

O problema de posicionamento de peças irregulares: uma abordagem por programação por restrições

Luiz Henrique Cherri

Instituto de Ciências Matemáticas e de Computação
Avenida Trabalhador São-carlense, 13566-590, São Carlos - SP, Brasil
INESC TEC, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465, Porto, Portugal
lhcherri@icmc.usp.br

Maria Antónia Carravilla

INESC TEC, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465, Porto, Portugal
mac@fe.up.pt

Cristina Ribeiro

INESC TEC, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465, Porto, Portugal
mcr@fe.up.pt

Franklina Maria Bragion de Toledo

Instituto de Ciências Matemáticas e de Computação
Avenida Trabalhador São-carlense, 13566-590, São Carlos - SP, Brasil
fran@icmc.usp.br

RESUMO

Neste trabalho, apresentamos um método de solução exato usando programação por restrições para o problema de posicionamento de peças irregulares. Problemas de corte e empacotamento de peças irregulares são estudados há mais de cinco décadas e possuem diversas variantes. Uma das principais variantes é o problema de posicionamento de peças irregulares no qual o maior número possível de peças deve ser cortado a partir de uma placa de dimensões fixas. Apesar do grande número de trabalhos abordando o problema de posicionamento de peças irregulares, nenhum método exato foi proposto até ao momento na literatura para resolver este problema. Na abordagem por programação por restrições apresentada neste trabalho, a otimalidade é condicionada à discretização utilizada. Os resultados computacionais comprovam a eficiência do método proposto na busca por soluções do problema sob diferentes discretizações. A formulação proposta é bastante flexível, podendo ser facilmente adaptada a outras variações dos problemas de corte e empacotamento de peças irregulares.

PALAVRAS CHAVE. Nesting, Programação por restrições, Métodos exatos.

Área Principal: Otimização Combinatória, Programação Matemática.

ABSTRACT

In this work we propose an exact method to solve the irregular placement problem, based on constraint programming. Irregular cutting and packing problems are studied for more than five decades and have many variants. The irregular placement problem is a common variant of this

problem where the maximum possible number of pieces must to be cut from a board with fixed dimensions. Despite the large number of papers dealing with the irregular placement problems, there is no exact method proposed on the literature to solve this problem. In the constraint programming approach proposed, the optimality of the solution is subject to the used discretization. The computational experiments prove the efficiency of the proposed method in the search for solutions over different discretizations. Because of the flexibility of the proposed formulation, this approach can be easily adapted to other variations of the irregular cutting and packing problems.

KEYWORDS. Nesting, Constraint programming, Exact methods.

Main Area: Combinatorial Optimization, Mathematical Programming.



1. Introdução

Problemas de corte e empacotamento de peças irregulares são estudados há mais de cinco décadas. As condições de factibilidade de uma solução do problema são a não-sobreposição das peças e a garantia que elas estejam inteiramente contidas na placa. O problema de posicionamento de peças irregulares (*PPPI*) é uma variação destes problemas na qual a placa em que as peças são cortadas é dada por um polígono convexo ou não-convexo de dimensões finitas. O objetivo é selecionar as peças que serão cortadas e seu posicionamento de forma a melhorar a qualidade da solução segundo algum objetivo, como por exemplo, minimizar desperdícios ou maximizar o lucro.

Apesar do grande número de trabalhos envolvendo o corte ou empacotamento de peças irregulares, poucos abordam o *PPPI*. Daniels e Milenkovic (1997) propõem estratégias para a resolução do problema de empacotamento de peças convexas em placas convexas ou não-convexas. Valle et al. (2012) apresentam uma heurística para a resolução do problema na qual as peças são agrupadas em clusters retangulares, os quais são cortados a partir de recipientes maiores via corte regular guilhotinado. Uma heurística para o empacotamento de peças irregulares em recipientes também irregulares foi proposta por Dalalah et al. (2014). Em relação a métodos exatos para a resolução do problema, Milenkovic (1997) propõe a resolução do problema no qual as peças e a placa podem ser convexas ou não-convexas. A performance do método está diretamente conectada à quantidade de peças que se deseja empacotar e à complexidade dos polígonos que representam estas peças e a placa.

