução do problema se eleva significativamente com o crescimento do número de peças e da quantidade de tipos de peças, podendo atingir mais de cinco horas para instâncias com menos de 35 peças.

Devido à dificuldade de solução, vários autores desenvolveram heurísticas e meta-heurísticas para a resolução do problema, no entanto, a maioria delas não é baseada em modelagem matemática. Podemos classificar os métodos heurísticos em dois grupos: heurísticas construtivas e heurísticas de melhoria. As heurísticas construtivas têm por objetivo criar um plano de corte, alocando na placa uma peça por vez, obtendo ao final do processo uma solução para o problema. Diversos trabalhos da literatura abordam heurísticas com esta característica, dentre eles destacamos: Albano e Sapuppo (1980); Segenreich e Braga (1986); Dowsland et al. (2002); Burke et al. (2006); Bennell e Song (2008). As heurísticas de melhoria têm o objetivo de melhorar soluções já existentes, nesta linha destacamos: Oliveira e Ferreira (1993); Jakobs (1996); Dowsland et al. (1998); Oliveira et al. (2000); Babu e Babu (2001); Bennell e Dowsland (2001); Gomes e Oliveira (2002); Takahara et al. (2003); Burke et al. (2006).

Neste artigo, desenvolvemos um método heurístico baseado no modelo matemático apresentado por Toledo et al. (2013) para a resolução do problema de corte de peças

irregulares. Nosso objetivo é obter soluções de boa qualidade para o problema em tempo computacional aceitável. A heurística proposta possui uma fase de construção, na qual é construída uma solução inicial para o problema, e uma fase de melhoria de solução, em que a solução obtida na primeira fase é aprimorada. Testes computacionais foram realizados e os resultados obtidos foram comparados com os apresentados por Toledo et al. (2013).

Este artigo está organizado da seguinte maneira. Na Seção 2, é apresentada a definição do problema abordado e o modelo proposto por Toledo et al. (2013). Na Seção 3, é descrita a heurística proposta. Na Seção 4, os resultados obtidos são reportados. Na Seção 5, são apresentadas as considerações finais do trabalho.

## 2 - Definição do problema

O problema de corte de peças irregulares consiste em cortar peças, convexas e não convexas, a partir de uma placa de altura fixa e comprimento infinito. O objetivo é minimizar o comprimento total utilizado da placa. Um modelo matemático inteiro misto para a resolução deste problema foi proposto por Toledo et al. (2013). Neste modelo a placa é representada por uma malha de pontos. A distância horizontal e vertical entre os pontos desta malha é denominada  $g_x$  e  $g_y$ , respectivamente. Estas distâncias definem o refinamento da malha utilizada, que deve ser escolhida com cuidado, pois influencia diretamente na qualidade da solução e no número de variáveis do problema. Na modelagem, uma solução é factível para o problema se: a) as peças não se sobrepõem; b) as peças são alocadas inteiramente dentro da placa; e c) todas as peças são cortadas.

Cada peça é definida por um conjunto de vértices e um vértice deste conjunto é selecionado para ser seu ponto de referência. A Figura 1 ilustra um exemplo com três peças com seus pontos de referência em destaque.

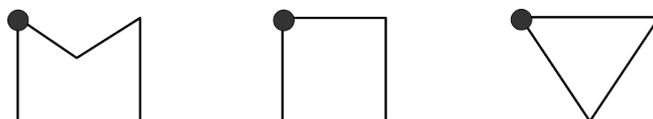


Figura 1: Exemplo de três peças com seus respectivos pontos de referência destacados.

A intersecção entre as peças utilizada é representada pelo *nofit polygon* entre elas (Art, 1966). O *nofit polygon* é obtido para cada possível par de peças, e sua forma depende do ponto de referência escolhido para cada peça (ver Figura 2). A partir do *nofit polygon*, verificar se há intersecção entre duas peças se resume a verificar se o ponto de referência está dentro, fora ou na fronteira do *nofit polygon*. Mais especificamente, considere o conjunto  $NFP_{t,u}$  que representa os pontos do *nofit polygon* entre as peças  $t$  e  $u$ . A sobreposição entre elas ocorre se o ponto de referência da peça  $u$  pertence ao interior do  $NFP_{t,u}$ . As peças estão em contato se o ponto de referência da peça  $u$  está na fronteira do  $NFP_{t,u}$ ; e as peças estão separadas se o ponto de referência da peça  $u$  não pertence ao interior e à fronteira do  $NFP_{t,u}$ .

