Modelos matemáticos para o problema de
empacotamento em faixas de peças irregulares

Marcos Okamura Rodrigues

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Modelos matemáticos para o problema de empacotamento em faixas de peças irregulares¹

Marcos Okamura Rodrigues

***Orientadora:* Profa. Dra. Franklina Maria Bragion de Toledo**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

**USP – São Carlos
Março de 2015**

¹ Este trabalho foi financiado pela FAPESP (processo 2013/14147-3) no período de 10/2013 a 02/2015 e pelo CNPq (processo 131992/2013-9) no período de 03/2013 a 09/2013.

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

R969m Rodrigues, Marcos Okamura
Modelos matemáticos para o problema de
empacotamento em faixas de peças irregulares /
Marcos Okamura Rodrigues; orientadora Franklina
Maria Bragion de Toledo. -- São Carlos, 2015.
70 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2015.

1. Empacotamento em faixas. 2. Peças irregulares.
3. Cobertura por cliques. 4. Programação inteira
mista. I. Toledo, Franklina Maria Bragion de,
orient. II. Título.

Aos meus pais, Antônio e Keiko.

Agradecimentos

À minha orientadora, Prof^a. Dr^a. Franklina Toledo, pelas ideias, indagações, críticas e sugestões discutidas em reuniões que deram origem a este trabalho.

Ao meu orientador do trabalho de conclusão de curso, Prof. Dr. Adilson Bonifácio, pela colaboração, dedicação e disponibilidade durante toda a pesquisa.

Ao meu orientador de iniciação científica e coorientador do trabalho de conclusão de curso, Prof. Dr. Robinson Hoto, pela introdução à pesquisa na área de otimização.

Aos meus pais, Antônio e Keiko, grandes responsáveis pela minha educação e formação.

Aos meus irmãos, Fernando e Danielli, pelo incentivo à minha formação acadêmica.

Aos meus amigos e colegas da Universidade de São Paulo (USP), em especial aos membros do Laboratório de Otimização (LOt), pelo apoio e motivação durante todo o curso de mestrado.

Aos meus amigos e colegas da Universidade Estadual de Londrina (UEL), pelo apoio e motivação durante todo o curso de graduação.

A todos os professores e pesquisadores que me inspiraram a seguir a carreira acadêmica.

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), pelo apoio financeiro na bolsa de mestrado.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pelo apoio financeiro nas bolsas de iniciação científica júnior, iniciação científica e mestrado.

Ao Instituto Nacional de Matemática Pura e Aplicada (IMPA), pela criação do Programa de Iniciação Científica e Mestrado (PICME).

Ao meu assessor ad-hoc da FAPESP, pela colaboração para analisar e emitir os pareceres de forma voluntária.

Aos membros da banca, pela atenção e disponibilidade.

A todos aqueles que de alguma forma contribuíram para a realização deste trabalho.

*“Because my mathematics has its origin in a real problem
doesn’t make it less interesting to me—just the other way around,
I find it makes the puzzle I am working on all the more exciting.
I get satisfaction out of knowing that I’m working on a relevant problem.”*
(Dantzig, George B.)

Resumo

RODRIGUES, M. O. **Modelos matemáticos para o problema de empacotamento em faixas de peças irregulares**. 2015. 70 p. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2015.

O problema de empacotamento em faixas de peças irregulares consiste em cortar um conjunto de peças bidimensionais a partir de um objeto de largura fixa utilizando o menor comprimento possível. Apesar de sua importância econômica para diversos setores industriais, há poucos trabalhos que abordam o problema de forma exata devido a sua dificuldade de resolução. Recentemente, [Toledo et al. \(2013\)](#) propuseram um modelo inteiro misto para este problema, no qual as peças são posicionadas em uma malha de pontos. Este modelo obteve bons resultados, provando a otimalidade para instâncias com até 21 peças. No entanto, o modelo possui um grande número de restrições de não-sobreposição, que cresce rapidamente de acordo com a discretização utilizada e a quantidade de peças distintas que devem ser alocadas. Neste trabalho, são propostas novas formulações matemáticas baseadas neste modelo, com o objetivo de reduzir o número de restrições. Na primeira abordagem, são propostos dois modelos reduzidos que mostraram ser eficientes para instâncias com poucas repetições de peças. Na segunda abordagem, foi proposto um modelo de cobertura por cliques para o problema. Este modelo obteve desempenho igual ou superior ao modelo da literatura para todas as instâncias avaliadas, obtendo uma solução ótima para instâncias com até 28 peças.

Palavras-chaves: Empacotamento em faixas, Peças irregulares, Cobertura por cliques, Programação inteira mista

Abstract

RODRIGUES, M. O. **Mathematical models for the irregular strip packing problem**. 2015. 70 p. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2015.

The irregular strip packing problem consists of cutting a set of two-dimensional pieces from an object of fixed width using the smallest possible length. Despite its economic importance for many industrial sectors, few exact studies have been made on this problem due to its difficulty of resolution. Recently, [Toledo et al. \(2013\)](#) proposed a mixed-integer model to this problem in which the pieces are placed on a grid. This model has worked successfully proving the optimality for instances up to 21 pieces. However, the model has a large number of non-overlapping constraints, which grows quickly in accordance with the discretization resolution and number of distinct pieces. In this work, we propose new mathematical formulations based on this model in order to reduce the number of constraints. In the first approach, we present two reduced models that have shown to be effective for instances with few repetitions of pieces. In the second approach, it was proposed a clique covering model for the problem. This model achieved a greater or equal performance than the literature for all instances, getting an optimal solution for instances up to 28 pieces.

Keywords: Strip packing, Nesting, Irregular, Clique covering, Mixed integer programming

Lista de ilustrações

Figura 1 – Um exemplo de plano de corte de uma indústria de vestuário.	3
Figura 2 – Representação de uma peça usando <i>raster method</i>	4
Figura 3 – Exemplo de posições relativas entre duas peças.	5
Figura 4 – Representação de <i>phi-functions</i> entre retângulos e círculos.	5
Figura 5 – Construção dos <i>nofit polygons</i> entre um quadrado (<i>i</i>) e um triângulo (<i>j</i>).	6
Figura 6 – Exemplo de <i>inner-fit polygon</i>	6
Figura 7 – Heurísticas <i>bottom-left</i> e <i>bottom-left-fill</i>	8
Figura 8 – Fatias utilizadas para definir o complemento do <i>nofit polygon</i>	12
Figura 9 – Representação de um tipo de peça.	15
Figura 10 – Exemplo de <i>inner-fit polygon</i> na malha de pontos.	15
Figura 11 – Construção do <i>nofit polygon</i> na malha de pontos.	16
Figura 12 – Problemas de empacotamento bidimensionais para um tipo de peça.	19
Figura 13 – Problema de empacotamento para dois tipos de peça.	20
Figura 14 – Tipos de peça das instâncias RCO.	22
Figura 15 – Tipos de peça das instâncias BLAZEWCZ.	22
Figura 16 – Tipos de peça das instâncias SHAPES.	22
Figura 17 – Soluções obtidas pelos modelos propostos para a instância SHAPES9.	25
Figura 18 – Heurística RLF	32
Figura 19 – Grafo 1.	33
Figura 20 – Grafo 2.	35
Figura 21 – Grafo 3.	38
Figura 22 – Exemplo de um problema de empacotamento em faixas de peças irregu- lares.	40
Figura 23 – <i>Nofit polygons</i> na malha dos pontos.	40
Figura 24 – Exemplo de grafo de conflitos.	40
Figura 25 – Soluções iniciais utilizadas para o grupo RCO.	65
Figura 26 – Soluções iniciais utilizadas para o grupo BLAZEWCZ.	66
Figura 27 – Soluções iniciais utilizadas para o grupo SHAPES.	67
Figura 28 – Melhores soluções obtidas para o grupo RCO.	68
Figura 29 – Melhores soluções obtidas para o grupo BLAZEWCZ.	69
Figura 30 – Melhores soluções obtidas para o grupo SHAPES.	70

Lista de tabelas

Tabela 1 – Características das instâncias.	23
Tabela 2 – Avaliação dos modelos propostos.	24
Tabela 3 – Comparação entre o modelo dos pontos e o modelo reduzido I.	25
Tabela 4 – Características dos grafos das instâncias.	42
Tabela 5 – Cálculo das coberturas de vértices por cliques.	43
Tabela 6 – Cálculo das coberturas de arestas por cliques.	44
Tabela 7 – Cardinalidade das coberturas de arestas por cliques.	44
Tabela 8 – Avaliação dos modelos propostos utilizando a heurística RLF.	45
Tabela 9 – Avaliação dos modelos propostos utilizando a heurística RLFM.	46
Tabela 10 – Comparação dos resultados com a literatura.	47
Tabela 11 – Avaliação das desigualdades válidas para o modelo dos pontos.	52
Tabela 12 – Avaliação das desigualdades válidas para o modelo de cobertura por cliques.	52
Tabela 13 – Avaliação dos modelos com adição das desigualdades válidas (7.5).	53
Tabela 14 – Cálculo do limitante inferior.	54
Tabela 15 – Avaliação do uso do novo limitante inferior e solução inicial para o modelo dos pontos.	55
Tabela 16 – Avaliação do uso do novo limitante inferior e solução inicial para o modelo de cobertura por cliques.	56

Sumário

1	INTRODUÇÃO	1
2	O PROBLEMA DE EMPACOTAMENTO EM FAIXAS DE PEÇAS IRREGULARES	3
2.1	Descrição do problema	3
2.2	Conceitos geométricos básicos	4
2.3	Modelos matemáticos e métodos exatos	6
2.4	Métodos heurísticos	7
2.5	Considerações	9
3	MODELOS DA LITERATURA	11
3.1	Modelo de Fischetti e Luzzi (2009)	11
3.2	Modelo de Álvarez-Valdés, Martínez e Tamarit (2013)	13
3.3	Modelo de Toledo et al. (2013)	14
4	MODELOS REDUZIDOS	17
4.1	Modelo reduzido I	17
4.2	Modelo reduzido II	18
4.3	Estimativa para o valor de M	19
4.4	Experimentos computacionais	21
4.5	Considerações	26
5	PROBLEMAS DE COBERTURA DE GRAFOS POR CLIQUES . . .	27
5.1	Conceitos de grafos	27
5.2	Cobertura de vértices	28
5.2.1	Heurística RLF	30
5.2.2	Geração de cliques maximais	31
5.3	Cobertura de arestas	33
5.3.1	Heurística de Kellerman	35
5.3.2	Pós-processamento de Kou, Stockmeyer e Wong	36
6	MODELO DE COBERTURA POR CLIQUES	39
6.1	Modelo proposto	39
6.2	Experimentos computacionais	41
6.2.1	Cálculo da cobertura de vértices por cliques	42
6.2.2	Cálculo da cobertura de arestas por cliques	42
6.2.3	Avaliação dos modelos de cobertura por cliques	44

6.2.4	Comparação com a literatura	45
6.3	Considerações	46
7	DESIGUALDADES VÁLIDAS, LIMITANTE INFERIOR E SOLU- ÇÃO INICIAL	49
7.1	Desigualdades válidas e limitante inferior	49
7.2	Experimentos computacionais	51
7.2.1	Avaliação das desigualdades válidas	51
7.2.2	Cálculo do limitante inferior	53
7.2.3	Avaliação do limitante inferior e solução inicial	54
7.3	Considerações	55
8	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	57
	Referências	59
	APÊNDICE A – SOLUÇÕES	65
A.1	Soluções iniciais	65
A.2	Melhores soluções	68

1 Introdução

Problemas de corte e empacotamento de peças irregulares possuem aplicações em diversas indústrias, por exemplo, na fabricação de calçados e peças de vestuário, na fabricação de móveis, no corte de placas metálicas, vidro e papel, entre outras. Esses problemas possuem grande relevância tanto pelo aspecto econômico quanto ambiental. Do ponto de vista econômico, o objetivo é reduzir os custos de produção realizando o processo de corte de forma a minimizar o desperdício de matéria-prima. O melhor aproveitamento da matéria-prima também proporciona um ganho ambiental.

Esses problemas são NP-completos (FOWLER; PATERSON; TANIMOTO, 1981) e combinam a dificuldade dos problemas de corte e empacotamento com a complexidade geométrica de posicionar peças irregulares convexas e não-convexas sem provocar sobreposição entre elas. Neste sentido, é necessário desenvolver um conjunto de ferramentas para tratar a geometria das peças, o que não é uma tarefa trivial e pode ser uma barreira para novos pesquisadores na área (BENNEL; OLIVEIRA, 2008).

Dentro do contexto de problemas de corte e empacotamento de peças irregulares, o problema de empacotamento em faixas (*irregular strip packing problem*) consiste em cortar um conjunto de peças bidimensionais a partir de uma placa de largura fixa e comprimento infinito. O objetivo é minimizar o comprimento da placa utilizado para cortar todas as peças. De acordo com a tipologia proposta por Wäscher, Haußner e Schumann (2007), esse problema pode ser classificado como um problema de corte bidimensional de dimensão aberta com peças irregulares.

Devido à dificuldade de resolução do problema, a maioria dos métodos propostos para tratar o problema são heurísticos. Dowsland e Dowsland (1995) e Bennell e Oliveira (2009) apresentam duas revisões de métodos heurísticos para o problema. De forma geral, os métodos heurísticos podem ser divididos em heurísticas construtivas e heurísticas de melhoria.

Na linha de heurísticas construtivas, há vários trabalhos na literatura baseados nas heurísticas *bottom-left* (OLIVEIRA; GOMES; FERREIRA, 2000; DOWSLAND; VAID; DOWSLAND, 2002) e *bottom-left-fill* (BURKE et al., 2006; BURKE et al., 2010), que consistem em inserir sequencialmente cada peça na posição mais à esquerda possível e em caso de empate, mais abaixo da placa.

Na linha de heurísticas de melhoria, diversas técnicas foram utilizadas para resolver o problema, dentre as quais destacam-se: algoritmo genético (BABU; BABU, 2001), algoritmos híbridos (BENNEL; DOWSLAND, 2001; GOMES; OLIVEIRA, 2006), *guided local search* (EGEBLAD; NIELSEN; ODGAARD, 2007; UMETANI et al., 2009), *iterated*

local search (IMAMICHI; YAGIURA; NAGAMOCCHI, 2009), *beam search* (BENNEL; SONG, 2010), *extended local search* (LEUNG; LIN; ZHANG, 2012), *simulated annealing* com regiões livres de colisões (SATO; MARTINS; TSUZUKI, 2012) e *cuckoo search* (ELKERAN, 2013).