Um método exato que vem sendo aplicado com sucesso na resolução de problemas de otimização combinatória é programação por restrições. Para resolver o problema de empacotamento ortogonal bidimensional, Clautiaux et al. (2008) propõem uma abordagem via programação por restrições com a qual foi possível obter melhores resultados que os encontrados previamente na literatura. Salas et al. (2014) propõem restrições de não-sobreposição entre peças descritas por funções não lineares baseada em somas de Minkowski. A velocidade de convergência do método proposto pelos autores é sensível ao formato das peças. Uma restrição global para a resolução do problema de particionamento de conjuntos foi proposta por Saldanha e Morgado (2003). A nova restrição tem uma propagação eficiente e segundo os autores poderia ser facilmente adaptada a outros problemas similares. Carravilla et al. (2003) utilizam programação por restrições para a resolução do problema de corte de peças irregulares no qual os objetos devem ser empacotados em uma faixa de altura fixa e comprimento a ser minimizado, sendo o primeiro método exato para a resolução deste problema. Também para a resolução deste problema, Ribeiro e Carravilla (2008) propõem a restrição global *outside* a qual, se satisfeita, garante a factibilidade da solução encontrada.

Neste artigo, é proposta uma estratégia que utiliza programação por restrições para a resolução do *PPPI*. Uma nova abordagem para a definição das variáveis e seus domínios é proposta, bem como restrições que garantem as condições de factibilidade da solução obtida. O método proposto encontra soluções ótimas para o problema, sujeito a ao nível de discretização utilizado. Nossa abordagem difere da apresentada em Carravilla et al. (2003) e Ribeiro e Carravilla (2008) desde a definição de variáveis, o que permite a resolução do problema em recipientes de tamanho finito. Testes computacionais realizados permitiram verificar a eficácia do método na resolução de instâncias com diferentes características.

Após esta breve seção introdutória, na Seção 2 definimos o problema abordado e as estruturas geométricas que são utilizadas ao longo do texto. Na Seção 3 propomos uma nova definição de variáveis, seus domínios e as restrições do problema. Os resultados computacionais são discutidos na Seção 4. Por fim, na Seção 5 apresentamos as considerações finais do trabalho.

2. Definição do problema e estruturas geométricas

O problema de posicionamento de peças irregulares objetiva empacotar T tipos de peças em um recipiente de dimensões fixas. A quantidade de cópias de cada tipo de peça pode ser ilimitada. No entanto, também é possível considerar um limite máximo de produção para cada tipo

de peça ($q_t, t = 1, \dots, T$). Neste trabalho, as peças são representadas por polígonos convexos e/ou não-convexos que podem conter buracos. Um número fixo de rotações R_t é permitido para cada peça do tipo t . A placa é definida por um retângulo de altura H e comprimento L . As peças podem ser posicionadas na placa em um conjunto finito de pontos (D) que é definido a priori. Logo, a qualidade da solução encontrada é condicionada à qualidade da discretização utilizada.

Uma peça é representada por um conjunto ordenado de vértices, sendo um deles escolhido para ser o ponto de referência da peça. O posicionamento de uma peça na placa é realizada posicionando seu ponto de referência em um dos pontos do conjunto D . A Figura 1 ilustra uma peça do tipo t na rotação r com os vértices destacados pelos círculos brancos e o ponto de referência destacado pelo círculo preto. Na figura, também são destacados parâmetros importantes para a compreensão do restante do artigo, são eles: l_{tr}^{\min} (l_{tr}^{\max}): distância horizontal entre o ponto de referência e o vértice mais à esquerda (direita) da peça; h_{tr}^{\min} (h_{tr}^{\max}): distância vertical entre o ponto de referência e o vértice mais acima (abaixo) da peça.