O modelo proposto pelos autores é dado por:

$$\text{minimize } z \tag{1}$$

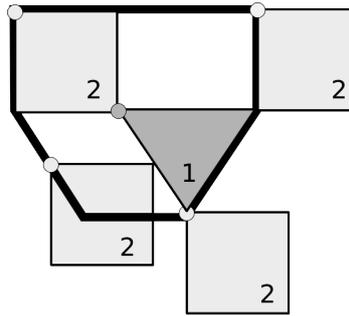


Figura 2: *Nofit polygon* da peça 2 em relação a peça 1 ( $NFP_{1,2}$ ).

Sujeito a:

$$(c \times g_x + x_t^M) \times \delta_t^{r,c} \leq z \quad \forall (r,c) \in \mathcal{IFP}_t, \forall t \in \mathcal{T}; \quad (2)$$

$$\sum_{(r,c) \in \mathcal{IFP}_t} \delta_t^{r,c} = q_t \quad \forall t \in \mathcal{T}; \quad (3)$$

$$\delta_u^{k,l} + \delta_t^{r,c} \leq 1 \quad \forall (k,l) \in \mathcal{NFP}_{t,u}^{r,c}, \forall t, u \in \mathcal{T}, \forall (r,c) \in \mathcal{IFP}_t; \quad (4)$$

$$\delta_t^{r,c} \in \{0, 1\} \quad \forall (r,c) \in \mathcal{IFP}_t, \forall t \in \mathcal{T}; \quad (5)$$

$$z \geq 0. \quad (6)$$

Índices:

- $\mathcal{T} = \{1, \dots, T\}$ : Conjunto de tipos de peças;
- $\mathcal{C} = \{0, \dots, C - 1\}$ : Conjunto das colunas da malha de pontos;
- $\mathcal{R} = \{0, \dots, R - 1\}$ : Conjunto das linhas da malha de pontos;
- $t, u \in \mathcal{T}$ : Tipos de peça;
- $c \in \mathcal{C}$ : Coluna da malha de pontos;
- $r \in \mathcal{R}$ : Linha da malha de pontos.

Parâmetros:

- $q_t$ : Número de peças do tipo  $t$  a serem cortadas;
- $x_t^M$ : Distância entre o ponto de referência e o vértice mais a direita da peça do tipo  $t$ ;
- $\mathcal{IFP}_t$ : Conjunto de pontos pertencentes à placa ( $\mathcal{R} \times \mathcal{C}$ ) no qual o ponto de referência da peça  $t$  pode ser alocado sem a peça ultrapassar os limites da placa;
- $\mathcal{NFP}_{t,u}^{r,c}$ : Conjunto de pontos que estão no interior do  $NFP_{t,u}$  se a peça do tipo  $t$  for posicionada no ponto  $(r, c)$ . Portanto, se  $t$  for posicionada em  $(r, c)$  e se  $u$  for posicionado num ponto de  $\mathcal{NFP}_{t,u}^{r,c}$  então as peças dos tipos  $t$  e  $u$  intersectam-se.

Variáveis:

- $\delta_t^{r,c}$ : Variável binária de decisão que recebe o valor 1 se a peça do tipo  $t$  está alocada no ponto  $(r, c)$  da placa, e 0 caso contrário.

No problema (1)-(6), a função objetivo (1) em conjunto com as restrições (2) visam minimizar o comprimento total utilizado da placa. As restrições (3) garantem que todas as peças serão alocadas na placa. A não sobreposição das peças é imposta pelas restrições (4). Em (5) é definido o domínio das variáveis de decisão para assegurar que as peças são alocadas sem extrapolar os limites da placa. O domínio da variável  $z$  é definido em (6).