Por outro lado, há poucos trabalhos que abordam o problema de empacotamento em faixas de peças irregulares de forma exata. Carravilla, Ribeiro e Oliveira (2003) apresentaram um método exato baseado em programação por restrições para a resolução do problema sujeito ao posicionamento das peças em pontos discretos de uma malha. Em 2009, Fischetti e Luzzi (2009) propuseram um modelo de programação inteira mista que utiliza variáveis contínuas para definir o posicionamento das peças e variáveis binárias para evitar a sobreposição entre elas. Álvarez-Valdés, Martínez e Tamarit (2013) estenderam o modelo linear de compactação de Gomes e Oliveira (2006) para o problema de corte de peças convexas. Além disso, os autores reformularam o modelo de Fischetti e Luzzi (2009) e desenvolveram um método *branch-and-bound* para sua resolução. Toledo et al. (2013) apresentaram um modelo inteiro-misto denominado modelo dos pontos, no qual as variáveis de decisão estão associadas aos tipos de peça e aos pontos de uma malha em que as peças podem ser alocadas na placa. As maiores instâncias da literatura foram resolvidas de forma exata utilizando esse modelo. Vale destacar que essas soluções estão sujeitas à dimensão da malha utilizada.

O objetivo deste trabalho é estudar o modelo dos pontos, buscando formulações matemáticas mais eficientes. Como o modelo possui um conjunto com grande número de restrições de não-sobreposição, substituiremos este conjunto por outro de tamanho reduzido, resultando em formulações mais compactas para o problema. Neste sentido, são propostas duas abordagens. Na primeira abordagem, são propostos modelos reduzidos que agregam informações de diversas restrições de não-sobreposição simultaneamente. Na segunda abordagem, é proposto um modelo de cobertura por cliques com dois objetivos: reduzir o número de restrições e melhorar a qualidade da relaxação linear do modelo.

O trabalho está organizado conforme descrito a seguir. No Capítulo 2, é feita uma revisão da literatura sobre o problema de empacotamento em faixas de peças irregulares. No Capítulo 3, são apresentados os modelos propostos na literatura para o problema. No Capítulo 4, são propostos dois modelos reduzidos para o problema. No Capítulo 5, são apresentadas revisões sobre os problemas de cobertura de vértices e de arestas por cliques. No Capítulo 6, é proposto um modelo de cobertura por cliques para o problema de empacotamento em faixas de peças irregulares. No Capítulo 7, são discutidos os usos de desigualdades válidas, limitantes inferiores e soluções iniciais nos modelos dos pontos e de cobertura por cliques. Por fim, no Capítulo 8, são apresentadas algumas considerações finais sobre o trabalho e são indicadas perspectivas de trabalhos futuros.

2 O problema de empacotamento em faixas de peças irregulares

Neste capítulo é apresentada uma revisão sobre o problema de empacotamento em faixas de peças irregulares. Na Seção 2.1, é definido formalmente o problema. Na Seção 2.2, são definidos os conceitos geométricos básicos necessários para a compreensão do presente trabalho. Nas Seções 2.3 e 2.4, são discutidos modelos matemáticos, métodos exatos e heurísticos propostos para o problema. Na Seção 2.5, são feitas algumas considerações sobre este capítulo.

2.1 Descrição do problema

O problema de empacotamento em faixas de peças irregulares abordado neste trabalho pode ser formalmente descrito da seguinte forma. Sejam $\mathcal{P} = 1, \dots, P$ um conjunto de peças bidimensionais (não necessariamente distintas) e uma placa retangular de largura fixa W e de comprimento ilimitado. Além disso, considere que cada peça $i \in \mathcal{P}$ seja representada por um polígono e possua um ponto de referência $v_i = (x_i, y_i)$ a ela associado. A partir desse ponto é definida a localização da peça na placa. Assim, o problema consiste em encontrar um plano de corte ou empacotamento que posicione todas as peças $i \in \mathcal{P}$ no interior da placa sem provocar sobreposições, de tal forma que o comprimento utilizado z da placa seja mínimo. A Figura 1 ilustra um exemplo de plano de corte utilizado na fabricação de peças de vestuário. Observe a dificuldade de definir o posicionamento das peças devido às questões geométricas do problema.

Figura 1 – Um exemplo de plano de corte de uma indústria de vestuário.

Fonte: [BENNEL; OLIVEIRA, 2008](#), p. 398.

2.2 Conceitos geométricos básicos

Para evitar a sobreposição entre as peças, diversas técnicas podem ser utilizadas, por exemplo, trigonometria direta, *raster method*, *nofit polygon* e *phi-functions*. Uma revisão sobre essas técnicas é apresentada por [Bennell e Oliveira \(2008\)](#).

A técnica *raster method* consiste em discretizar a placa e as peças através de malhas que são representadas por matrizes. Para verificar se há sobreposição entre as peças, são feitas operações de soma e comparação entre essas matrizes. A Figura 2 ilustra um exemplo de representação de uma peça pelo *raster method* usando a codificação binária.

Figura 2 – Representação de uma peça usando *raster method*.

Fonte: [BENNEL; OLIVEIRA, 2008](#), p. 400.

A técnica de trigonometria direta consiste em avaliar a sobreposição a partir dos vértices e das arestas dos polígonos que representam as peças. Nessa abordagem, diversas verificações podem ser realizadas sequencialmente. [Bennell e Oliveira \(2008\)](#) sugerem a seguinte ordem de análise (Figura 3):

1. há sobreposição entre os retângulos envolventes de duas peças? (Figura 3 (a) - não, Figura 3 (b) - sim). Em caso negativo, as peças não se sobrepõem;
2. há sobreposição entre os retângulos envolventes das arestas de duas peças? (Figura 3 (b) - não, Figura 3 (c) - sim). Em caso negativo, as peças não se sobrepõem;
3. há interseção entre arestas de peças distintas? (Figura 3 (c) - sim, Figura 3 (d) - não). Em caso positivo, as peças se sobrepõem;
4. existe algum vértice de uma peça dentro do polígono de outra? (Figura 3 (d) - sim, Figura 3 (e) - não). Em caso positivo, as peças se sobrepõem e, caso contrário, as peças não se sobrepõem.

[Stoyan et al. \(2001\)](#) e [Stoyan et al. \(2004\)](#) propuseram as *phi-functions* para representar a posição relativa entre duas peças bidimensionais. As *phi-functions* são expressões matemáticas que podem ser utilizadas para calcular a distância e a sobreposição entre duas peças: se o valor da *phi-function* é maior que zero, as peças estão separadas;

Figura 3 – Exemplo de posições relativas entre duas peças.

Fonte: [BENNEL; OLIVEIRA, 2008](#), p. 402.

se o valor é exatamente zero, as peças se tocam; se o valor é menor que zero, as peças se sobrepõem. A Figura 4 ilustra *phi-functions* entre círculos e retângulos. A região destacada em negrito indica onde as peças se tocam e, conseqüentemente, a *phi-function* se anula.

Figura 4 – Representação de *phi-functions* entre retângulos e círculos. (a) Dois círculos. (b) Dois retângulos. (c) Círculo e retângulo.

Fonte: [STOYAN et al., 2001](#), p. 9 e 11.

No modelo de [Toledo et al. \(2013\)](#) e neste trabalho, a técnica de verificação de não-sobreposição adotada é o *nofit polygon*, que é definida a seguir. Dadas duas peças i e j , o *nofit polygon* $NFP_{i,j}$ é a região na qual o ponto de referência da peça j não pode ser posicionado, uma vez que causaria sobreposição com a peça i .

Podemos interpretar geometricamente o $NFP_{i,j}$ como a região definida pela trajetória seguida pelo ponto de referência de uma peça j quando ele desliza em volta de outra peça i e vice-versa (Figura 5).

Assim, conforme pode ser observado na Figura 5, temos três casos:

- se o ponto de referência da peça j pertence ao interior do $NFP_{i,j}$ então há sobreposição entre as peças;

Figura 5 – Construção dos *nofit polygons* entre um quadrado (i) e um triângulo (j).
 (a) Quadrado e triângulo; (b) Triângulo e quadrado.

- b) se o ponto de referência da peça j não pertence ao interior do $NFP_{i,j}$ então não há sobreposição entre as peças;
- c) se o ponto de referência da peça j está na fronteira do $NFP_{i,j}$ então as peças estão em contato, mas não há sobreposição.

Analogamente, definimos o *inner-fit polygon* IFP_i de uma peça i como a região da placa onde ela está inteiramente contida na placa. A Figura 6 ilustra o *inner-fit polygon* de um triângulo (destacado) numa placa retangular.

Figura 6 – Exemplo de *inner-fit polygon*.

2.3 Modelos matemáticos e métodos exatos

Na literatura, diversas abordagens foram empregadas para resolver o problema de empacotamento em faixas de peças irregulares. Devido à sua dificuldade de resolução, há poucos trabalhos que abordam o problema de forma exata. Por outro lado, diversos autores desenvolveram métodos heurísticos para resolver o problema.

As primeiras abordagens exatas para o problema foram baseadas em programação por restrições. [Ribeiro, Carravilla e Oliveira \(1999\)](#) desenvolveram uma estratégia de busca para resolução do problema de empacotamento em faixas de peças convexas. Os domínios iniciais das variáveis foram definidos de acordo com os pontos admissíveis onde o ponto de referência de cada polígono pode ser posicionado. Para cada par de peças foi criada

uma restrição de não-sobreposição baseada em seu respectivo *nofit polygon*. Carravilla, Ribeiro e Oliveira (2003) estenderam o trabalho anterior para peças irregulares convexas e não-convexas. Além disso, os autores criaram restrições para eliminação de simetria para peças com o mesmo formato. Em ambas as abordagens, as variáveis associadas aos pontos de referências das peças pertencem a um domínio discreto. Os autores encontraram a solução ótima para uma instância com sete peças, sujeita à discretização.

Em 2009, Fischetti e Luzzi propuseram o primeiro modelo de programação inteira mista para o problema. O modelo utiliza variáveis contínuas para definir o posicionamento das peças e variáveis binárias para evitar a sobreposição entre elas. A não-sobreposição é garantida através da imposição do posicionamento do ponto de referência de uma peça em uma das partições associadas ao complemento do *nofit polygon* entre cada par de peças. Os autores provaram a otimalidade de instâncias com até sete peças.

Álvarez-Valdés, Martínez e Tamarit (2013) definiram um procedimento para obter as partições descritas por Fischetti e Luzzi (2009) através de fatias horizontais (*horizontal slices*). Além disso, os autores estenderam o modelo linear de compactação de Gomes e Oliveira (2006) para o problema de empacotamento em faixas de peças convexas e reformularam o modelo de Fischetti e Luzzi (2009), utilizando a técnica de *lifting* nos limitantes das variáveis contínuas. Os autores também desenvolveram um método *branch-and-bound* que encontrou soluções ótimas para instâncias com até 16 peças.

Toledo et al. (2013) apresentaram um modelo inteiro-misto em que as variáveis de decisão são binárias e estão associadas aos tipos de peça e aos pontos de uma malha em que as peças podem ser alocadas na placa. Os autores conseguiram provar a otimalidade de instâncias com 16 peças (quatro tipos), 21 peças (sete tipos), 35 peças (um tipo) e 56 peças (dois tipos). Vale destacar que a solução ótima encontrada está sujeita à discretização da malha utilizada.

2.4 Métodos heurísticos

Diversas heurísticas foram propostas para o problema, como pode ser visto nas revisões descritas por Dowsland e Dowsland (1995) e Bennell e Oliveira (2009). Na linha de heurísticas construtivas, há vários trabalhos na literatura baseados nas heurísticas *bottom-left* (OLIVEIRA; GOMES; FERREIRA, 2000; DOWSLAND; VAID; DOWSLAND, 2002) e *bottom-left-fill* (BURKE et al., 2006; BURKE et al., 2010). Na linha de heurísticas de melhoria, matheurísticas e metaheurísticas, diversas técnicas foram utilizadas para resolver o problema.

As heurísticas *bottom-left* e *bottom-left-fill* são algoritmos gulosos utilizados para construir uma solução inicial para o problema. Dada uma sequência de peças, as heurísticas inserem cada peça na posição mais à esquerda possível e em caso de empate, mais abaixo

da placa. A heurística *bottom-left-fill* considera o preenchimento de espaços vazios entre as peças, ao contrário da *bottom-left*, conforme é ilustrado na Figura 7.

Figura 7 – Heurísticas *bottom-left* e *bottom-left-fill*. (a) *Bottom-left*. (b) *Bottom-left-fill*.
Fonte: DOWSLAND; VAID; DOWSLAND, 2002. p. 372.

Uma estratégia muito comum para resolver o problema é o posicionamento das peças baseado em uma sequência. Neste sentido, vários autores propuseram algoritmos de busca local baseados nessa estratégia. Jakobs (1996), Babu e Babu (2001) e Fischer e Dagli (2004) empregaram algoritmos genéticos para o problema considerando rotações fixas das peças. Jakobs (1996) utilizou uma aproximação das peças por retângulos enquanto que Babu e Babu (2001) e Fischer e Dagli (2004) empregaram a técnica de *raster method*. Gomes e Oliveira (2002) propuseram uma heurística que percorre uma vizinhança obtida por meio da troca da posição entre duas peças em uma determinada sequência. Bennell e Song (2010) desenvolveram um algoritmo *beam search* para o problema. Os autores representaram o problema por meio de uma lista de ordenação das peças, que é decodificada usando uma heurística construtiva baseada no algoritmo TOPOS (OLIVEIRA; GOMES; FERREIRA, 2000). Sato, Martins e Tsuzuki (2012) desenvolveram uma heurística *simulated annealing* utilizando a técnica de região livre de colisões (*collision free region*). Os autores também definiram uma ordem de posicionamento das peças, priorizando regiões de encaixes exatos (*exact fit* e *exact slide*) que representam vértices e arestas degenerados obtidos pela região livre de colisões.

Na linha de matheurísticas, destacam-se os algoritmos baseados em modelos de compactação e separação. Li e Milenkovic (1995) e Stoyan, Novozhilova e Kartashov (1996) propuseram modelos de programação linear de compactação e separação de peças para o problema. Os modelos de compactação são utilizados para deslocar as peças com o intuito de reduzir o comprimento utilizado da placa, enquanto que os de separação são utilizados para gerar uma configuração factível após remover a sobreposição entre as peças. Neste sentido, foram propostos algoritmos híbridos utilizando programação linear e busca tabu

(BENNEL; DOWSLAND, 2001) e programação linear e *simulated annealing* (GOMES; OLIVEIRA, 2006).