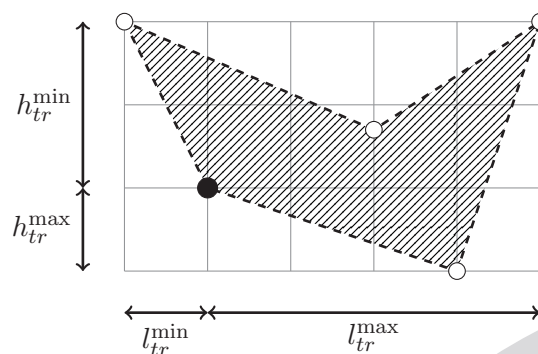


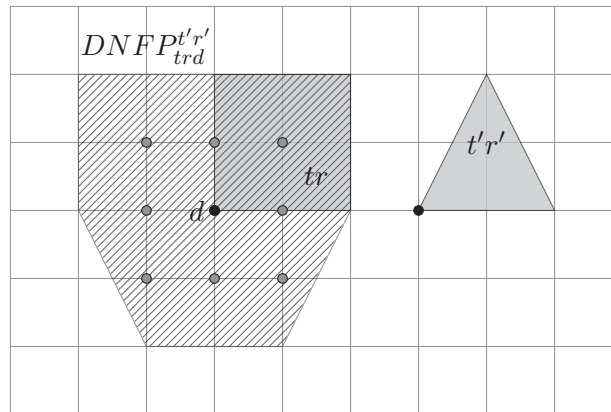
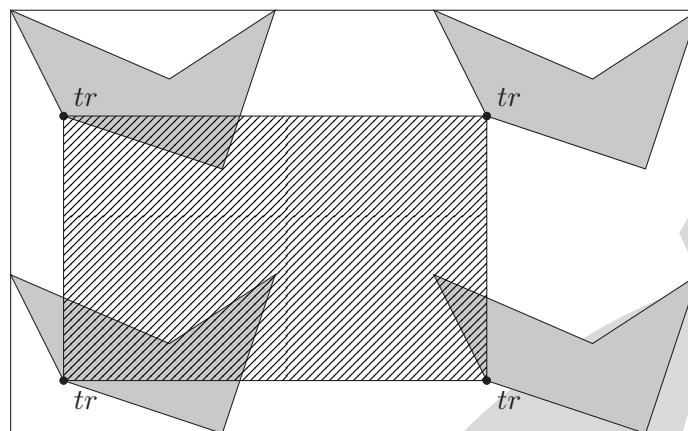
Figura 1: Exemplo de peça. Demarcando os vértices, ponto de referência e algumas medidas importantes.

Para identificar se há sobreposição entre as peças, algumas estratégias foram propostas na literatura tais como, *raster points*, *D-functions* e *nofit-polygons*. Uma revisão mais detalhada sobre estratégias para a avaliação de sobreposição entre peças pode ser encontrada em Bennell e Oliveira (2008). Nesta abordagem utilizamos *nofit-polygons* para garantir a não-sobreposição entre as peças. Como em Toledo et al. (2013), na nossa abordagem a placa é discretizada por uma malha regular e é possível evitar a sobreposição entre as peças garantindo que, se uma peça do tipo t na rotação r é alocada em um ponto d da placa então, a peça t' na rotação r' não pode ser alocada nos pontos interiores ao *nofit-polygon* entre elas. O conjunto de pontos interiores ao *nofit-polygon* entre a peça t na rotação r alocada no ponto d e a peça t' na rotação r' é definido pelo conjunto $DNFP_{trd}^{t'r'}$. A Figura 2 ilustra o conjunto $DNFP_{trd}^{t'r'}$ constituído pelos círculos apresentados dentro da região hachurada que define o *nofit-polygon* entre estas peças. No caso ilustrado, a peça t na rotação r está fixa no ponto d impedindo a peça t' na rotação r' de ser posicionada nos pontos presentes no interior do $DNFP_{trd}^{t'r'}$.

Além da não-sobreposição, para garantir a factibilidade da solução, é necessário que as peças sejam posicionadas inteiramente dentro da placa. Garantimos esta restrição utilizando o conceito de *inner-fit polygon* que define a região factível de posicionamento de cada peça dentro da placa. Considerando uma peça t na rotação r , seu *inner-fit polygon* é dado por IFP_{tr} . A Figura 3 ilustra o *inner-fit polygon* de uma peça. Na figura é possível notar que a peça t na rotação r somente será posicionada inteiramente dentro da placa se estiver dentro da região hachurada que define o IFP_{tr} . No caso abordado, IFP_{tr} sempre será dado por um retângulo que pode ser facilmente definido utilizando os parâmetros l_{tr}^{\min} , l_{tr}^{\max} , h_{tr}^{\min} e h_{tr}^{\max} apresentados na Figura 1.