### 3 - Math-heurística proposta

Nesta seção é proposta uma math-heurística para a resolução do problema de corte de peças irregulares baseada no modelo de Toledo et al. (2013). Por ser um problema de difícil resolução, nosso objetivo é propor uma math-heurística que obtenha soluções de boa qualidade para o problema em tempo computacional aceitável.

A math-heurística proposta pode ser dividida em duas fases: 1) construção de uma solução e 2) melhoria desta solução. Com inspiração no método *relax-and-fix*, em cada um dos passos da primeira fase, as peças são divididas em quatro coleções: a) coleção de peças fixas, ou seja, peças que estão fixadas na placa; b) coleção de peças posicionadas, ou seja, peças que foram alocadas na placa, mas que ainda podemos movimentar parte delas; c) coleção de peças livres, ou seja, peças que podem ser livremente alocadas em qualquer posição da placa desde que não gere intersecção entre as peças; e d) coleção de peças em espera, ou seja, que não serão consideradas no passo atual de resolução do problema.

No primeiro passo da fase inicial, as coleções de peças fixas e posicionadas estão vazias, um pequeno número de peças é designado à coleção de peças livres e todas as peças restantes são designadas à coleção de peças em espera. Em cada passo, um subproblema definido para um número limitado de peças é resolvido. Para definir o subproblema de cada passo são necessários os valores dos limitantes superiores dos dois últimos subproblemas resolvidos,  $\mu'$  e  $\mu$ . Inicialmente esses parâmetros são zero. Ao final de cada passo as coleções são redefinidas, as peças que têm o ponto de referência posicionado no intervalo  $[0, \mu')$  definem a coleção de peças fixas, as peças com o ponto de referência posicionado no intervalo  $[\mu', \mu]$  definem a coleção de peças posicionadas e um pequeno número de peças da coleção de peças em espera é movido para a coleção de peças livres. No último passo desta fase inicial, a coleção de peças em espera é vazia. Ao resolver o subproblema a ele associado temos uma solução factível para o problema. Esta primeira fase é ilustrada na Figura 3 em que, as peças em cinza correspondem à coleção de peças alocadas, porém não fixadas. As peças em preto, pertencem à coleção de peças fixadas. O valor abaixo da placa indica o comprimento utilizado até o passo em questão. Destacamos que como no Passo 1 estas duas coleções são vazias, não há peças alocadas na placa. As peças em branco fora da placa ainda não foram alocadas e o número abaixo delas indica o número de vezes que ainda devem ser inseridas. A cada passo são adicionadas oito peças da coleção de peças em espera à coleção de peças livres. Para o exemplo, em cinco passos, foi obtida uma solução factível para o problema.

Para obtermos a modelagem do subproblema que será utilizado nos passos da primeira fase considere:

- as constantes  $\bar{\delta}_t^{r,c}$ , que têm o valor 1 se a peça  $t$  foi alocada no ponto  $(r, c)$  da placa no passo anterior e 0 caso contrário;
- o intervalo  $\mathcal{M} = [0, \mu')$ , que define o intervalo das colunas em que as peças da coleção de peças fixas estão na placa;
- o intervalo  $\mathcal{W} = [\mu', \mu]$ , que define o intervalo das colunas em que as peças da coleção de peças alocadas estão na placa;