Recentemente, foram propostas heurísticas promissoras que a princípio permitem a sobreposição das peças na placa. Nessas heurísticas, dada uma solução factível, é criada uma placa auxiliar com comprimento menor e resolvido um problema de minimização de sobreposição. Caso seja encontrada uma nova solução factível, o processo é reexecutado, caso contrário, a solução obtida é um mínimo local e é necessário aplicar algum procedimento de perturbação. Egeblad, Nielsen e Odgaard (2007) e Umetani et al. (2009) desenvolveram algoritmos *guided local search* para o problema utilizando essa estratégia. Egeblad, Nielsen e Odgaard (2007) criaram uma busca local que percorre a vizinhança vertical e horizontal de um polígono com o objetivo de minimizar a área de sobreposição entre as peças. Umetani et al. (2009) propuseram o uso da profundidade de penetração direcional (*directional penetration depth*) para calcular a penalização da sobreposição entre as peças. Imamichi, Yagiura e Nagamochi (2009) empregaram um algoritmo *iterated local search* para o problema utilizando um método de separação baseado em programação não-linear. Leung, Lin e Zhang (2012) desenvolveram um algoritmo *extended local search* que utiliza uma busca tabu para evitar a repetição de movimentos das peças em ciclos. Além disso, os autores empregaram uma heurística de compactação na fase final do algoritmo para melhorar a solução obtida. Atualmente, a heurística híbrida *guided cuckoo search* proposta por Elkeran (2013) apresentou os melhores resultados para 14 das 15 instâncias da literatura avaliadas no artigo. Neste trabalho também é introduzida a técnica *pairwise clustering*, que consiste em agrupar duplas de polígonos com *nofit polygons* não-convexos.

2.5 Considerações

Neste capítulo foi apresentada uma revisão bibliográfica sobre o problema de empacotamento em faixas de peças irregulares, introduzindo formalmente conceitos geométricos básicos, modelos matemáticos e métodos exatos e heurísticos propostos para sua resolução. No capítulo seguinte, são detalhados os modelos matemáticos propostos na literatura para esse problema.

3 Modelos da literatura

Neste capítulo é apresentada uma revisão sobre os modelos matemáticos propostos na literatura para o problema de empacotamento em faixas de peças irregulares. Na Seção 3.1, é apresentado o modelo proposto por Fischetti e Luzzi (2009). Na Seção 3.2, é discutido o modelo proposto por Álvarez-Valdés, Martínez e Tamarit (2013). Na Seção 3.3, é detalhado o modelo de Toledo et al. (2013), que é o modelo utilizado como base para o desenvolvimento dos modelos propostos nos Capítulos 4 e 6 deste trabalho.

3.1 Modelo de Fischetti e Luzzi (2009)

Fischetti e Luzzi (2009) propuseram o primeiro modelo de programação inteira mista para o problema de empacotamento em faixas de peças irregulares. Nesse modelo, são utilizadas variáveis contínuas para definir o posicionamento das peças e variáveis binárias para evitar a sobreposição entre elas.

Sejam $\mathcal{P} = 1, \dots, P$ o conjunto de peças irregulares, W a largura da placa e z o comprimento da placa a ser minimizado. Além disso, sejam (x_i, y_i) as coordenadas do ponto de referência da peça $i \in \mathcal{P}$, ℓ_i o comprimento e w_i a largura do seu retângulo envolvente. Para garantir que cada peça seja posicionada na placa, temos que:

$$0 \leq x_i \leq z - \ell_i, \quad \forall i \in \mathcal{P}, \quad (3.1)$$

$$0 \leq y_i \leq W - w_i, \quad \forall i \in \mathcal{P}. \quad (3.2)$$

Para evitar a sobreposição entre duas peças i e j , a abordagem utilizada é o conceito de *nofit polygon* ($NFP_{i,j}$), que é a região na qual o ponto de referência da peça j não pode ser posicionado, uma vez que causaria sobreposição com a peça i . Neste sentido, os autores introduzem o conceito de fatias S_{ij}^k que definem o complemento do $NFP_{i,j}$. Desta forma, se o ponto de referência da peça j for posicionado em qualquer uma das fatias S_{ij}^k , não haverá sobreposição entre as duas peças (Figura 8).

Observe que a não sobreposição das peças é garantida quando o ponto de referência da peça j está localizado em uma determinada fatia $k = 1, \dots, m_{ij}$ que define o complemento do $NFP_{i,j}$. Assim, é necessário que esse ponto de referência atenda às desigualdades para todas as facetas $f = 1, \dots, t_{ij}^k$ que definem a fatia S_{ij}^k , ou seja:

$$\alpha_{ij}^{kf}(x_j - x_i) + \beta_{ij}^{kf}(y_j - y_i) \leq \gamma_{ij}^{kf}, \quad (3.3)$$

Figura 8 – Fatias utilizadas para definir o complemento do *nofit polygon*.

em que $\alpha_{ij}^{kf}, \beta_{ij}^{kf}, \gamma_{ij}^{kf}$ são os coeficientes da equação da reta que define a f -ésima reta da k -ésima fatia do complemento do $NFP_{i,j}$.

Como o ponto de referência da peça j só pode estar contido em uma fatia, visto que as fatias são regiões disjuntas, também é necessário acrescentar um coeficiente *big-M* e uma variável binária ξ_{ij}^k , definida abaixo:

$$\xi_{ij}^k = \begin{cases} 1, & \text{se o ponto de referência } v_j = (x_j, y_j) \text{ está contido na fatia } S_{ij}^k; \\ 0, & \text{caso contrário.} \end{cases} \quad (3.4)$$

Assim, as restrições de não sobreposição podem ser escritas como:

$$\alpha_{ij}^{kf}(x_j - x_i) + \beta_{ij}^{kf}(y_j - y_i) \leq \gamma_{ij}^{kf} + M(1 - \xi_{ij}^k), \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.5)$$

$$k = 1, \dots, m_{ij}, f = 1, \dots, t_{ij}^k.$$

Além disso, é necessário garantir que o ponto de referência da peça j esteja contido em uma única fatia S_{ij}^k , isto é:

$$\sum_{k=1}^{m_{ij}} \xi_{ij}^k = 1, \quad \forall i, j \in \mathcal{P}, i < j. \quad (3.6)$$

Fischetti e Luzzi (2009) aplicam a técnica de *lifting* na restrição (3.5) para obter uma restrição mais apertada (3.7).

$$\alpha_{ij}^{kf}(x_j - x_i) + \beta_{ij}^{kf}(y_j - y_i) \leq \sum_{\ell=1}^{m_{ij}} \psi_{ij}^{kf\ell} \xi_{ij}^{\ell}, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.7)$$

$$k = 1, \dots, m_{ij}, f = 1, \dots, t_{ij}^k$$

Para calcular os coeficientes de *lifting*, é resolvido o subproblema (3.8).

$$\psi_{ij}^{kf\ell} = \max_{v_j - v_i \in S_{ij}^{\ell} \cap B} \alpha_{ij}^{kf}(x_j - x_i) + \beta_{ij}^{kf}(y_j - y_i), \quad (3.8)$$

em que B é uma caixa larga o suficiente para incluir todos os posicionamentos das peças i e j de largura $2W$ e comprimento $2L$ (L é um limitante superior para o comprimento da placa).

O modelo proposto por Fischetti e Luzzi (2009) é dado por:

$$\min z \quad (3.9)$$

$$\text{s.a: } x_i \leq z - \ell_i, \quad \forall i \in \mathcal{P}, \quad (3.10)$$

$$y_i \leq W - w_i, \quad \forall i \in \mathcal{P}, \quad (3.11)$$

$$\alpha_{ij}^{kf}(x_j - x_i) + \beta_{ij}^{kf}(y_j - y_i) \leq \sum_{\ell=1}^{m_{ij}} \psi_{ij}^{k\ell} \xi_{ij}^{\ell}, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.12)$$

$$k = 1, \dots, m_{ij}, f = 1, \dots, t_{ij}^k,$$

$$\sum_{k=1}^{m_{ij}} \xi_{ij}^k = 1, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.13)$$

$$\xi_{ij}^k \in \{0, 1\}, \quad \forall i, j \in \mathcal{P}, i < j, k = 1, \dots, m_{ij}, \quad (3.14)$$

$$x_i, y_i \geq 0, \quad \forall i \in \mathcal{P}. \quad (3.15)$$

3.2 Modelo de Álvarez-Valdés, Martínez e Tamarit (2013)

O modelo de Álvarez-Valdés, Martínez e Tamarit (2013) consiste na aplicação de técnicas de *lifting* nas restrições (3.10) e (3.11) do modelo de Fischetti e Luzzi (2009). Sejam $\overline{X}_{ij}$ e $\underline{X}_{ij}$, respectivamente, os valores máximo e mínimo da coordenada x de $NFP_{i,j}$ e $\overline{Y}_{ij}$ e $\underline{Y}_{ij}$, respectivamente, os valores máximo e mínimo da coordenada y de $NFP_{i,j}$. Além disso, sejam $\overline{x}_{ij}^k$ e $\underline{x}_{ij}^k$ os valores máximo e mínimo que a variável x_j pode assumir nas coordenadas de $NFP_{i,j}$ quando a variável ξ_{ij}^k está ativa. Do mesmo modo, sejam $\overline{y}_{ij}^k$ e $\underline{y}_{ij}^k$ os valores máximo e mínimo que a variável y_j pode assumir nas coordenadas de $NFP_{i,j}$ quando a variável ξ_{ij}^k está ativa. Logo, podemos definir os subconjuntos de variáveis associados ao $NFP_{i,j}$ que forcem, respectivamente, o posicionamento do ponto de referência da peça j acima (3.16), abaixo (3.17), ao lado direito (3.18) ou esquerdo (3.19) do ponto de referência da peça i :

$$U_{ij} = \{\xi_{ij}^k \in V NFP_{i,j} \mid \underline{y}_{ij}^k \geq 0\}, \quad (3.16)$$

$$D_{ij} = \{\xi_{ij}^k \in V NFP_{i,j} \mid \overline{y}_{ij}^k \leq 0\}, \quad (3.17)$$

$$R_{ij} = \{\xi_{ij}^k \in V NFP_{i,j} \mid \underline{x}_{ij}^k \geq 0\}, \quad (3.18)$$

$$L_{ij} = \{\xi_{ij}^k \in V NFP_{i,j} \mid \overline{x}_{ij}^k \leq 0\}, \quad (3.19)$$

em que $V NFP_{i,j}$ é o conjunto de todas as variáveis ξ_{ij}^k associadas ao $NFP_{i,j}$.

Após a definição desses conjuntos, os autores propõem os seguintes limitantes dos

lados esquerdo (3.20), direito (3.21), inferior (3.22) e superior (3.23) para uma peça i :

$$x_i \geq \sum_{k \in R_{ij}} \underline{x}_{ij}^k \xi_{ij}^k, \quad (3.20)$$

$$x_i \leq z - \ell_i - \sum_{\substack{k \in L_{ij} \\ \underline{\lambda}_{ij}^k > 0}} \underline{\lambda}_{ij}^k \xi_{ij}^k, \quad (3.21)$$

$$y_i \geq \sum_{k \in U_{ij}} \underline{y}_{ij}^k \xi_{ij}^k, \quad (3.22)$$

$$y_i \leq W - w_i - \sum_{\substack{k \in D_{ij} \\ \underline{\mu}_{ij}^k > 0}} \underline{\mu}_{ij}^k \xi_{ij}^k, \quad (3.23)$$

em que $\underline{\lambda}_{ij}^k = \ell_i - (\bar{x}_{ij}^k - \underline{X}_{ij})$ e $\underline{\mu}_{ij}^k = w_i - (\bar{y}_{ij}^k - \underline{Y}_{ij})$.

O modelo proposto por [Álvarez-Valdés, Martínez e Tamarit \(2013\)](#) é dado por:

$$\min z \quad (3.24)$$

$$\text{s.a: } \sum_{k \in R_{ij}} \underline{x}_{ij}^k \xi_{ij}^k \leq x_i \leq z - \ell_i - \sum_{\substack{k \in L_{ij} \\ \underline{\lambda}_{ij}^k > 0}} \underline{\lambda}_{ij}^k \xi_{ij}^k, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.25)$$

$$\sum_{k \in U_{ij}} \underline{y}_{ij}^k \xi_{ij}^k \leq y_i \leq W - w_i - \sum_{\substack{k \in D_{ij} \\ \underline{\mu}_{ij}^k > 0}} \underline{\mu}_{ij}^k \xi_{ij}^k, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.26)$$

$$\alpha_{ij}^{kf} (x_j - x_i) + \beta_{ij}^{kf} (y_j - y_i) \leq \sum_{\ell=1}^{m_{ij}} \psi_{ij}^{k\ell} \xi_{ij}^\ell, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.27)$$

$$k = 1, \dots, m_{ij}, f = 1, \dots, t_{ij}^k,$$

$$\sum_{k=1}^{m_{ij}} \xi_{ij}^k = 1, \quad \forall i, j \in \mathcal{P}, i < j, \quad (3.28)$$

$$\xi_{ij}^k \in \{0, 1\}, \quad \forall i, j \in \mathcal{P}, i < j, k = 1, \dots, m_{ij}, \quad (3.29)$$

$$x_i, y_i \geq 0, \quad \forall i \in \mathcal{P}. \quad (3.30)$$

3.3 Modelo de Toledo et al. (2013)

[Toledo et al. \(2013\)](#) propuseram um modelo inteiro misto para o problema de empacotamento em faixas de peças irregulares em que a placa é descrita como uma malha de pontos. Dada uma placa com largura fixa W e comprimento inicial L (limitante superior), o objetivo do problema é minimizar o comprimento utilizado z da placa. Assim, para obter a malha de pontos, é feita uma discretização horizontal e vertical da placa utilizando as granularidades g_x e g_y . Os autores ressaltam que esses valores influenciam diretamente na qualidade da solução e no número de variáveis e de restrições do problema.

A seguir são apresentados os índices utilizados no modelo:

- $t, u \in \mathcal{T} = \{1, \dots, T\}$ - são tipos de peça;

- $c \in \mathcal{C} = \{0, \dots, C - 1\}$ - é uma coluna da malha de pontos;
- $r \in \mathcal{R} = \{0, \dots, R - 1\}$ - é uma linha da malha de pontos;
- $d, e \in \mathcal{D} = \{1, \dots, D\}$ - são pontos da malha. Alternativamente, cada ponto também pode ser representado por um par ordenado de valores $(c, r) \in \mathcal{C} \times \mathcal{R}$, no qual $c = \lfloor \frac{d}{R} \rfloor$ e $r = d - \lfloor \frac{d}{R} \rfloor (R - 1)$. Analogamente, $d = c R - r + 1$.

Nesse modelo, cada tipo de peça t é representado por um polígono e possui uma quantidade de peças a serem cortadas q_t . O polígono é descrito por um conjunto de vértices cujas coordenadas são definidas a partir de um ponto de referência. Além disso, cada tipo de peça t possui um envelope retangular que define, respectivamente, limites inferiores x_t^m e y_t^m e superiores x_t^M e y_t^M para os eixos x e y , como ilustra a Figura 9.

Figura 9 – Representação de um tipo de peça.

Inicialmente, para garantir que cada peça esteja inteiramente contida na placa, é usado o conceito de *inner-fit polygon*. Dado um tipo de peça t e a placa, o *inner-fit polygon* $\mathcal{IFP}_t$ é o conjunto de pontos pertencentes à placa no qual o ponto de referência de uma peça do tipo t pode ser alocado sem que a peça extrapole os limites da placa (Figura 10).