3. Abordagem por programação por restrições

A definição do domínio das variáveis é de grande importância na resolução do problema via programação por restrições pois o método de solução se baseia na redução de domínios. Desta


 Figura 2: Exemplificando o conjunto $DNFP_{trd}^{t'r'}$.

 Figura 3: Ilustrando o IFP_{tr} : região factível de posicionamento da peça t na rotação r na placa.

forma, o sucesso do método de solução está diretamente ligado a definição das variáveis. Nesta seção definimos as variáveis e restrições para a resolução do problema de posicionamento de peças irregulares utilizando programação por restrições.

3.1. Variáveis e domínios

As variáveis utilizadas para a representação dos pontos de posicionamento das peças têm domínio binário, para isto, para cada ponto $d \in D$ é definida uma variável binária δ_{trd} para cada peça do tipo $t = 1, \dots, T$ na rotação $r = 1, \dots, R_t$. Esta variável assume o valor 1 se a peça t na rotação r está posicionada no ponto d e zero caso contrário. Note que a restrição de que as peças devem estar inteiramente contidas dentro da placa pode ser garantida na definição de variáveis, fazendo $\delta_{trd} = 0$ se $d \notin D \cap IFP_{tr}$.

3.2. Restrições básicas

A restrição mais importante a satisfazer nos problemas de corte e empacotamento de peças irregulares é a que garante que as peças não se sobrepõem. Para definir tal restrição, assume-se que a peça do tipo t na rotação r está posicionada sobre um ponto $d \in D$. Para evitar que esta peça se sobreponha com outras peças, temos que garantir que toda peça t' na rotação r' não seja alocada nos pontos $d' \in DNFP_{trd}^{t'r'}$, ou seja, todas as variáveis $\delta_{t'r'd'}$ devem ser iguais a zero. Estas condições são garantidas pelas restrições (1).

$$\text{Se } (\delta_{trd} = 1) \text{ então } (\delta_{t'r'd'} = 0), \\ t = 1, \dots, T, t' = t, \dots, T, r = 1, \dots, R_t, r' = 1, \dots, R_{t'}, d \in D, d' \in DNFP_{trd}^{t'r'} \quad (1)$$

Esta restrição deve ser propagada (executada) sempre que, durante a busca, a variável de decisão δ_{trd} assumir o valor 1. Note-se que para cada par de peças, em seus respectivos tipos e rotações, há restrições que impedem estas de serem posicionadas em pontos que gerem sobreposição, desta forma a sobreposição entre as peças é eliminada. Sempre que uma restrição deste tipo é propagada, o valor de uma única variável é fixada em zero, ou seja, o domínio de uma variável é inferido. O número de restrições necessário para eliminar a sobreposição entre as peças é dado por $T \times T \times R_t \times R_{t'} \times \#(D \cap IFP_{tr} \cap IFP_{t'r'}) \times \#DNFP_{trd}^{t'r'} / 2$, em que $\#$ representa a cardinalidade do conjunto.

Em algumas variações do problema, o corte de uma peça em maior quantidade que sua demanda (q_t) pode não ser atrativo, então, para estes casos, uma nova restrição para o controle da demanda deve ser imposta. A restrição (2) faz a contagem do número de pontos d referentes à peça t em qualquer rotação que são ativos na solução e requer que este número não seja maior que a quantidade demandada q_t das peças do tipo t .

$$\text{count}_{d \in D, r=1, \dots, R_t} (\delta_{trd} = 1) \leq q_t, \quad t = 1, \dots, T; \quad (2)$$

O número de restrições necessárias para representar a demanda máxima, quando esta existe, é igual ao número de tipos de peças T , que é de magnitude muito inferior quando comparado ao número de restrições de não-sobreposição. Estas restrições podem no entanto limitar substancialmente o número de soluções factíveis do problema, levando o método a uma convergência mais rápida.

Definidas as restrições de factibilidade, resta definir o objetivo para a busca. No problema abordado, devemos extrair o máximo valor possível realizando o corte das peças a partir da placa. Considere que cada peça do tipo t tem valor v_t , a função objetivo (3) visa maximizar o valor total do corte das peças.

$$\text{maximizar} \sum_{d \in D} \sum_{t=1}^T \sum_{r=1}^{R_t} v_t \delta_{trd}. \quad (3)$$

Se o valor v_t é diretamente proporcional a área da peça t , então, esta função objetivo levará a utilizar o máximo de material da placa, o que é equivalente a minimizar o desperdício.