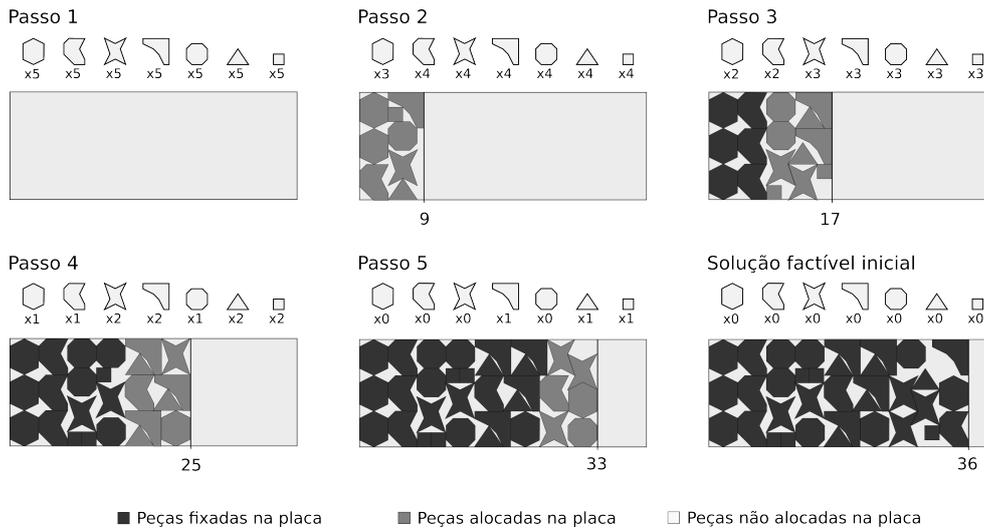


Figura 3: Primeira fase: processo de construção de uma solução inicial.

- o parâmetro  $\alpha_t$  é igual a  $\sum_{r \in \mathcal{R}, c \in \mathcal{W}} \bar{\delta}_t^{r,c}$ , ou seja, é igual ao número de peças do tipo  $t$  com o ponto de referência contido no conjunto  $\mathcal{R} \times \mathcal{W}$ ;
- $\tilde{q}_t$  a quantidade de peças do tipo  $t$  presentes nas coleções de peças fixas, posicionadas e livres.

Então, o subproblema pode ser modelado como descrito em (7) - (15).

$$\text{SP} = \text{minimize} \quad z \quad (7)$$

Sujeito a:

$$(c \times g_x + x_t^M) \times \delta_t^{r,c} \leq z \quad \forall (r, c) \in \mathcal{IFP}_t, \forall t \in \mathcal{T}; \quad (8)$$

$$\sum_{(r,c) \in \mathcal{IFP}_t} \delta_t^{r,c} = \tilde{q}_t \quad \forall t \in \mathcal{T}; \quad (9)$$

$$\delta_u^{k,l} + \delta_t^{r,c} \leq 1 \quad \forall (k, l) \in \mathcal{NFP}_{t,u}^{r,c}, \forall t, u \in \mathcal{T}, \forall (r, c) \in \mathcal{IFP}_t; \quad (10)$$

$$\sum_{\substack{\bar{\delta}_t^{r,c} = 1 \\ c \in \mathcal{W} \\ r \in \mathcal{R}}} (1 - \delta_t^{r,c}) + \sum_{\substack{\bar{\delta}_t^{r,c} = 0 \\ c \in \mathcal{W} \\ r \in \mathcal{R}}} \delta_t^{r,c} \leq \alpha_t, \quad \forall t \in \mathcal{T}; \quad (11)$$

$$\delta_t^{r,c} = 1 \quad \forall (r, c) \in \mathcal{IFP}_t \cap (\mathcal{R} \times \mathcal{M}), \quad (12)$$

$$\forall t \in \mathcal{T}, \forall \bar{\delta}_t^{r,c} = 1; \quad (13)$$

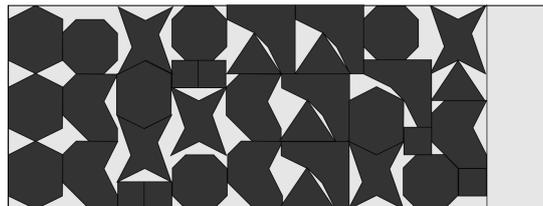
$$\delta_t^{r,c} \in \{0, 1\} \quad \forall (r, c) \in \mathcal{IFP}_t, \forall t \in \mathcal{T}; \quad (14)$$