Figura 10 – Exemplo de *inner-fit polygon* na malha de pontos.

Para garantir que não haja sobreposição entre duas peças é usado o conceito de *nofit polygon*. Dados dois tipos de peças t e u e um ponto d , o *nofit polygon* $\mathcal{NFP}_{t,u}^d$ é o conjunto de pontos que estão no interior do $\mathcal{NFP}_{t,u}$ se uma peça do tipo t for posicionada no ponto d (Figura 11). Observe que as peças se tocam nos pontos pretos e se sobrepõem nos pontos brancos.

Após a introdução dos conceitos de *inner-fit polygon* e *nofit polygon* para a malha de pontos, é necessário descrever as variáveis de decisão utilizadas no modelo. Para cada

Figura 11 – Construção do *nofit polygon* na malha de pontos.

tipo de peça t e um ponto da placa d , tal que $d \in \mathcal{IFP}_t$, temos uma variável binária de decisão δ_t^d , definida a seguir.

$$\delta_t^d = \begin{cases} 1, & \text{se uma peça do tipo } t \text{ está alocada ao ponto } d \text{ da placa;} \\ 0, & \text{caso contrário.} \end{cases} \quad (3.31)$$

O modelo proposto por [Toledo et al. \(2013\)](#) é dado por:

$$\min \quad z \quad (3.32)$$

$$\text{s.a:} \quad (c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.33)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t, \quad \forall t \in \mathcal{T}, \quad (3.34)$$

$$\delta_t^d + \delta_u^e \leq 1, \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t, \forall e \in \mathcal{NFP}_{t,u}^d, \quad (3.35)$$

$$\delta_t^d \in \{0, 1\}, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.36)$$

$$z \geq 0, \quad (3.37)$$

A função objetivo (3.32) em conjunto com as restrições (3.33) minimiza o comprimento total da placa necessário para cortar todas as peças. As restrições (3.34) garantem que todas as peças serão alocadas completamente na placa, enquanto que as restrições (3.35) impedem a sobreposição das peças. As restrições (3.36) e (3.37) definem, respectivamente, o domínio das variáveis de decisão e da variável associada ao comprimento da placa.

4 Modelos reduzidos

No modelo de [Toledo et al. \(2013\)](#), a matriz de restrições do problema possui uma grande quantidade de linhas devido às restrições de não-sobreposição das peças (3.35), que é no pior caso $\mathcal{O}(T^2D^2)$, em que T é o número de tipo de peças distintas e D é o número de pontos da malha utilizada na discretização da placa. Assim, buscamos substituir as restrições de não-sobreposição por um conjunto equivalente de menor dimensão, o que reduz a ordem da matriz de restrições do problema e pode levar à solução do problema de forma mais eficiente. Neste sentido, são propostos neste capítulo dois modelos reduzidos (Seções 4.1 e 4.2) baseados no modelo de [Toledo et al. \(2013\)](#). Na Seção 4.3, é descrita a estimativa do valor de alguns coeficientes utilizados nos modelos reduzidos. Na Seção 4.4, são discutidos os experimentos computacionais realizados para avaliar os modelos. Na Seção 4.5, são apresentadas algumas considerações sobre os modelos propostos.

4.1 Modelo reduzido I

O modelo reduzido I evita a sobreposição das peças considerando simultaneamente todos os pontos de $\mathcal{NFP}_{t,u}^d$ para um determinado ponto d e um par de tipos de peças t e u (restrições (4.1)), ou seja, faz-se um somatório em $\mathcal{NFP}_{t,u}^d$ das restrições:

$$\delta_t^d + \delta_u^e \leq 1, \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t, \forall e \in \mathcal{NFP}_{t,u}^d, \quad (3.35)$$

para obter as restrições compactas de não-sobreposição dadas por:

$$\sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq M(1 - \delta_t^d) \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t. \quad (4.1)$$

Observe que apesar do número de restrições ser significativamente menor, no pior caso $\mathcal{O}(T^2D)$, é necessário estimar um valor adequado para M , pois esse valor pode afetar a qualidade do limitante obtido através da relaxação linear. Nas restrições (4.1), quando a peça t não é alocada ao ponto d tem-se que $\delta_t^d = 0$, logo, o valor de M deve ser suficientemente grande para permitir que o maior número possível de peças do tipo u sejam alocadas aos pontos pertencentes ao conjunto $\mathcal{NFP}_{t,u}^d$.

O modelo reduzido I é dado por:

$$\min \quad z \quad (3.32)$$

$$\text{s.a:} \quad (c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.33)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t, \quad \forall t \in \mathcal{T}, \quad (3.34)$$

$$\sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq M(1 - \delta_t^d), \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t, \quad (4.1)$$

$$\delta_t^d \in \{0, 1\}, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.36)$$

$$z \geq 0. \quad (3.37)$$

4.2 Modelo reduzido II

O modelo reduzido II é baseado no modelo reduzido I, mas evita a sobreposição das peças considerando simultaneamente todos os pontos do $\mathcal{NFP}_{t,u}^d$ de todas as peças do tipo u tais que $u \geq t$ para um determinado ponto d e um tipo de peça t . Neste caso, tomamos a somatória das restrições (4.1) para todos os tipos de peças u tais que $u \geq t$, ou seja,

$$\sum_{\substack{u \in \mathcal{T} \\ u \geq t}} \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq M(1 - \delta_t^d), \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t. \quad (4.2)$$

Assim como no modelo reduzido I, o número de restrições é significativamente menor, no pior caso $\mathcal{O}(TD)$, porém, mais uma vez, é necessário estimar um valor adequado para M .

O modelo reduzido II é dado por:

$$\min \quad z \quad (3.32)$$

$$\text{s.a:} \quad (c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.33)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t, \quad \forall t \in \mathcal{T}, \quad (3.34)$$

$$\sum_{\substack{u \in \mathcal{T} \\ u \geq t}} \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq M(1 - \delta_t^d), \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (4.2)$$

$$\delta_t^d \in \{0, 1\}, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.36)$$

$$z \geq 0. \quad (3.37)$$

Este modelo é mais compacto que o anterior, logo de menor dimensão. No entanto, seu limitante pode ser pior, uma vez que o valor de M tende a ser maior.

4.3 Estimativa para o valor de M

Para estimar o valor de M do modelo reduzido I (restrições (4.1)), podemos utilizar um limitante trivial, que é o valor da demanda das peças do tipo u , isto é:

$$M_{t,u}^d = q_u, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t. \quad (4.3)$$

Alternativamente, para determinar o valor de $M_{t,u}^d$ podemos resolver um problema de empacotamento bidimensional com apenas peças do tipo u . O objetivo desse problema é maximizar o número de peças do tipo u que podem ser posicionadas sobre os pontos de $\mathcal{NFP}_{t,u}^d$ sem sobreposição, caso uma peça do tipo t não seja alocada ao ponto d . Por exemplo, dados um quadrado (peça do tipo t) e um triângulo (peça do tipo u), a Figura 12 (a) ilustra o número máximo de peças do tipo u que podem ser dispostas nos pontos de $\mathcal{NFP}_{t,u}^d$ sem sobreposição. Analogamente, a Figura 12 (b) representa o número máximo de peças do tipo t que podem ser dispostas em $\mathcal{NFP}_{u,t}^e$.

Figura 12 – Problemas de empacotamento bidimensionais para um tipo de peça.

Suponha que para todo par de pontos $d \in \mathcal{IFP}_t$ e $e \in \mathcal{IFP}_u$ a seguinte desigualdade seja satisfeita: $M_{u,t}^e \leq M_{t,u}^d$. Como ambos os conjuntos de *nofit polygons* garantem que as peças t e u não se sobreponham, é melhor escolher o conjunto de restrições associado ao $\mathcal{NFP}_{u,t}^e$ (Figura 12 (b)), que fornece um limite mais apertado para as restrições (4.1).

Para determinar o valor de $M_{t,u}^d$, é necessário resolver o seguinte problema:

$$M_{t,u}^d = \max \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \quad (4.4)$$

$$\text{s.a:} \quad \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq q_u, \quad (4.5)$$

$$\delta_u^e + \delta_u^f \leq 1, \quad \forall e \in \mathcal{NFP}_{t,u}^d, \forall f \in \mathcal{NFP}_{t,u}^d \cap \mathcal{NFP}_{u,u}^e, e < f, \quad (4.6)$$

$$\delta_u^e \in \{0, 1\}, \quad \forall e \in \mathcal{NFP}_{t,u}^d. \quad (4.7)$$

A função objetivo (4.4) maximiza o número de peças empacotadas em $\mathcal{NFP}_{t,u}^d$. A restrição (4.5) define um limitante superior baseado na demanda das peças do tipo u . As restrições (4.6) impedem a sobreposição das peças para cada par de pontos $e, f \in \mathcal{NFP}_{t,u}^d$ tal que $e < f$, ou seja, é adicionada uma restrição de não-sobreposição se o ponto f pertencer ao $\mathcal{NFP}_{u,u}^e$. As restrições (4.7) definem o domínio das variáveis de decisão.

Analogamente, para o modelo reduzido II um valor trivial para M é o somatório do valor das demandas das peças do tipo u tais que $u \geq t$, que é dado por:

$$M_t^d = \sum_{\substack{u \in \mathcal{T} \\ u \geq t}} q_u, \quad \forall d \in IFP_t. \quad (4.8)$$

No caso do modelo reduzido II, também é possível estimar o valor de M_t^d das restrições (4.2), utilizando a ideia de empacotamento das peças, ou seja:

$$M_t^d = \sum_{\substack{u \in \mathcal{T} \\ u \geq t}} M_{t,u}^d. \quad (4.9)$$

No entanto, o valor obtido pela equação (4.9) não considera a influência da sobreposição de peças de tipos distintos u e v nos pontos de interseção dos seus conjuntos $\mathcal{NFP}_{t,u}^d$ e $\mathcal{NFP}_{t,v}^d$. Por exemplo, dados dois triângulos (tipos de peça t e u) e um quadrado (tipo de peça v), a Figura 13 ilustra a região de sobreposição entre os $\mathcal{NFP}_{t,u}^d$ e $\mathcal{NFP}_{t,v}^d$. Logo, é possível resolver outro problema de empacotamento bidimensional considerando mais de um tipo de peça simultaneamente. Esse problema é dado por:

Figura 13 – Problema de empacotamento para dois tipos de peça.

$$M_t^d = \max \sum_{\substack{u \in \mathcal{T} \\ u \geq t}} \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \quad (4.10)$$

$$\text{s.a:} \quad \sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq q_u, \quad \forall u \in \mathcal{T}, u \geq t, \quad (4.11)$$

$$\delta_u^e + \delta_v^f \leq 1, \quad \forall u, v \in \mathcal{T}, t \leq u \leq v, \forall e \in \mathcal{NFP}_{t,u}^d, \quad (4.12)$$

$$\delta_u^e \in \{0, 1\}, \quad \forall u \in \mathcal{T}, u \geq t, \forall e \in \mathcal{NFP}_{t,u}^d, \quad (4.13)$$

em que $(u, e) < (v, f)$ se $u < v$ ou $(u = v \text{ e } e < f)$.

A função objetivo (4.10) maximiza o número de peças empacotadas em $\mathcal{NFP}_{t,u}^d$ para qualquer tipo de peça u tal que $u \geq t$. As restrições (4.11) limitam a quantidade de peças empacotadas de acordo com as demandas de cada tipo de peça. As restrições (4.12) impedem a sobreposição das peças conforme descrito a seguir. Para cada par de pontos $e \in \mathcal{NFP}_{t,u}^d$ e $f \in \mathcal{NFP}_{t,v}^d$ tal que $(u, e) < (v, f)$, é adicionada uma restrição de não-sobreposição se o ponto $f \in \mathcal{NFP}_{u,v}^e$. As restrições (4.13) definem o domínio das variáveis de decisão.

Neste trabalho, a estimativa para o valor de M foi feita apenas para um ponto d de uma placa artificial onde todos os pontos dos *nofit polygons* estão inteiramente contidos, uma vez que é impraticável calcular esses valores para todos os pontos $d \in \mathcal{IFP}_t$. É importante destacar que esses valores devem ser determinados em uma etapa de pré-processamento.

4.4 Experimentos computacionais

Os experimentos computacionais foram realizados em um computador com processador Intel Core i7-2600 de 3.4 GHz e 16 GB de memória RAM, utilizando o sistema operacional Ubuntu 13.10 64 bits. Os modelos foram escritos na linguagem C++ (Concert) e resolvidos utilizando o *software* de otimização CPLEX versão 12.6. Para avaliar os modelos propostos, foram utilizadas as mesmas instâncias avaliadas em Toledo et al. (2013), ou seja, os grupos de instâncias RCO (RIBEIRO; CARRAVILLA; OLIVEIRA, 1999), BLAZEWICZ (BLAZEWICZ; HAWRYLUK; WALKOWIAK, 1993) e SHAPES (OLIVEIRA; FERREIRA, 1993). O tempo limite de execução foi de 1.800 segundos e a discretização da malha utilizada foi $g_x = g_y = 1$.

O grupo de instâncias RCO consiste em um conjunto de sete tipos de peças convexas (Figura 14), enquanto o grupo de instâncias BLAZEWICZ é formado por um conjunto de sete tipo de peças convexas e não-convexas (Figura 15). Note que as peças das instâncias RCO descrevem as envoltórias convexas das peças das instâncias BLAZEWICZ.

Figura 14 – Tipos de peça das instâncias RCO.

Figura 15 – Tipos de peça das instâncias BLAZEWCZ.

O grupo de instâncias SHAPES é formado por um conjunto de quatro tipo de peças convexas e não-convexas (Figura 16).

Figura 16 – Tipos de peça das instâncias SHAPES.

A Tabela 1 apresenta a demanda total de cada tipo de peça, a largura e o comprimento inicial da placa de cada instância.

O limitante inferior $\underline{z}$ é dado pelo máximo entre o maior comprimento das peças e o somatório da área de todas as peças dividido pela largura da placa, isto é:

$$\underline{z} = \max \left(\max_{t \in \mathcal{T}} (\ell_t), \frac{1}{W} \sum_{t \in \mathcal{T}} q_t A_t \right), \quad (4.14)$$

em que ℓ_t e A_t são, respectivamente, a largura e área de uma peça do tipo t .

Inicialmente, na etapa de préprocessamento dos modelos reduzidos I e II, foram resolvidos os problemas de empacotamento bidimensionais para cada tipo de peça (4.4) - (4.7) e para todos os pares de tipos de peças (4.10) - (4.13), respectivamente. O tempo total para resolver os problemas foi inferior a 20 segundos para qualquer uma das instâncias.

A Tabela 2 apresenta os resultados computacionais obtidos para os modelos propostos considerando duas estimativas para o valor de M (limitante trivial e problema de empacotamento bidimensional). A primeira coluna da tabela apresenta o nome da instância avaliada. Da segunda à quarta coluna são reportados para o modelo reduzido I

Tabela 1 – Características das instâncias.