4. Experimentos computacionais

Nesta seção apresentamos os testes computacionais realizados utilizando a abordagem proposta. Estes testes foram realizados em um computador com processador Intel Xeon Processor E5-2450 com 16 threads e 32GB de memória utilizando o sistema operacional Scientific Linux 6. Para resolução da formulação apresentada, o software de otimização CPLEX 12.6.1 foi utilizado. O tempo limite para a execução do método de solução foi restringido a 1 hora.

4.1. Instâncias

Para realizar os testes computacionais, algumas instâncias conhecidas da literatura foram adaptadas ao problema estudado. A Tabela 1 apresenta estas instâncias e suas características originais. A primeira coluna da tabela apresenta o nome da instância, as colunas dois e três reportam o número de tipos de peças e o número total de peças demandadas, respectivamente. As rotações disponíveis para cada tipo de peça são descritos na coluna quatro. A altura da placa utilizada é apresentada na quinta coluna. Por fim, a coluna seis reporta o artigo em que a instância foi utilizada pela primeira vez. Vale ressaltar que todas estas instâncias estão disponíveis no endereço eletrônico do ESICUP¹.

¹<http://paginas.fe.up.pt/esicup>

Tabela 1: Instâncias utilizadas para a realização dos testes computacionais.

Instância	Tipos de peças	Número de peças	Rotações	Altura da placa	Origem
Blaz1	7	4 de cada tipo	0-180	15	Błażewicz et al. (1993)*
Blaz2	4	4 de cada tipo	0-180	15	Błażewicz et al. (1993)*
Shapes0	4	15, 7, 9 12	0	40	Oliveira et al. (2000)
Shapes1	4	15, 7, 9 12	0-180	40	Oliveira et al. (2000)
Fu	12	1 de cada tipo	0-90-180-270	38	Fujita et al. (1993)

* Com a aproximação poligonal das peças proposta em Oliveira et al. (2000).

Como nestas instâncias apenas a altura H da placa é especificada, definimos o comprimento L da placa sendo igual à altura da placa. O número de peças a serem posicionadas na placa é desconsiderado para a versão do problema que não tem limite de demanda e definido como limite para a variante do problema em que a demanda é restrita. Ainda, o valor v_t associado a cada peça é definido como a área da peça.

Estas instâncias foram selecionadas por possuírem diferentes características tais como número de tipos de peças, número de rotações, tamanho de placa e formato de peças. Nosso objetivo foi analisar melhor a estratégia de solução aqui proposta.

Três discretizações foram utilizadas para cada instância. Em todas as discretizações, a malha de pontos é dada por uma malha regular, em que os pontos são dados por coordenadas inteiras múltiplas de um parâmetro Δ que define a granularidade da malha. As três discretizações diferem apenas na definição do parâmetro Δ que é 1, 2 ou 4. Quanto maior é o parâmetro Δ , menos refinada é a malha. O espaço de solução de uma malha mais refinada contém todo o espaço de solução das malhas menos finas que esta, promovendo uma relação entre as representações.

4.2. Resultados computacionais: caso de demanda irrestrita

A Tabela 2 resume os resultados computacionais para as instâncias apresentadas na Seção 4.1. Na tabela, a primeira coluna apresenta o nome da instância. As colunas subsequentes apresentam, em pares, os resultados para a malha regular gerada com o parâmetro $\Delta = 1$ (colunas dois e três), $\Delta = 2$ (colunas quatro e cinco) e $\Delta = 4$ (colunas seis e sete), respectivamente. Nas duas colunas sob cada tipo de malha distinto são reportados o valor da melhor solução encontrada e o tempo computacional, respectivamente. Quando o tempo limite de 1 hora é atingido reportamos o tempo como TL.

É importante ressaltarmos que, diferentes dos métodos de solução para formulações de programação inteira mista, os métodos de solução para programação por restrições não utilizam informação sobre o limitante inferior no método de solução. Desta forma, não é possível apresentar alguma medida que mostre o quão distante a melhor solução obtida pode estar da solução ótima. A certeza é que esta distância é igual a zero quando o método atinge a otimalidade antes do tempo limite.