$$z \geq 0. \quad (15)$$

No modelo (7 - 15), a função objetivo visa minimizar o comprimento total utilizado da placa. As restrições (8), (9), (10), (14) e (15) são similares às equações (2), (3), (4), (5) e (6) do modelo (1 - 6). As restrições (11) limitam as possibilidades de alterações das variáveis pertencentes ao intervalo  $\mathcal{W}$  da placa. Mais especificamente, uma alteração é contabilizada quando uma peça antes alocada no intervalo  $\mathcal{W}$  é reposicionada fora deste intervalo, ou quando uma peça da coleção de peças livres é posicionada no intervalo  $\mathcal{W}$ . Quando uma peça que está posicionada no intervalo  $\mathcal{W}$  muda de posição dentro do intervalo, duas alterações

são contabilizadas. No máximo,  $\alpha_t$  alterações são permitidas. As restrições (13) mantêm fixas as peças que foram anteriormente alocadas no intervalo  $\mathcal{M}$ .

Na segunda fase, a partir da solução encontrada, definimos o intervalo  $\mathcal{W}$  como  $[0, \mu]$ , ou seja, todas as peças são pertencentes à coleção de peças alocadas deixando as demais coleções vazias. A resolução do modelo (7-15) com estes parâmetros define a segunda fase da busca. O conjunto de restrições deste problema define uma vizinhança da solução encontrada na primeira fase. Ao resolvermos este problema, obtemos a melhor solução nesta vizinhança. A Figura 4 ilustra a solução obtida na segunda fase para o exemplo ilustrado na Figura 3.



35

Figura 4: Solução obtida ao final da segunda fase.

O algoritmo 1 resume os passos da heurística proposta.

---

**Algoritmo 1** Heurística proposta

---

0 Inicialização:

Considere as coleções:

$\Gamma$ : de peças fixas;

$\Delta$ : de peças posicionadas;

$\Theta$ : de peças livres;

$\Omega$ : de peças em espera;

Seja:

$\beta$  o número de peças acrescentadas à coleção  $\Theta$  a cada passo;

$\bar{\delta} = \{\bar{\delta}_t^{r,c}, \forall r \in \mathcal{R}, c \in \mathcal{C} \text{ e } t \in \mathcal{T}\}$ ;

Faça:

$\bar{\delta} = \mathbf{0}$  e  $\mu' = \mu = 0$ ;

$\Gamma = \emptyset$ ;  $\Delta = \emptyset$  e  $\Theta = \emptyset$ ;

$\Omega$  Possui todas as peças do problema;

1 Primeira fase:

Enquanto todas as peças não forem alocadas:

Defina  $\mathcal{M} = [0, \mu']$  e  $\mathcal{W} = [\mu', \mu]$ ;

$\Gamma = \{\text{peças contidas no intervalo } \mathcal{M}\}$ ;

$\Delta = \{\text{peças contidas no intervalo } \mathcal{W}\}$ ;

$\Theta = \emptyset$ ;

Remova  $\min\{\beta, |\Omega|\}$  peças de  $\Omega$  e inclua em  $\Theta$ ;

Resolva o subproblema (7 - 15) obtendo a solução  $\bar{\delta}$  com valor  $\bar{z}$ ;

Faça  $\mu' = \mu$  e  $\mu = z$ ;

2 Segunda fase:

Faça,  $\mathcal{W} = [0, \mu]$  e  $\Delta = \{\text{todas as peças do problema}\}$ ;

Resolva o subproblema (7 - 15);

3 Retorne a solução obtida.

---

As peças da coleção de peças em espera ( $\Omega$ ) adicionadas à coleção de peças livres

( $\Theta$ ) a cada etapa são selecionadas sequencialmente em relação aos índices dos tipos. Partindo de um índice  $k$ , se  $\tilde{q}_t < q_t$  então uma unidade desta peça é selecionada para fazer parte de  $\Theta$  e  $k$  é atualizado para o índice da próxima peça. O processo é executado até que  $|\Theta| = \min\{\beta, |\Omega|\}$ . Esta seleção maximiza a diversidade de peças livres na resolução dos subproblemas.

## 4 - Resultados

Nesta seção, apresentamos os resultados computacionais obtidos a partir da math-heurística apresentada na Seção 3. Para a resolução dos subproblemas gerados foi utilizado o software de otimização CPLEX 12.5.