Instância	Número de peças por tipo							Total de peças	Largura da placa	Comprimento inicial da placa
	1	2	3	4	5	6	7			
RCO1	1	1	1	1	1	1	1	7	15	8
RCO2	2	2	2	2	2	2	2	14	15	16
RCO3	3	3	3	3	3	3	3	21	15	24
RCO4	4	4	4	4	4	4	4	28	15	32
RCO5	5	5	5	5	5	5	5	35	15	40
BLAZEWCZ1	1	1	1	1	1	1	1	7	15	8
BLAZEWCZ2	2	2	2	2	2	2	2	14	15	16
BLAZEWCZ3	3	3	3	3	3	3	3	21	15	24
BLAZEWCZ4	4	4	4	4	4	4	4	28	15	32
BLAZEWCZ5	5	5	5	5	5	5	5	35	15	40
SHAPES2	2	2	2	2				8	40	14
SHAPES4	4	4	4	4				16	40	28
SHAPES5	5	5	5	5				20	40	35
SHAPES7	7	7	7	7				28	40	49
SHAPES9	9	7	9	9				34	40	63
SHAPES15	15	7	9	12				43	40	74

(limitante trivial), respectivamente, o valor da melhor solução obtida (z), o desvio máximo para a solução ótima ($GAP = 100(z - \underline{z})/z$) e o tempo para obter esta solução. As colunas seguintes (quinta à décima terceira) seguem o mesmo padrão. As duas últimas colunas descrevem, respectivamente, a média e o desvio padrão dos valores da função objetivo para cada instância.

Os modelos reduzidos apresentaram resultados semelhantes para a maioria das instâncias, tanto em tempo computacional quanto em qualidade da solução, conforme é destacado pelos valores da média e do desvio padrão da função objetivo. Para cada grupo de instâncias, os melhores valores das funções objetivo foram obtidos a partir de modelos distintos: SHAPES, reduzido I (limitante trivial); RCO, reduzido I (problema de empacotamento); e BLAZEWCZ, reduzido II (problema de empacotamento). Observe que a estimativa para o valor de M pode tanto melhorar quanto piorar o desempenho dos modelos propostos, devido às diversas características do *software* CPLEX, dentre os quais destacam-se: tempo de resolução e qualidade da relaxação linear, tempo e qualidade do preprocessamento, geração de planos de cortes, eficiência das heurísticas e quantidade de nós explorados. Neste sentido, a estimativa obtida pelo limitante trivial apresentou melhores resultados que o problema de empacotamento bidimensional para três instâncias para o modelo reduzido I (SHAPES5-9) e II (SHAPES7-9 e RCO4). Por outro lado, a mesma estimativa apresentou piores resultados para três instâncias para o modelo reduzido I (RCO4 e BLAZEWCZ4-5) e II (BLAZEWCZ2 e SHAPES7-9).

A seguir são comparados o modelo dos pontos e o modelo reduzido I utilizando o limitante trivial para o valor de M (Tabela 3), uma vez que este apresentou, de forma geral, os melhores resultados entre os modelos reduzidos. O modelo dos pontos obteve me-

Tabela 2 – Avaliação dos modelos propostos.

	Modelo reduzido I						Modelo reduzido II						μ_z	σ_z
	Limitante trivial			Problema de empacotamento			Limitante trivial			Problema de empacotamento				
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)		
RCO1	8	0,00	0,35	8	0,00	0,35	8	0,00	0,89	8	0,00	0,89	8,00	0,00
RCO2	15	16,00	TL	15	16,00	TL	15	16,00	TL	15	16,00	TL	15,00	0,00
RCO3	23	17,83	TL	23	17,83	TL	23	17,83	TL	23	17,83	TL	23,00	0,00
RCO4	32	21,25	ML	30	16,00	TL	30	16,00	TL	31	18,71	TL	30,75	0,96
RCO5	38	17,11	TL	38	17,11	TL	39	19,23	TL	39	19,23	TL	38,50	0,58
BLAZEWCZ1	8	0,00	0,36	8	0,00	0,36	8	0,00	0,88	8	0,00	0,88	8,00	0,00
BLAZEWCZ2	15	28,00	TL	15	28,00	TL	15	28,00	TL	14	22,86	TL	14,75	0,50
BLAZEWCZ3	22	26,36	TL	22	26,36	TL	22	26,36	TL	22	26,36	TL	22,00	0,00
BLAZEWCZ4	30	28,00	TL	29	25,52	TL	29	25,52	TL	29	25,52	TL	28,75	0,50
BLAZEWCZ5	37	27,03	TL	36	25,00	TL	36	25,00	TL	36	25,00	TL	36,25	0,50
SHAPES2	14	0,00	0,42	14	0,00	0,42	14	0,00	0,17	14	0,00	0,17	14,00	0,00
SHAPES4	26	38,46	TL	26	38,46	TL	27	40,74	TL	27	40,74	TL	26,50	0,58
SHAPES5	33	39,39	TL	35	42,86	TL	34	41,18	TL	34	41,18	TL	34,00	0,82
SHAPES7	45	37,78	TL	48	41,67	TL	45	37,78	TL	47	40,43	TL	46,25	1,50
SHAPES9	53	38,87	TL	59	45,08	TL	53	38,87	TL	59	45,08	TL	56,00	3,46
SHAPES15	-	-	TL	-	-	TL	-	-	TL	-	-	TL	-	-

ML - limite de memória atingido.

TL - limite de tempo atingido.

lhor desempenho para oito instâncias (RCO2-4 e BLAZEWCZ2-4) e o modelo reduzido I para sete (RCO1, BLAZEWCZ1 e SHAPES2-9). No entanto, o modelo reduzido I provou otimalidade apenas para três instâncias em que o número de repetições de cada peça é pequeno (RCO1, BLAZEWCZ1 e SHAPES2) enquanto que o modelo dos pontos conseguiu provar para sete (RCO1-3, BLAZEWCZ1-3 e SHAPES2).

Tabela 3 – Comparação entre o modelo dos pontos e o modelo reduzido I.

Instância	Modelo dos pontos			Modelo reduzido I		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,83	8	0,00	0,35
RCO2	15	0,00	2,80	15	16,00	TL
RCO3	22	0,00	154,39	23	17,83	TL
RCO4	29	3,45	TL	32	21,25	ML
RCO5	36	11,79	TL	38	17,11	TL
BLAZEWCZ1	8	0,00	0,80	8	0,00	0,36
BLAZEWCZ2	14	0,00	15,65	15	28,00	TL
BLAZEWCZ3	20	0,00	1321,01	22	26,36	TL
BLAZEWCZ4	28	22,61	TL	30	28,00	TL
BLAZEWCZ5	35	20,00	TL	37	27,03	TL
SHAPES2	14	0,00	1,00	14	0,00	0,42
SHAPES4	27	37,04	TL	26	38,46	TL
SHAPES5	35	42,86	TL	33	39,39	TL
SHAPES7	49	42,86	TL	45	37,78	TL
SHAPES9	63	48,57	TL	53	38,87	TL
SHAPES15	-	-	TL	-	-	TL

ML - limite de memória atingido.

TL - limite de tempo atingido.

Vale destacar que para a instância SHAPES9, os modelos reduzidos (limitante trivial) encontraram soluções melhores que as apresentadas pelo modelo dos pontos. Estas soluções utilizam uma placa de comprimento 53 (Figura 17), enquanto que em [Toledo et al. \(2013\)](#) foi obtida uma solução cujo valor da função objetivo é 54, com limite de tempo de execução de cinco horas.

Figura 17 – Soluções obtidas pelos modelos propostos para a instância SHAPES9. (a) Modelo reduzido I. (b) Modelo reduzido II.

4.5 Considerações

Neste capítulo foram propostos dois modelos reduzidos para o problema de empacotamento em faixas de peças irregulares. Esses modelos apresentaram um desempenho melhor que o modelo dos pontos para instâncias com poucas repetições de peças, uma vez que provaram a otimalidade em menor tempo computacional. Além disso, os modelos propostos conseguiram encontrar soluções cujo comprimento utilizado da placa é menor que o do modelo dos pontos para algumas instâncias, o que demonstra a sua relevância para a literatura. No entanto, ambos os modelos não conseguiram provar a otimalidade para instâncias com várias repetições de peças, devido à baixa qualidade das suas relaxações lineares. Neste sentido, no Capítulo 6 são propostos modelos baseados em coberturas por cliques, com o objetivo de reduzir o número de restrições e aumentar a qualidade da relaxação linear do modelo dos pontos. Para facilitar a apresentação desses modelos, uma breve revisão de problemas de cobertura de grafos por cliques é apresentada no Capítulo 5.

5 Problemas de cobertura de grafos por cliques

Neste capítulo, são apresentadas revisões sobre problemas de cobertura de grafos por cliques. Na Seção 5.1, são descritos alguns conceitos de grafos necessários para a compreensão deste trabalho. Na Seção 5.2, é apresentada uma revisão sobre o problema de cobertura de vértices por cliques, enquanto que na Seção 5.3 é discutido o problema de cobertura de arestas por cliques. Em resumo, este capítulo tem como objetivo apresentar os algoritmos que serão utilizados nos modelos propostos no Capítulo 6.

5.1 Conceitos de grafos

Seja um grafo simples não-dirigido $G = (V, E)$, em que V é o conjunto de seus vértices e $E \subseteq V \times V$ é o conjunto de suas arestas. Dizemos que dois vértices $i, j \in V$ são **adjacentes** se existe uma aresta $\{i, j\} \in E$. Neste sentido, a **vizinhança** de um vértice $i \in V$ é representada pelo conjunto de todos seus vértices adjacentes, isto é, $N(i) = \{j \in V \mid \{i, j\} \in E\}$.

O **grau** de um vértice $i \in V$, denotado por $\deg(i)$, representa o número de vértices adjacentes a i , ou seja, $\deg(i) = |N(i)|$. O **grau máximo** de um grafo G , denotado por $\Delta(G)$, representa o grau máximo de seus vértices. Analogamente, o **grau mínimo** de G , denotado por $\delta(G)$, é dado pelo grau mínimo de seus vértices.

O **grafo complementar** de $G = (V, E)$ é o grafo $\overline{G} = (V, \overline{E})$, cujo conjunto de arestas é dado por $\overline{E} = \{\{i, j\} \mid i, j \in V, i \neq j, \{i, j\} \notin E\}$. Um grafo G é dito **completo**, se todo par de vértices distintos são adjacentes, isto é, para quaisquer $i, j \in V$ tais que $i \neq j$, $\{i, j\} \in E$. $G' = (V', E')$ é um **subgrafo** de $G = (V, E)$, se $V' \subseteq V$ e $E' \subseteq E \cap V' \times V'$. Dado um subconjunto de vértices $S \subseteq V$, temos que o **subgrafo induzido** por S é dado por $G(S) = (S, E \cap S \times S)$.

Um **clique** C é um subconjunto de V tal que $G(C)$ é completo. Um **conjunto estável (independente)** S é um subconjunto de V cujos vértices não são adjacentes entre si, ou seja, $\forall i, j \in S, \{i, j\} \notin E$. Um clique é dito **maximal** se não for um subconjunto próprio de qualquer outro clique. Um clique é dito **máximo** se for um clique maximal de máxima cardinalidade de G . Analogamente, são definidos os conceitos de conjunto estável maximal e máximo.

O **número de estabilidade** de G que representa a cardinalidade do conjunto estável máximo é denotado por $\alpha(G)$, enquanto que a cardinalidade do clique máximo

é denotada por $\omega(G)$. Como um clique em G representa um conjunto estável no grafo complementar $\overline{G}$ e vice-versa, temos que $\alpha(G) = \omega(\overline{G})$ e $\alpha(\overline{G}) = \omega(G)$. Para mais detalhes sobre o problema do clique máximo, é sugerida a leitura das revisões descritas por [Pardalos e Xue \(1994\)](#), [Bomze et al. \(1999\)](#) e [Wu e Hao \(2015\)](#).

O **número cromático** $\chi(G)$ de um grafo G é o número mínimo de conjuntos estáveis (cores) utilizados para cobrir seus vértices, utilizando cores distintas para vértices adjacentes. Analogamente, o número mínimo de cliques necessários para cobrir um grafo G é denotado por $\overline{\chi}(G)$. Note que $\chi(G) = \overline{\chi}(\overline{G})$.

5.2 Cobertura de vértices

O problema de cobertura de vértices por cliques (*vertex clique covering problem*) consiste em encontrar o menor número de cliques que cobrem todos os vértices de um grafo G , ou seja, cada vértice deve ser coberto por pelo menos um clique. Quando cada vértice deve ser coberto por exatamente um clique, temos um problema de particionamento de vértices por cliques (*vertex clique partitioning problem*). Observe que este problema é equivalente ao problema de coloração do grafo complementar $\overline{G}$.

O problema de decisão associado à coloração de grafos é NP-completo ([KARP, 1972](#)). Dados um grafo $G = (V, E)$ e um número inteiro positivo k de cores disponíveis, existe uma função de coloração $\phi : V \rightarrow \mathbb{Z}_+^k$ tal que se $i, j \in V$ são adjacentes, então $\phi(i) \neq \phi(j)$?

Na literatura, o problema de coloração de grafos foi abordado por meio de métodos exatos em alguns trabalhos, por exemplo: [Brown \(1972\)](#), [Brélaç \(1979\)](#), [Mehrotra e Trick \(1996\)](#), [Méndez-Díaz e Zabala \(2006\)](#), [Méndez-Díaz e Zabala \(2008\)](#) e [Hansen, Labbé e Schindl \(2009\)](#). Uma das formulações clássicas para o problema é dada a seguir. Seja $\mathcal{K}$ o conjunto de cores disponíveis para colorir o grafo. Dados um vértice $i \in V$ e uma cor $k \in \mathcal{K}$, definimos as seguintes variáveis:

$$x_{ik} = \begin{cases} 1, & \text{se o vértice } i \text{ é colorido com a cor } k; \\ 0, & \text{caso contrário.} \end{cases} \quad (5.1)$$

$$y_k = \begin{cases} 1, & \text{se a cor } k \text{ é usada para colorir o grafo;} \\ 0, & \text{caso contrário.} \end{cases} \quad (5.2)$$

Se G for um grafo sem vértices isolados, um possível modelo de programação inteira é dado por:

$$\min \sum_{k=1}^n y_k \quad (5.3)$$

$$\text{s.a.: } \sum_{k=1}^n x_{ik} = 1, \quad \forall i \in V, \quad (5.4)$$

$$x_{ik} + x_{jk} \leq y_k, \quad \forall (i, j) \in E, \forall k \in \mathcal{K}, \quad (5.5)$$

$$x_{ik} \in \{0, 1\}, \quad \forall i \in V, \forall k \in \mathcal{K}, \quad (5.6)$$

$$y_k \in \{0, 1\}, \quad \forall k \in \mathcal{K}. \quad (5.7)$$

A função objetivo (5.3) minimiza o número de cores utilizadas. As restrições (5.4) garantem que cada vértice é colorido com uma única cor. As restrições (5.5) impedem que vértices adjacentes sejam coloridos com a mesma cor. As restrições (5.6) - (5.7) definem o domínio das variáveis de decisão.