 Tabela 2: Resultados computacionais para o *PPPI* sem restrições de demanda sobre as peças.

	$\Delta = 1$		$\Delta = 2$		$\Delta = 4$	
	Solução	Tempo	Solução	Tempo	Solução	Tempo
Blaz1	215,5	TL	196,0	TL	164,5	44,7
Blaz2	178,0	TL	156,0	TL	134,0	0,2
Shapes0	1096,0	TL	1260,0	TL	992,0	TL
Shapes1	1100,0	TL	1226,0	TL	1000,0	TL
Fu	-	-	-	-	1346,0	TL

É possível analisar que, para as granularidades de malha mais refinadas ($\Delta = 1$ e $\Delta = 2$), a otimalidade nunca é alcançada antes do tempo limite. Isso ocorre pois, apesar do método de programação por restrições ser uma estratégia eficiente para explorar espaços de solução e inferir domínios, ele não é eficiente em relação à prova de otimalidade da solução.

Em relação à discretização, como esperado ela pode alterar drasticamente a qualidade da solução. Por exemplo, nas instâncias Blaz1 e Blaz2 o valor da função objetivo é maior, e portanto de melhor qualidade para as malhas mais refinadas, mesmo quando a otimalidade é provada para a malha com $\Delta = 4$. Em contrapartida, malhas mais refinadas também podem resultar em um espaço de solução muito grande, que o método de solução pode ter dificuldades em explorar. As instâncias Shapes0 e Shapes1 ilustram este fato, pois para elas o método de solução pôde encontrar soluções de melhor qualidade para a malha com $\Delta = 2$. Além disso, para instâncias de maior porte, o método de solução pode não ser capaz de encontrar uma solução factível em menos de 1 hora de execução devido ao tamanho do problema ou à falta de recursos da máquina. A instância Fu é um bom exemplo deste caso, no qual somente foi possível obter soluções factíveis para o problema em que a granularidade da malha era mais esparsa ($\Delta = 4$).

4.3. Resultados computacionais: caso de demanda restrita

Problemas cuja demanda pelas peças é limitada são abordados nesta seção. A Tabela 3 apresenta os resultados computacionais para as instâncias apresentadas na Tabela 1 levando em consideração a demanda pelas peças. As colunas desta tabela contêm o mesmo tipo de informação apresentada na Tabela 2.

Tabela 3: Resultados computacionais para o *PPPI* com demanda restrita.

	$\Delta = 1$		$\Delta = 2$		$\Delta = 4$	
	Solução	Tempo	Solução	Tempo	Solução	Tempo
Blaz1	190.0	TL	178.0	TL	158.5	57.5
Blaz2	168.0	TL	154.0	TL	121.5	0.2
Shapes0	1032.0	TL	1060.0	TL	920.0	TL
Shapes1	1056.0	TL	1036.0	TL	980.0	TL
Fu	-	-	-	-	1083.0	97.2

Primeiramente, vale ressaltar que os resultados computacionais desta seção não são diretamente comparáveis com os apresentados na Seção 4.2, pois com a restrição de demanda o espaço de solução a ser explorado é diferente.

Características similares às apresentadas nos testes da Seção 4.2 podem ser notadas na Tabela 3. Exceto para algumas instâncias com malha com granularidade $\Delta = 4$, a prova de otimalidade das soluções não foi atingida. No caso das instâncias Blaz1 e Blaz2, o uso de uma malha menos fina proporcionou uma redução da qualidade da solução. No caso da instância Fu, com a maior granularidade da malha foi possível provar a otimalidade. Isso ocorreu por que, com a demanda limitada, o método de solução conseguiu em pouco tempo encontrar uma solução na qual todas as peças foram alocadas na placa, conseguindo assim provar a otimalidade da solução. Nas malhas mais refinadas, pelo alto número de restrições e variáveis, não foi possível encontrar soluções factíveis para esta instância dentro do limite de tempo.