A math-heurística foi codificada em linguagem C/C++ e os testes foram conduzidos num computador com processador Intel Core i7-2600 3.4 GHz com 16 Gb de memória e sistema operacional Ubuntu 12.04 (64 bits). O modelo de Toledo et al. (2013) foi implementado nesta mesma plataforma para realizarmos uma avaliação justa entre o método heurístico e o método exato.

O conjunto de peças BLAZEWICZ (Oliveira et al., 2000) foi utilizado para a avaliação da heurística. Este conjunto é composto por sete tipos de peças distintas que são ilustradas na Figura 5.

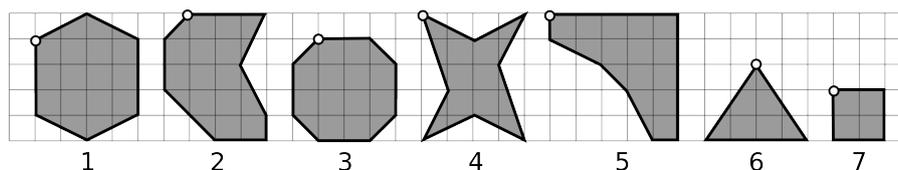


Figura 5: Tipos de peças presentes nas instâncias BLAZEWICZ respectivamente numeradas. Os comprimentos e alturas das peças da instância estão compreendidos no intervalo  $[2,5]$ . O ponto de referência de cada peça é destacado por um círculo branco.

A construção das instâncias utilizando este conjunto de peças procedeu como em Toledo et al. (2013), a fim de tornar possível a comparação entre os resultados obtidos pelo método exato e pela math-heurística. O conjunto de testes  $Bn$  consiste do conjunto de instâncias formadas por todos os sete tipos de peças do conjunto BLAZEWICZ e  $n$  repetições de cada tipo de peça. O conjunto de testes  $BP2P4\_m\_n$  compreende apenas peças do tipo 2 e 4 do conjunto BLAZEWICZ, sendo  $m$  peças do tipo 2 e  $n$  peças do tipo 4. A altura da placa utilizada para ambos os conjuntos foi de 15 unidades. A distância horizontal ( $g_x$ ) e vertical ( $g_y$ ) entre os pontos da malha utilizada foi de 1 unidade.

Na Tabela 1 apresentamos os resultados computacionais obtidos para as instâncias teste avaliadas. Na tabela, a primeira coluna apresenta o nome da instância, a segunda coluna apresenta o número total de peças presentes na instância. Os limitantes superiores para as instâncias obtidos de Toledo et al. (2013) são apresentados na terceira coluna. Nas colunas 4, 5 e 6 são apresentados os resultados obtidos pelo modelo matemático de Toledo et al. (2013), respectivamente o valor da solução, o tempo computacional para encontrar a solução e o tempo decorrido na prova de otimalidade. Nas colunas 8 e 9 são apresentados o valor das soluções e os respectivos tempos computacionais utilizando somente a primeira fase da heurística. Por fim, nas colunas 10 e 11 são apresentados o valor das soluções e tempos computacionais para resolução do problema com heurística completa. Todos os tempos apresentados estão na escala de segundos.

Analisando os resultados obtidos ao final da primeira e da segunda fase da heurística, podemos notar que houve um ganho na qualidade da solução quando realizamos a

Tabela 1: Resultados computacionais obtidos a partir das instâncias teste

Instância	Nº total de peças	Lim. superior	Toledo et al. (2013)			Heur. (início)		Heurística	
			Sol.	T. (s)	T. (opt.)	Sol.	T.	Sol.	T.
B1	7	8	8	1	1	8	1	8	1
B2	14	15	14	16	19	15	10	14	12
B3	21	33	20	1885	2002	22	30	21	36
B4	28	38	28	336	TL	28	71	28	116
B5	35	37	35	1478	TL	36	217	35	273
BP2P4_4_3	7	60	11	1	1	11	0	11	0
BP2P4_7_7	14	60	19	2	1	19	0	19	0
BP2P4_11_10	21	60	28	2	3	28	2	28	2
BP2P4_14_14	28	60	38	3	50	38	5	38	6
BP2P4_18_17	35	60	47	3	208	47	5	47	8
BP2P4_21_21	42	60	56	3	364	56	6	56	13
BP2P4_28_28	56	100	74	3	TL	75	12	74	39
BP2P4_35_35	70	100	93	3	TL	95	13	94	110