Uma outra modelagem para o problema, baseada em conjuntos estáveis, foi proposta por [Mehrotra e Trick \(1996\)](#). Seja S o conjunto formado por todos os conjuntos estáveis de G . Cada conjunto estável $s \in S$ é representado por uma variável binária de decisão x_s , que possui o valor 1 se os vértices de s são coloridos com a mesma cor e 0 caso contrário. Assim, o modelo é dado por:

$$\min \sum_{s \in S} x_s \quad (5.8)$$

$$\text{s.a.: } \sum_{\substack{s \in S \\ i \in s}} x_s \geq 1, \quad \forall i \in V, \quad (5.9)$$

$$x_s \in \{0, 1\}, \quad s \in S. \quad (5.10)$$

A função objetivo (5.8) minimiza o número de conjuntos estáveis (cores) utilizados para cobrir os vértices do grafo. As restrições (5.9) garantem que cada vértice seja coberto por pelo menos um conjunto estável. As restrições (5.10) definem o domínio das variáveis de decisão. De forma geral, o número de conjuntos estáveis $|S|$ é exponencial. Neste sentido, os autores propuseram um método *branch-and-price* para o problema. O subproblema utilizado para a geração de colunas é o problema do conjunto estável de peso máximo.

Diversos autores desenvolveram heurísticas construtivas para o problema ([WELSH; POWELL, 1967](#); [BRÉLAZ, 1979](#); [LEIGHTON, 1979](#); [BOLLOBÁS; THOMASON, 1985](#); [CULBERSON; LUO, 1996](#)). Dentre elas, as mais populares são a DSATUR (abreviação do inglês *SATURation Degree*) proposta por [Brélaz \(1979\)](#) com tempo computacional $\mathcal{O}(n^2)$ e a *Recursive Largest First* (RLF) desenvolvida por [Leighton \(1979\)](#) com tempo $\mathcal{O}(n^3)$, em que n é o número de vértices de G . [Galinier e Hertz \(2006\)](#) realizaram uma revisão específica sobre métodos de otimização local para o problema. [Pardalos, Mavridou](#)

e Xue (1998) e Malaguti e Toth (2010) apresentaram duas revisões da literatura sobre o problema.

Neste trabalho, foi adotada a heurística RLF para a resolução do problema de coloração do grafo complementar (cobertura de vértices por cliques), uma vez que esta heurística obteve resultados melhores que a DSATUR nos testes realizados por Johnson et al. (1991). Além disso, utilizamos um algoritmo de geração de cliques maximais a partir de um conjunto de cliques. A seguir, descrevemos detalhadamente estes algoritmos.

5.2.1 Heurística RLF

A heurística RLF é um algoritmo construtivo proposto por Leighton (1979). Em cada passo do algoritmo, um nó é selecionado para ser colorido com o objetivo de reduzir o número de cores utilizadas para colorir os nós não-coloridos remanescentes. Neste algoritmo, as cores são atribuídas sequencialmente, ou seja, a cor $k + 1$ é utilizada somente após a coloração de todos os nós possíveis com a cor k .

A heurística RFL é descrita a seguir. Sejam $G = (V, E)$ um grafo e $U \subseteq V$ um subconjunto de vértices. Definimos $N_U(v)$ como a vizinhança de um vértice $v \in V$ no subgrafo induzido $G(U \cup \{v\})$. Analogamente, $\deg_U(v) = |N_U(v)|$ é o número de vértices adjacentes a v em $G(U \cup \{v\})$. Além disso, considere que na etapa na qual a cor k é usada, $U_1 \subseteq V$ representa o conjunto de vértices não-coloridos e não-adjacentes a nenhum vértice colorido e $U_2 \subseteq V$ o conjunto de vértices não-coloridos adjacentes a pelo menos um vértice colorido. Inicialmente, a primeira cor é utilizada pelo algoritmo para colorir o vértice com grau máximo em G ($U_1 = V$). A seguir, o algoritmo seleciona, caso seja possível, o vértice $v \in U_1$ com grau máximo em $G(U_2 \cup \{v\})$. Em caso de empate, é escolhido o vértice com grau mínimo em $G(U_1)$ para ser colorido. Após colorir todos os vértices possíveis em U_1 com a primeira cor, o algoritmo continua no subgrafo induzido $G(U_2)$, utilizando a próxima cor enquanto houver nós não-coloridos em G . O Algoritmo 1 descreve formalmente a heurística RLF.

Na Figura 18 é ilustrada a execução da heurística RLF. O grafo inicial é dado pela Figura 18 (a). Na primeira etapa é selecionado o vértice com grau máximo em G , ou seja, o vértice 1. Logo, $U_1 = \{6, 9\}$ e $U_2 = \{2, 3, 4, 5, 7, 8, 10, 11\}$. Além disso, é selecionado o vértice 6, pois ele possui grau máximo em U_2 conforme descrito na Figura 18 (b). Após a remoção dos vértices $\{1, 6\}$, é obtido o grafo da Figura 18 (c). Novamente, na segunda etapa, é selecionado o vértice com grau máximo neste novo grafo, isto é, o vértice 3. Assim, temos que $U_1 = \{7, 8, 10, 11\}$ e $U_2 = \{2, 4, 5, 9\}$. Apesar do empate entre os vértices 7 e 10 com relação a U_2 ($\deg_{U_2}(7) = \deg_{U_2}(10)$), o vértice 7 possui grau menor em U_1 , e, por isso, é selecionado. Logo, $U_1 = \{8, 11\}$ e $U_2 = \{2, 4, 5, 9, 10\}$. Como o vértice 8 possui grau máximo em U_2 , ele é selecionado. Após a remoção dos vértices $\{3, 7, 8\}$, é obtido o grafo da Figura 18 (d). O mesmo procedimento é feito até que todos os vértices sejam coloridos

Algoritmo 1: Heurística RLF**Entrada:** Grafo $G = (V, E)$.**Saída:** Função $\text{color} : V \rightarrow \mathbb{Z}_+$.

```

1   $U_1 \leftarrow V$ ;
2   $k \leftarrow 0$ ;
   ► Enquanto houver vértices não-coloridos
3  while  $U_1 \neq \emptyset$  do
4       $U_2 \leftarrow \emptyset$ ;
       ► Selecciona o vértice  $u \in U_1$  com grau máximo em  $G(U_1 \cup \{u\})$ 
5       $v \leftarrow \arg \max_{u \in U_1} \deg_{U_1}(u)$ ;
6       $\text{color}(v) \leftarrow k$ ;
7       $U_2 \leftarrow U_2 \cup N_{U_1}(v)$ ;
8       $U_1 \leftarrow U_1 \setminus (\{v\} \cup N_{U_1}(v))$ ;
9      while  $U_1 \neq \emptyset$  do
       ► Selecciona o vértice  $u \in U_1$  com grau máximo em  $G(U_2 \cup \{u\})$ , em caso de
         empate, selecciona o vértice com grau mínimo em  $G(U_1 \cup \{u\})$ 
10      $W \leftarrow \{w \mid w = \arg \max_{u \in U_1} \deg_{U_2}(u)\}$ ;
11      $v \leftarrow \arg \min_{u \in W} \deg_{U_1}(u)$ ;
12      $\text{color}(v) \leftarrow k$ ;
13      $U_2 \leftarrow U_2 \cup N_{U_1}(v)$ ;
14      $U_1 \leftarrow U_1 \setminus (\{v\} \cup N_{U_1}(v))$ ;
15     end
16      $U_1 \leftarrow U_2$ ;
17      $k \leftarrow k + 1$ ;
18 end
19 return  $\text{color}$ ;

```

(Figura 18 (e)), obtendo a seguinte coloração: $\{1, 6\}$, $\{3, 7, 8\}$, $\{5, 9, 11\}$, $\{2, 10\}$ e $\{4\}$.

5.2.2 Geração de cliques maximais

Dados um grafo $G = (V, E)$ e um conjunto de cliques Ω , o algoritmo de geração de cliques maximais consiste em incluir sequencialmente o maior número de vértices possíveis em cada clique, tornando-os maximais. O algoritmo procede conforme descrito a seguir. Dado um clique $C_k \in \Omega$, escolhe-se o vértice com menor grau $v \in C_k$ e explora-se sua vizinhança $N(v)$. Para cada vértice $i \in N(v) \setminus C_k$, verifica-se se o subgrafo induzido $G(C_k \cup \{i\})$ é completo. Em caso positivo, o vértice i é inserido no clique C_k . O Algoritmo 2 descreve formalmente a geração de cliques maximais. Note que o tempo de execução do algoritmo, no pior caso, é $\mathcal{O}(|\Omega| \Delta(G) \omega(G))$, em que $|\Omega|$ representa o número de cliques do conjunto Ω , $\Delta(G)$ o grau máximo de G e $\omega(G)$ a cardinalidade do clique máximo de G .

A geração de cliques maximais é ilustrada no grafo 1 (Figura 19). Seja o conjunto de cliques inicial $\Omega = \{\{1, 2\}, \{3, 4\}, \{3, 5, 6\}\}$. Após a execução do procedimento, é gerado

Figura 18 – Heurística RLF. Fonte: Adaptado de [LEIGHTON, 1979](#), p. 498-499.

Algoritmo 2: Geração de cliques maximais**Entrada:** Grafo $G = (V, E)$;Conjunto de cliques $\Omega = \{C_1, \dots, C_{|\Omega|}\}$.**Saída:** Conjunto de cliques maximais $\Omega' = \{C'_1, \dots, C'_{|\Omega'|}\}$.

```

1  for  $k \leftarrow 1$  to  $|\Omega|$  do
    ▶ Selecciona o vértice de  $C_k$  com grau mínimo
2   $v \leftarrow \arg \min_{i \in C_k} \deg(i)$ ;
    ▶ Adiciona mais vértices ao clique  $C_k$  até torná-lo maximal
3  foreach  $i \in N(v)$  do
4      if  $i \notin C_k$  then
        ▶ Verifica se o grafo  $G(C_k \cup \{i\})$  é completo
5         $complete \leftarrow true$ ;
6        foreach  $j \in C_k$  do
7            if  $\{i, j\} \notin E$  then
8                 $complete \leftarrow false$ ;
9                break;
10           end
11        end
12        if  $complete = true$  then
13             $C_k \leftarrow C_k \cup \{i\}$ ;
14        end
15    end
16 end
17 end
18 return  $\Omega' = \{C_1, \dots, C_{|\Omega|}\}$ ;

```

o conjunto de cliques maximais $\Omega' = \{\{1, 2, 5\}, \{3, 4, 6\}, \{2, 3, 5, 6\}\}$.

Figura 19 – Grafo 1.

5.3 Cobertura de arestas

O problema de cobertura de arestas por cliques (*edge clique covering problem*) consiste em encontrar o menor número de cliques que cobrem todas as arestas de um grafo G , ou seja, cada aresta deve ser coberta por pelo menos um clique. Quando cada

aresta deve ser coberta por exatamente um clique, temos um problema de particionamento de arestas por cliques (*edge clique partitioning problem*). O número mínimo de cliques utilizados para cobrir as arestas de um grafo (*clique covering number*) é denotado por $cc(G)$, enquanto que o mínimo para particioná-las (*clique partitioning number*) é denotado por $cp(G)$. Note que $cc(G) \leq cp(G)$.

O problema de decisão associado à cobertura de arestas por cliques é NP-completo (ORLIN, 1977; KOU; STOCKMEYER; WONG, 1978). Dados um grafo $G = (V, E)$ e um número inteiro positivo k , existem k cliques que cobrem todas as arestas de G ?

O problema de cobertura de arestas por cliques é equivalente ao problema de conflito de palavras chaves (*keyword conflict problem*) (KOU; STOCKMEYER; WONG, 1978) e ao problema do grafo de intersecções (*intersection graph problem*) (ERDŐS; GOODMAN; PÓSA, 1966). Duas revisões sobre o problema de cobertura de arestas por cliques podem ser consultadas em Pullman (1983), Monson, Pullman e Rees (1995). Para mais informações sobre as aplicações do problema, é sugerido o texto descrito por Roberts (1985).

O problema é resolvido em tempo polinomial para algumas classes de grafos, como os cordais (MA; WALLIS; WU, 1989), linhas (ORLIN, 1977), arco-circulares (HSU; TSAI, 1991) e com grau máximo quatro (PULLMAN, 1984) e cinco (HOOVER, 1992). Gramm et al. (2006 e 2008) propuseram regras de redução em tempo polinomial e métodos exatos para esse problema de cobertura de arestas por cliques.

Ao contrário do problema de cobertura de vértices por cliques, não há na literatura um grande número de heurísticas desenvolvidas para resolver o problema de cobertura de arestas. Kellerman (1973) propôs uma heurística construtiva para o problema, considerando a cobertura gradativa das arestas a partir de uma sequência de vértices. Esta heurística foi melhorada por Kou, Stockmeyer e Wong (1978), utilizando uma técnica de pós-processamento na qual são removidos cliques que cobrem arestas já cobertas pela união de outros cliques. Gramm et al. (2006) afirmaram que o tempo computacional das heurísticas propostas por Kellerman (1973) e Kou, Stockmeyer e Wong (1978) é $\mathcal{O}(nm^2)$, em que n e m são, respectivamente, o número de vértices e de arestas de G . Além disso, os autores propuseram modificações nestas heurísticas e o uso de estruturas de dados adicionais, reduzindo os tempos computacionais de ambas as heurísticas para $\mathcal{O}(nm)$.

Neste trabalho, por simplicidade, foi adotada a heurística de Kellerman (1973) (Subseção 5.3.1) em conjunto com outras técnicas de pós-processamento para a resolução do problema de cobertura de arestas por cliques. Após a execução desta heurística, obtemos cliques maximais (Subseção 5.2.2) e removemos cliques idênticos ou utilizamos a técnica proposta por Kou, Stockmeyer e Wong (1978) para eliminação de cliques redundantes (Subseção 5.3.2).

5.3.1 Heurística de Kellerman

Kellerman (1973) propôs uma heurística que constrói iterativamente uma cobertura de arestas por cliques. Em cada etapa i do algoritmo, o conjunto de cliques $\{C_1, \dots, C_k\}$ cobre todas as arestas do subgrafo induzido $G(V_i)$, em que $V_i = \{1, 2, \dots, i\}$. Inicialmente, dado um vértice i , é verificado o conjunto de vértices adjacentes com índice menor, ou seja, o conjunto $W = \{j \mid j < i, \{i, j\} \in E\}$. Caso W seja vazio, é criado um clique com apenas o vértice i e o algoritmo prossegue para o próximo vértice $i + 1$. Caso contrário, o algoritmo tenta adicionar o vértice i nos cliques existentes. Após esta verificação, pode haver arestas não cobertas em $G(V_i)$ entre o vértice i e os vértices de W . Para cobrir estas arestas, é criado um novo clique que contém o vértice i e seus vizinhos usando cliques existentes, com o objetivo de maximizar o número de arestas a serem cobertas. Esse processo é realizado até que todas as arestas em $G(V_i)$ sejam cobertas. O Algoritmo 3, adaptado de Gramm et al. (2006), descreve formalmente esta heurística.