Para as instâncias Blaz1, Blaz2 e Shapes1, as melhores soluções foram encontradas utilizando a granularidade de malha $\Delta = 1$. Para a instância Shapes0, a granularidade $\Delta = 2$ permitiu obter a melhor solução. E, como já referido, $\Delta = 4$ foi a única granularidade que possibilitou o encontro da solução para a instância Fu.

5. Conclusões

Neste trabalho são apresentados métodos de resolução, via programação por restrições, do problema de posicionamento de peças irregulares com e sem restrições de demanda. A abordagem

proposta se mostra efetiva na busca por soluções factíveis e de boa qualidade para o problema. No entanto, e conforme esperado, a prova de otimalidade está fortemente condicionada pela dimensão do problema e pela granularidade da representação. A formulação apresentada é bastante flexível e pode ser facilmente adaptada para outras variações de problemas de corte e empacotamento de peças irregulares.

Como estudo futuro, pretendemos avaliar outras representações de variáveis e seus domínios para a representação do problema com o objetivo de aumentar a eficiência do método. Também, investiremos na criação de novas restrições que ajudem a reduzir os domínios de busca do método para promover a aceleração da prova de otimalidade da solução.

Agradecimentos

Esta pesquisa foi financiada pela FAPESP (2014/10740-4, 2012/18653-8, 2010/10133-0) e CNPq (306918/2014-5). Este trabalho também foi financiado em Portugal pelo Programa Operacional Regional (ON.2–O Novo Norte), parte do National Strategic Reference Framework (NSRF), através do European Regional Development Fund (ERDF), e por fundos nacionais da Fundação para a Ciência e a Tecnologia (FCT), nos projetos NORTE-07-0124-FEDER-000057 e NORTE-07-0124-FEDER-000059.

Referências

- Bennell, J. A.; Oliveira, J. F.** (2008). The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, 184(2), 397 – 415.
- Błażewicz, J.; Hawryluk, P.; Walkowiak, R.** (1993). Using a tabu search approach for solving the two-dimensional irregular cutting problem. *Annals of Operations Research*, 41(4), 313–325.
- 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.
- Clautiaux, F.; Jouglet, A.; Carlier, J.; Moukrim, A.** (2008). A new constraint programming approach for the orthogonal packing problem. *Computers & Operations Research*, 35(3), 944 – 959.
- Dalalah, D.; Khrais, S.; Bataineh, K.** (2014). Waste minimization in irregular stock cutting. *Journal of Manufacturing Systems*, 33(1), 27 – 40.
- Daniels, K.; Milenkovic, V. J.** (1997). Multiple Translational Containment Part I: An Approximate Algorithm. *Algorithmica*, 19(1-2), 148–182.
- Fujita, K.; Akagji, S.; Kirokawa, N.** (1993). Hybrid approach for optimal nesting using a genetic algorithm and a local minimization algorithm. *Advances in Design Automation; American Society of Mechanical Engineers*, 65(1), 477–484.
- Milenkovic, V. J.** (1997). Multiple Translational Containment Part II: Exact Algorithms. *Algorithmica*, 19(1-2), 183–218.
- Oliveira, J. F.; Gomes, A. M.; Ferreira, J. S.** (2000). TOPOS - A new constructive algorithm for nesting problems. *OR Spektrum*, 22(2), 263–284.
- Ribeiro, C.; Carravilla, M. A.** (2008). A global constraint for nesting problems. *Artificial Intelligence Review*, 30(1-4), 99–118.
- Salas, I.; Chabert, G.; Goldsztejn, A.** (2014). The non-overlapping constraint between objects described by non-linear inequalities. In *Principles and Practice of Constraint Programming*, volume 8656 of *Lecture Notes in Computer Science*, pages 672–687. Springer International Publishing.
- Saldanha, R.; Morgado, E.** (2003). Solving set partitioning problems with global constraint propagation. In *Progress in Artificial Intelligence*, volume 2902 of *Lecture Notes in Computer Science*, pages 101–115. Springer Berlin Heidelberg.
- 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. *International Journal of Production Economics*, 145(2), 478 – 487.

Valle, A. M. D.; de Queiroz, T. A.; Miyazawa, F. K.; Xavier, E. C. (2012). Heuristics for two-dimensional knapsack and cutting stock problems with items of irregular shape. *Expert Systems with Applications*, 39(16), 12589–12598.