segunda fase. Para as instâncias com dois tipos de peças 2 soluções foram melhoradas e para as instâncias com sete tipos de peças 3 soluções foram melhoradas. Em relação ao tempo computacional, para as instâncias com dois tipos de peças, os tempos obtidos ao final da primeira e segunda fase da math-heurística podem ter diferenças muito significativas, chegando a mais de 8 vezes na instância BP2P4\_35\_35. Para as instâncias com sete tipos de peças, a diferença entre os tempos de execução entre o final da primeira e da segunda fase da heurística são menos significativos.

Quando comparamos os resultados obtidos em Toledo et al. (2013) aos encontrados pela math-heurística proposta ao final da segunda fase podemos notar que, na maioria dos casos, os valores das soluções obtidas pelos métodos são os mesmos, e apenas para B3 e BP2P4\_35\_35 a math-heurística obtém soluções com uma unidade de comprimento a mais.

Em relação ao tempo computacional na obtenção de soluções, verificamos que a heurística apresenta valores menores que os apresentados pelo método exato quando este prova a otimalidade da solução, principalmente quando focamos em problemas com um maior número de peças em que as diferenças entre os tempos de execução são de grande magnitude.

Comparando o tempo computacional do modelo encontrar a solução e para a math-heurística obter uma solução ao final da segunda fase, notamos que para dois tipos de peças o modelo obtém a solução ótima em poucos segundos, o que não justifica o uso da heurística nesses casos. Para as instâncias com sete peças, a heurística obtém a solução mais rápido que o método exato, indicando que a heurística pode ser útil na resolução de problemas com esta característica. Vale ressaltar que o método proposto, conseguiu encontrar uma solução de boa qualidade para todas as instâncias apresentadas em menos de cinco minutos.

Levando em consideração o número de tipos distintos de peças, podemos ver que existe uma correlação entre este valor e o tempo computacional. Isso ocorre pelo fato de que um número maior de tipos de peças resulta em subproblemas maiores, ou seja, com mais variáveis e restrições.

Dentre as instâncias avaliadas, desconsiderando o trabalho de Toledo et al. (2013), encontramos na literatura resultados apenas para a instância B4. O melhor resultado reportado considerando rotações de 0° e 180° foi de 25,64 (Imamichi et al., 2009), ou seja, 8,4% melhor que o resultado obtido pelo método proposto não considerando rotações.

## 5 - Conclusões

Neste artigo, foi abordado o problema de corte de peças irregulares com base na modelagem matemática proposta por Toledo et al. (2013). Dado uma instância para o problema, o objetivo foi encontrar um plano de corte que atenda toda a demanda pelas peças e minimize o comprimento total utilizado da placa.

Uma math-heurística para a resolução do problema foi proposta. O método é composto por duas fases: 1) construção de uma solução; e 2) melhoria da solução. Com ela foi possível obter soluções de boa qualidade para o problema em menos de cinco minutos, o que abre os horizontes para a exploração de problemas de maior porte e para a utilização de uma malha de cálculo mais precisa para a resolução do problema.

## Agradecimentos

Os autores agradecem as valiosas sugestões dos revisores. Este trabalho contou com o apoio financeiro da FAPESP(2012/18653-8 e 2010/10133-0), CNPq (300713/2010-0) e FP7-PEOPLE-2009-IRSES (246881) da European Commission.