Para ilustrar a execução desta heurística é utilizado o grafo 2 (Figura 20). Inicialmente, como $W^{(1)} = \emptyset$, é criado um clique $C_1 = \{1\}$. Analogamente, como $W^{(2)} = \emptyset$, é criado outro clique $C_2 = \{2\}$. A seguir, $W^{(3)} = \{1, 2\}$, e como ambos os cliques estão contidos em $W^{(3)}$, adiciona-se o vértice 3 aos cliques, ou seja, $C_1 = \{1, 3\}$ e $C_2 = \{2, 3\}$. Na quarta etapa, $W^{(4)} = \{2, 3\}$ e o clique C_2 está contido em $W^{(4)}$, logo $C_2 = \{2, 3, 4\}$ e não há arestas remanescentes. Na quinta etapa, $W^{(5)} = \{1, 3, 4\}$ e o clique C_1 está contido em $W^{(5)}$, logo $C_1 = \{1, 3, 5\}$, mas ainda é necessário cobrir a aresta $\{4, 5\}$. Assim, é criado um novo clique com cardinalidade de interseção máxima baseado no clique C_2 , ou seja, $C_3 = \{4\} \cup \{5\} = \{4, 5\}$. Na última etapa $W^{(6)} = \{3, 4, 5\}$ e o clique C_3 está contido em $W^{(6)}$, logo $C_3 = \{4, 5, 6\}$, mas ainda é necessário cobrir a aresta $\{3, 6\}$. Assim, é criado um novo clique com cardinalidade de interseção máxima baseado no clique C_1 , ou seja, $C_4 = \{3\} \cup \{6\} = \{3, 6\}$. Portanto, os cliques resultantes são $C_1 = \{1, 3, 5\}$, $C_2 = \{2, 3, 4\}$, $C_3 = \{4, 5, 6\}$ e $C_4 = \{3, 6\}$.

Figura 20 – Grafo 2. Fonte: KOU; STOCKMEYER; WONG, 1978, p. 138.

Algoritmo 3: Heurística de Kellerman**Entrada:** Grafo sem vértices isolados $G = (V, E)$, $V = \{1, \dots, n\}$.**Saída:** Cobertura de arestas por cliques $\mathcal{E} = \{C_1, \dots, C_{|\mathcal{E}|}\}$.

```

1  $k \leftarrow 0$ ;
2 for  $i \leftarrow 1$  to  $n$  do
3    $W \leftarrow \{j \mid j < i, \{i, j\} \in E\}$ ;
4   ► Caso a vizinhança de  $i$  em  $G(V_i)$  seja vazia, cria um clique com o vértice  $i$ 
5   if  $W = \emptyset$  then
6      $k \leftarrow k + 1$ ;
7      $C_k \leftarrow \{i\}$ ;
8   else
9     ► Verifica se é possível adicionar o vértice  $i$  aos cliques existentes
10     $U \leftarrow \emptyset$ ;
11    for  $\ell \leftarrow 1$  to  $k$  do
12      if  $C_\ell \subseteq W$  then
13         $U \leftarrow U \cup C_\ell$ ;
14         $C_\ell \leftarrow C_\ell \cup \{i\}$ ;
15        if  $U = W$  then
16          break;
17        end
18    end
19     $W \leftarrow W \setminus U$ ;
20    ► Para as arestas restantes, cria cliques com a máxima cardinalidade possível
21    baseados nos cliques existentes
22    while  $W \neq \emptyset$  do
23       $U = \{u \mid u = \arg \max_{1 \leq u \leq k} |C_u \cap W|\}$ ;
24       $\ell \leftarrow \min_{u \in U} u$ ;
25       $k \leftarrow k + 1$ ;
26       $C_k \leftarrow (C_\ell \cap W) \cup \{i\}$ ;
27       $W \leftarrow W \setminus C_k$ ;
28    end
29  end
30 return  $\mathcal{E} = \{C_1, \dots, C_k\}$ ;

```

5.3.2 Pós-processamento de Kou, Stockmeyer e Wong

Dado um conjunto de cobertura de arestas por cliques $\mathcal{E} = \{C_1, \dots, C_{|\mathcal{E}|}\}$, o pós-processamento proposto por [Kou, Stockmeyer e Wong \(1978\)](#) consiste em verificar se todas arestas cobertas por um clique $C_k \in \mathcal{E}$ são cobertas por pelo menos um outro clique $C_\ell \in \mathcal{E} \mid k \neq \ell$. Em caso positivo o clique C_k é considerado redundante e é removido. Um modo eficiente de verificar quais cliques são redundantes, consiste em contar quantos cliques cobrem cada aresta. Seja $\alpha_{\{i,j\}}$ o número de cliques que cobrem uma aresta $\{i, j\} \in E$.

Para cada clique $C_k \in \mathcal{E}$, verificamos se existe alguma aresta $\{i, j\} \in E, i, j \in C_k$ tal que $\alpha_{\{i, j\}} = 1$. Caso exista alguma aresta, o clique C_k não é redundante. Caso contrário, todas as arestas são cobertas por pelo menos um outro clique $C_\ell \in \mathcal{E}$, com $k \neq \ell$, e o clique C_k pode ser removido da cobertura de arestas. Para manter a consistência da contagem, os valores de $\alpha_{\{i, j\}}$ devem ser decrementados em uma unidade para todas as arestas cobertas por C_k . O Algoritmo 4 descreve o pós-processamento de [Kou, Stockmeyer e Wong \(1978\)](#), utilizando a contagem do número de cliques que cobrem cada aresta. Note que o tempo de execução do algoritmo, no pior caso, é $\mathcal{O}(|\mathcal{E}|\omega_{\mathcal{E}}^2)$, em que $|\mathcal{E}|$ representa o número de cliques da cobertura de arestas e $\omega_{\mathcal{E}}$ a cardinalidade do clique máximo de $\mathcal{E}$.

Algoritmo 4: Pós-processamento de Kou, Stockmeyer e Wong

Entrada: Grafo $G = (V, E), V = \{1, \dots, n\}$;
 Cobertura de arestas por cliques $\mathcal{E} = \{C_1, \dots, C_{|\mathcal{E}|}\}$.
Dados: Número de cliques de $\mathcal{E}$ incidentes em cada aresta $\{i, j\} : \alpha_{\{i, j\}}$.
Saída: Cobertura de arestas por cliques não redundantes $\mathcal{E}' = \{C'_1, \dots, C'_{|\mathcal{E}'|}\}$.

```

1  $\mathcal{E}' \leftarrow \emptyset$ ;
2 for  $k \leftarrow 1$  to  $|\mathcal{E}|$  do
    ▶ Verifica se o clique  $C_k$  é redundante
3    $redundant \leftarrow true$ ;
4   foreach  $\{i, j\} \mid i, j \in C_k, i < j$  do
5     if  $\alpha_{\{i, j\}} = 1$  then
6        $redundant \leftarrow false$ ;
7       break;
8     end
9   end
10  if  $redundant = true$  then
    ▶ Decrementa o valor do número de cliques incidentes nas arestas de  $G(C_k)$ 
11    foreach  $\{i, j\} \mid i, j \in C_k, i < j$  do
12       $\alpha_{\{i, j\}} \leftarrow \alpha_{\{i, j\}} - 1$ ;
13    end
14  else
15     $\mathcal{E}' \leftarrow \mathcal{E}' \cup \{C_k\}$ ;
16  end
17 end
18 return  $\mathcal{E}'$ ;

```

Para ilustrar a execução desta heurística é utilizado o grafo 3 (Figura 21). Inicialmente, são obtidos os cliques $C_1 = \{1, 2, 3\}, C_2 = \{1, 2, 4\}, C_3 = \{1, 3, 5\}$ e $C_4 = \{2, 3, 6\}$ pela heurística de [Kellerman \(1973\)](#). No entanto, as arestas cobertas por C_1 são um subconjunto das arestas cobertas pelos demais cliques, isto é, $C_1 \subseteq C_2 \cup C_3 \cup C_4$, e, consequentemente, C_1 é um clique redundante que pode ser removido. Portanto, os cliques resultantes são $C_1 = \{1, 2, 4\}, C_2 = \{1, 3, 5\}$ e $C_3 = \{2, 3, 6\}$.

Figura 21 – Grafo 3. Fonte: [KOU; STOCKMEYER; WONG, 1978](#), p. 138.

6 Modelo de cobertura por cliques

Neste capítulo é proposto um modelo de cobertura por cliques para o problema de empacotamento em faixas de peças irregulares. Na Seção 6.1, é apresentado o modelo de cobertura por cliques. Na Seção 6.2, são discutidos os experimentos computacionais realizados para avaliar o modelo. Na Seção 6.3, são apresentadas algumas considerações sobre o modelo proposto.

6.1 Modelo proposto

Para o modelo de Toledo et al. (2013), a sobreposição das peças pode ser descrita através de um grafo de conflitos $G = (V, E)$, em que V é o conjunto de vértices que representam as variáveis binárias (par ordenado formado por um tipo de peça e um ponto da placa) e E é o conjunto de arestas que descrevem as restrições de não-sobreposição entre as peças. Assim, para cada tipo de peça t e ponto $d \in \mathcal{IFP}_t$, temos que sua variável binária correspondente δ_t^d pode ser representada por um vértice $v = (d, t) \in V$. Além disso, dados dois vértices $v_1 = (d, t)$ e $v_2 = (e, u)$ tais que $e \in \mathcal{NFP}_{t,u}^d$, isto é, conjunto de pontos em que as peças se sobrepõem se uma peça do tipo t for posicionada no ponto $d \in \mathcal{IFP}_t$ e outra do tipo u for posicionada no ponto $e \in \mathcal{IFP}_u$, podemos descrever a não-sobreposição por uma aresta $\{v_1, v_2\}$.

A Figura 22 ilustra um exemplo de um problema de empacotamento em faixas de peças irregulares. Na Figura 22 (a) são apresentados dois tipos de peças e seus respectivos pontos de referências. Na Figura 22 (b) são indicados os pontos admissíveis de posicionamento na malha de pontos para pelo menos um tipo de peça, uma vez que $\mathcal{IFP}_1 = \{1, 2, 4, 5\}$ e $\mathcal{IFP}_2 = \{2, 3\}$. A Figura 22 (c) ilustra um padrão de corte factível para esse problema. Dados os *nofit polygons* entre os tipos de peças (Figura 23), podemos representar a sobreposição pelo grafo de conflitos da Figura 24, no qual vértices de mesma cor representam o mesmo tipo de peça. Assim, caso a demanda seja unitária para cada tipo de peça, uma solução factível para esse problema pode ser obtida a partir de dois vértices não-adjacentes de cores distintas no grafo. Logo, os vértices associados às variáveis $\{\delta_1^1, \delta_2^3\}$ descrevem uma solução factível para o problema (Figura 22 (c)).

Podemos generalizar o exemplo anterior para um problema de empacotamento em faixas de peças irregulares arbitrário. Assim, uma solução factível na malha de pontos pode ser obtida através de um conjunto estável que satisfaça as restrições de demanda (3.34). Neste sentido, é possível substituir as restrições de não-sobreposição (3.35) pelas restrições (6.2) que representam coberturas de arestas por cliques (*edge clique covering*).

Figura 22 – Exemplo de um problema de empacotamento em faixas de peças irregulares. (a) Tipos de peças. (b) Pontos admissíveis. (c) Solução factível.

Figura 23 – *No-fit polygons* na malha dos pontos. (a) $\mathcal{NFP}_{1,1}$. (b) $\mathcal{NFP}_{1,2}$. (c) $\mathcal{NFP}_{2,2}$.

Figura 24 – Exemplo de grafo de conflitos.

Para calcular a função objetivo, é necessário encontrar o maior coeficiente associado às variáveis binárias ativas. Ao invés de usar as restrições (3.33) para cada vértice, podemos usar coberturas de vértices por cliques (*vertex clique covering*), dadas pelas restrições (6.1).

Sejam $K \subseteq V$ um clique de G e $\mathcal{V}$ e $\mathcal{E}$, respectivamente, os conjuntos de cobertura

de vértices e arestas por cliques de G , então o modelo de cobertura por cliques é dado por:

$$\min \quad z \quad (3.32)$$

$$\text{s.a:} \quad \sum_{(d,t) \in K} (c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall K \in \mathcal{V}, \quad (6.1)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t, \quad \forall t \in \mathcal{T}, \quad (3.34)$$

$$\sum_{(d,t) \in K} \delta_t^d \leq 1, \quad \forall K \in \mathcal{E}, \quad (6.2)$$

$$\delta_t^d \in \{0, 1\}, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t. \quad (3.36)$$

Note que o modelo dos pontos pode ser considerado um caso particular do modelo de cobertura por cliques, quando são utilizadas as coberturas triviais para $\mathcal{V}$ e $\mathcal{E}$.

6.2 Experimentos computacionais

Os experimentos computacionais foram realizados em um computador com processador Intel Core i7-2600 de 3,4 GHz e 16 GB de memória RAM, utilizando o sistema operacional Ubuntu 14.04 LTS 64 bits. Os modelos de programação matemática foram escritos na linguagem C++ (Concert) e resolvidos utilizando o *software* de otimização CPLEX versão 12.6. Para avaliar os modelos propostos, foram utilizadas as mesmas instâncias utilizadas em Toledo et al. (2013), ou seja, os grupos de instâncias RCO (RIBEIRO; CARRAVILLA; OLIVEIRA, 1999), BLAZEWCZ (BŁAŻEWICZ; HAWRYLUK; WALKOWIAK, 1993) e SHAPES (OLIVEIRA; FERREIRA, 1993), utilizando a discretização da malha $g_x = g_y = 1$.

A Tabela 4 apresenta algumas características dos grafos das instâncias utilizadas, como o número total de peças, comprimento da placa, número de vértices e de arestas, graus mínimo, médio e máximo e densidade dos grafos. Observe que apesar das características serem semelhantes, as instâncias do grupo BLAZEWCZ apresentam um menor número de arestas, graus médio e máximo e densidade menores que as do grupo RCO.