## Referências

- Albano, A.; Sapuppo, G.** (1980). Optimal allocation of two-dimensional irregular shapes using heuristic search methods. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-10, 242–248.
- Art, R. C.** (1966). An approach to the two dimensional irregular cutting stock problem. Technical Report 36. Y08, IBM Cambridge Scientific Center, Cambridge, Massachusetts, USA.
- Babu, A. R.; Babu, N. R.** (2001). A generic approach for nesting of 2-d parts in 2-d sheets using genetic and heuristic algorithms. *Computer-Aided Design*, 33, 879–891.
- Bennell, J. A.; Dowsland, K. A.** (2001). Hybridising tabu search with optimisation techniques for irregular stock cutting. *Management Science*, 47, 1160–1172.
- Bennell, J. A.; Oliveira, J. F.** (2008). The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, 184, 397 – 415.
- Bennell, J. A.; Oliveira, J. F.** (2009). A tutorial in irregular shape packing problems. *Journal of the Operational Research Society*, 60, S93–S105.
- Bennell, J. A.; Song, X.** (2008). A comprehensive and robust procedure for obtaining the no-fit-polygon using minkowski sums. *Computers and Operations Research*, 35, 267–281.
- Burke, E.; Hellier, R.; Kendall, G.; Whitwell, G.** (2006). A new bottom-left-fill heuristic algorithm for the two-dimensional irregular packing problem. *Operations Research*, 54, 587–601.
- Carravilla, M. A.; Ribeiro, C.** (2005). CP and MIP in the resolution of hard combinatorial problems: a case study with nesting problems. *In CSCLP 2005 Joint ER-CIM/CoLogNET International Workshop on Constraint Solving and Constraint Logic Programming*, 1, 113–127.
- Carravilla, M. A.; Ribeiro, C.; Oliveira, J. F.** (2003). Solving nesting problems with non-convex polygons by constraint logic programming. *International Transactions in Operational Research*, 10, 651–663.
- Dowsland, K. A.; Dowsland, W. B.; Bennell, J. A.** (1998). Jostle for position: Local improvement for irregular cutting patterns. *The Journal of the Operational Research Society*, 49, 647–658.

- Dowsland, K. A.; Vaid, S.; Dowsland, W. B.** (2002). An algorithm for polygon placement using a bottom-left strategy. *European Journal of Operational Research*, 141, 371–381.
- Fowler, R. J.; Paterson, M.; Tanimoto, S. L.** (1981). Optimal packing and covering in the plane are np-complete. *Information Processing Letters*, 12, 133–137.
- Gomes, A. M.; Oliveira, J. F.** (2002). A 2-exchange heuristic for nesting problems. *European Journal of Operational Research*, 141, 359 – 370.
- Gomes, A. M.; Oliveira, J. F.** (2006). Solving irregular strip packing problems by hybridising simulated annealing and linear programming. *European Journal of Operational Research*, 171, 811–829.
- Imamichi, T.; Yagiura, M.; Nagamochi, H.** (2009). An iterated local search algorithm based on nonlinear programming for the irregular strip packing problem. *Discrete Optimization*, 6, 345–361.
- Jakobs, S.** (1996). On genetic algorithms for the packing of polygons. *European Journal of Operational Research*, 88, 165–181.
- Oliveira, J. F.; Ferreira, J. S.** (1993). Algorithms for nesting problems, applied simulated annealing. In Vidal, R.V.V. (Ed., *Lecture Notes in Economics and Maths Systems*), 396, 255–274.
- Oliveira, J. F.; Gomes, A. M.; Ferreira, J. S.** (2000). Topos - a new constructive algorithm for nesting problems. *OR Spektrum*, 22, 263–284.
- Segenreich, S.; Braga, L.** (1986). Optimal nesting of general plane figures - a monte-carlo heuristic approach. *Computers & Graphics*, 10, 229–237.
- Takahara, S.; Kusumoto, Y.; S., M.** (2003). Solution for textile nesting problems using adaptive meta-heuristics and grouping. *Soft Computing*, 7, 154 – 159.
- Toledo, F. M. B.; Carravilla, M. A.; Ribeiro, C.; Oliveira, J. F.; Gomes, A. M.** (2013). The Dotted-Board Model: a new MIP model for nesting irregular shapes (on-line). *International Journal of Production Economics*, pages 1–20. DOI: 10.1016/j.ijpe.2013.04.009.