Inicialmente, são descritos os métodos utilizados para gerar as restrições do modelo de cobertura por cliques. Na Subseção 6.2.1, é descrito o cálculo da cobertura de vértices por cliques utilizada nas restrições (6.1). Na Subseção 6.2.2, é descrito o cálculo da cobertura de arestas por cliques utilizada nas restrições (6.2). Uma vez escritos os modelos, os testes computacionais foram realizados em duas etapas. A primeira etapa tem por objetivo buscar a melhor combinação (dentre as estudadas) para gerar as restrições e a função objetivo dos modelos, ou seja, buscamos a melhor combinação das heurísticas descritas nas seções anteriores. Como são seis combinações diferentes, optamos por avaliar as instâncias considerando o tempo limite de 3.600 segundos (Subseção 6.2.3). Na segunda

Tabela 4 – Características dos grafos das instâncias.

Instância	Peças	Comprimento inicial da placa	Vértices	Arestas	Grau			Densidade
					min	med	max	
RCO1	7	8	432	36.049	22	166,89	274	0,387
RCO2	14	15	1.013	112.174	22	221,47	386	0,219
RCO3	21	22	1.594	188.299	22	236,26	386	0,148
RCO4	28	29	2.175	264.424	22	243,15	386	0,112
RCO5	35	37	2.839	351.424	22	247,57	386	0,087
BLAZEWICZ1	7	8	432	35.462	22	164,18	267	0,381
BLAZEWICZ2	14	15	1.013	110.145	22	217,46	371	0,215
BLAZEWICZ3	21	22	1.594	184.828	22	231,90	371	0,146
BLAZEWICZ4	28	28	2.092	248.842	22	237,90	371	0,114
BLAZEWICZ5	35	37	2.839	344.863	22	242,95	371	0,086
SHAPES2	8	14	578	42.945	52	148,60	263	0,258
SHAPES4	16	27	2.333	556.308	83	476,90	846	0,204
SHAPES5	20	32	3.008	796.063	83	529,30	964	0,176
SHAPES7	28	48	5.168	1.563.279	83	604,98	1,014	0,117
SHAPES9	34	54	5.978	1.850.985	83	619,27	1,014	0,104
SHAPES15	43	67	7.733	2.474.348	83	639,95	1,014	0,083

etapa, é realizada uma comparação entre os melhores modelos obtidos e o modelo dos pontos, utilizando o tempo limite de cinco horas (Subseção 6.2.4).

6.2.1 Cálculo da cobertura de vértices por cliques

Para obter as coberturas de vértices por cliques, utilizamos a heurística construtiva RLF proposta por [Leighton \(1979\)](#) e descrita na Subseção 5.2.1. No entanto, os cliques obtidos por esta heurística formam um particionamento e geralmente não são maximais. Neste sentido, a partir dos cliques obtidos pela heurística são gerados cliques maximais conforme descrito na Subseção 5.2.2. Denotaremos esta heurística por RLFM. A Tabela 5 descreve o número de cliques gerados, a cardinalidade mínima, média e máxima dos cliques e o tempo de execução de cada algoritmo. Observe que o tempo de geração de cliques maximais é inferior a 0,1 segundo para todas as instâncias, mas é responsável por aumentar significativamente as cardinalidades mínima, média e máxima das coberturas. Além disso, assim como o grau médio, a cardinalidade média aumenta gradativamente nas instâncias de um mesmo grupo (RCO, BLAZEWICZ e SHAPES), visto que a região central da placa, onde há inúmeras possibilidades de sobreposição, torna-se maior proporcionalmente aos extremos da placa, onde há poucas possibilidades de posicionamento das peças.

6.2.2 Cálculo da cobertura de arestas por cliques

Para obter as coberturas de arestas por cliques, utilizamos a heurística proposta por [Kellerman \(1973\)](#) descrita na Subseção 5.3.1, denotada por heurística K. A partir das soluções obtidas, usamos algumas técnicas de pós-processamento:

Tabela 5 – Cálculo das coberturas de vértices por cliques.

Instância	Heurística RLF					Heurística RLFM				
	Cliques	Cardinalidade			Tempo(s)	Cliques	Cardinalidade			Tempo(s)
		min	med	max			min	med	max	
RCO1	28	2	15,43	27	0,01	28	23	56,68	95	0,01
RCO2	49	4	20,67	28	0,08	49	23	69,65	100	0,08
RCO3	77	2	20,70	28	0,27	77	23	72,82	100	0,28
RCO4	98	4	22,19	28	0,66	98	23	75,47	100	0,67
RCO5	126	4	22,53	28	1,41	126	23	76,76	100	1,43
BLAZEWCZ1	28	2	15,43	26	0,01	28	23	52,21	81	0,01
BLAZEWCZ2	49	4	20,67	27	0,08	49	23	62,33	87	0,09
BLAZEWCZ3	77	2	20,70	27	0,28	77	23	64,90	87	0,29
BLAZEWCZ4	98	2	21,35	27	0,60	98	23	66,45	87	0,61
BLAZEWCZ5	126	4	22,53	27	1,42	126	23	68,33	87	1,44
SHAPES2	26	4	22,23	35	0,01	26	35	53,81	59	0,01
SHAPES4	50	9	46,66	68	0,41	50	59	109,82	164	0,42
SHAPES5	64	10	47,00	67	0,84	64	59	117,73	166	0,86
SHAPES7	100	5	51,68	74	3,66	100	59	129,79	165	3,71
SHAPES9	114	1	52,44	71	5,47	114	59	132,72	167	5,53
SHAPES15	142	5	54,46	81	11,53	142	59	135,18	165	11,63

- **heurística KK** - pós-processamento de [Kou, Stockmeyer e Wong \(1978\)](#), descrito na Subseção 5.3.2;
- **heurística KMR** - geração de cliques maximais e remoção de cliques idênticos;
- **heurística KMK** - geração de cliques maximais e pós-processamento de [Kou, Stockmeyer e Wong \(1978\)](#).

A Tabela 6 reporta o número de cliques e o tempo de execução das heurísticas de cobertura de arestas. Note que o tempo de todas as heurísticas é semelhante, uma vez que os algoritmos da fase de pós-processamento das heurísticas KK, KMR e KMK possuem menor complexidade computacional que a heurística K. Por outro lado, a heurística KMK apresentou resultados significativamente melhores que as outras heurísticas com relação ao número de cliques das coberturas para todas as instâncias avaliadas, com reduções superiores a 50% para algumas instâncias (SHAPES7-15). A heurística KMR apresentou um resultado intermediário, gerando mais cliques que a heurística KMK e menos que as demais heurísticas. As heurísticas K e KK apresentaram resultados muito semelhantes, com variação de um ou dois cliques em algumas instâncias. A Tabela 7 apresenta as cardinalidades mínima, média e máxima das heurísticas. Como esperado, as heurísticas KMR e KMK resultam em cardinalidades maiores que as heurísticas K e KK e, novamente, as heurísticas K e KK possuem resultados quase idênticos. Além disso, assim como o grau médio, a cardinalidade média aumenta gradativamente nas instâncias de um mesmo grupo (RCO, BLAZEWCZ e SHAPES).

Tabela 6 – Cálculo das coberturas de arestas por cliques.

Instância	Heurística K		Heurística KK		Heurística KMR		Heurística KMK	
	Cliques	Tempo(s)	Cliques	Tempo(s)	Cliques	Tempo(s)	Cliques	Tempo(s)
RCO1	331	0,13	331	0,13	254	0,15	160	0,17
RCO2	879	0,81	879	0,83	717	0,91	408	0,98
RCO3	1.397	1,75	1.396	1,78	1.125	1,93	667	2,04
RCO4	1.929	3,08	1.929	3,13	1.573	3,31	934	3,53
RCO5	2.501	4,99	2.501	5,06	2.062	5,42	1.231	5,61
BLAZEWICZ1	554	0,27	552	0,27	506	0,31	271	0,35
BLAZEWICZ2	1.436	1,65	1.435	1,67	1.309	1,80	694	1,89
BLAZEWICZ3	2.300	3,93	2.298	3,96	2.117	4,23	1.108	4,37
BLAZEWICZ4	3.029	6,61	3.027	6,65	2.796	7,06	1.470	7,23
BLAZEWICZ5	4.080	11,78	4.078	11,85	3.771	12,51	1.998	12,72
SHAPES2	911	0,42	911	0,42	902	0,48	543	0,51
SHAPES4	5.675	29,28	5.673	29,38	5.636	30,88	3.435	31,77
SHAPES5	8.240	59,87	8.240	60,06	8.089	62,71	4.280	64,13
SHAPES7	14.432	172,79	14.431	173,24	14.257	179,96	6.856	182,37
SHAPES9	16.557	225,52	16.556	226,08	16.383	234,15	7.878	236,78
SHAPES15	21.794	377,11	21.794	377,92	21.611	391,63	10.117	394,37

Tabela 7 – Cardinalidade das coberturas de arestas por cliques.

Instância	Heurística K			Heurística KK			Heurística KMR			Heurística KMK		
	min	med	max	min	med	max	min	med	max	min	med	max
RCO1	2	26,85	92	2	26,85	92	23	66,56	98	23	67,97	98
RCO2	2	33,27	100	2	33,27	100	23	76,41	100	23	78,07	100
RCO3	2	34,96	100	2	34,97	100	23	79,84	100	23	80,54	100
RCO4	2	35,73	100	2	35,73	100	23	80,31	100	23	82,08	100
RCO5	2	36,60	100	2	36,60	100	23	80,70	100	23	82,08	100
BLAZEWICZ1	2	23,96	79	2	23,89	79	23	65,75	86	23	63,60	86
BLAZEWICZ2	2	28,59	87	2	28,58	87	23	71,61	88	23	70,45	88
BLAZEWICZ3	2	29,52	87	2	29,51	87	23	72,89	88	23	72,01	88
BLAZEWICZ4	2	30,21	87	2	30,20	87	23	73,51	88	23	72,75	88
BLAZEWICZ5	2	31,01	87	2	31,01	87	23	74,02	88	23	73,27	88
SHAPES2	2	18,21	58	2	18,21	58	34	53,82	66	34	54,25	66
SHAPES4	2	33,96	158	2	33,94	158	44	119,76	164	44	118,95	162
SHAPES5	2	40,77	162	2	40,77	162	44	125,96	165	44	124,18	165
SHAPES7	2	47,77	162	2	47,77	162	49	135,23	167	49	132,43	165
SHAPES9	2	49,15	162	2	49,15	162	49	137,50	167	49	134,26	165
SHAPES15	2	50,82	162	2	50,82	162	44	139,70	168	44	136,83	168

6.2.3 Avaliação dos modelos de cobertura por cliques

Para avaliar os modelos de cobertura por cliques (CC) foram gerados seis modelos, utilizando diferentes combinações de heurísticas de cobertura de vértices (RLF e RLFM) e arestas (KK, KMR e KMK) por cliques: CC_RLF_KK, CC_RLF_KMR, CC_RLF_KMK, CC_RLFM_KK, CC_RLFM_KMR e CC_RLFM_KMK. A heurística K não foi utilizada devido à similaridade de resultados com a heurística KK. Nesta primeira etapa de testes, o tempo de resolução de cada instância foi limitado a 3600 segundos. A

Tabela 8 apresenta uma comparação dos modelos que utilizam a heurística RLF para cobertura de vértices. O modelo CC_RLF_KMK apresentou os melhores resultados em função objetivo e *gap* de otimalidade para todas as instâncias. Além disso, esse modelo provou a otimalidade em menor tempo para todas as instâncias, com exceção da instância RCO3.

Tabela 8 – Avaliação dos modelos propostos utilizando a heurística RLF.

	Modelo CC_RLF_KK			Modelo CC_RLF_KMR			Modelo CC_RLF_KMK		
	<i>z</i>	<i>GAP</i> (%)	Tempo(s)	<i>z</i>	<i>GAP</i> (%)	Tempo(s)	<i>z</i>	<i>GAP</i> (%)	Tempo(s)
RCO1	8	0,00	0,21	8	0,00	0,19	8	0,00	0,10
RCO2	15	0,00	1,80	15	0,00	1,55	15	0,00	0,52
RCO3	22	0,00	130,00	22	0,00	72,15	22	0,00	92,34
RCO4	29	3,45	TL	29	3,45	TL	29	0,00	3.445,00
RCO5	36	6,04	TL	36	6,29	TL	36	5,90	TL
BLAZEWICZ1	8	0,00	0,20	8	0,00	0,25	8	0,00	0,12
BLAZEWICZ2	14	0,00	10,15	14	0,00	15,33	14	0,00	6,05
BLAZEWICZ3	20	0,00	697,07	20	0,00	98,48	20	0,00	37,42
BLAZEWICZ4	28	14,77	TL	27	9,61	TL	27	7,56	TL
BLAZEWICZ5	35	8,57	TL	34	5,88	TL	34	5,88	TL
SHAPES2	14	0,00	0,22	14	0,00	1,00	14	0,00	0,18
SHAPES4	25	8,00	TL	26	19,23	TL	25	8,00	TL
SHAPES5	31	32,98	TL	32	35,24	TL	31	21,03	TL
SHAPES7	48	41,67	TL	-	-	TL	48	41,23	TL
SHAPES9	-	-	TL	-	-	TL	-	-	TL
SHAPES15	67	40,45	TL	67	40,45	TL	67	40,45	TL

TL - limite de tempo atingido.

A Tabela 9 apresenta uma comparação dos modelos que utilizam a heurística RLFM para cobertura de vértices. O modelo CC_RLFM_KMK apresentou os melhores resultados em função objetivo e *gap* de otimalidade para todas as instâncias exceto BLAZEWICZ4 e SHAPES7. Além disso, esse modelo provou a otimalidade em menor tempo para quatro instâncias (RCO1-2, BLAZEWICZ2-3). Por outro lado, o modelo CC_RLFM_KK foi o único dos modelos que obteve uma solução factível para a instância SHAPES7 em uma hora. Esse modelo também provou otimalidade em menor tempo para três instâncias (RCO3, BLAZEWICZ1 e SHAPES2).

6.2.4 Comparação com a literatura

Na segunda etapa de testes, o tempo limite de execução foi de 18.000 segundos, o mesmo utilizado por Toledo et al. (2013). A Tabela 10 apresenta os resultados obtidos pelo modelo dos pontos e pelos modelos CC_RLF_KMK e CC_RLFM_KMK, que apresentaram os melhores resultados na primeira etapa de testes. Os resultados reportados para o modelo dos pontos foram obtidos no mesmo ambiente computacional descrito nesta seção.

Tabela 9 – Avaliação dos modelos propostos utilizando a heurística RLFM.

	Modelo CC_RLFM_KK			Modelo CC_RLFM_KMR			Modelo CC_RLFM_KMK		
	<i>z</i>	<i>GAP</i> (%)	Tempo(s)	<i>z</i>	<i>GAP</i> (%)	Tempo(s)	<i>z</i>	<i>GAP</i> (%)	Tempo(s)
RCO1	8	0,00	0,20	8	0,00	0,20	8	0,00	0,12
RCO2	15	0,00	1,15	15	0,00	1,61	15	0,00	0,56
RCO3	22	0,00	59,85	22	0,00	72,19	22	0,00	63,19
RCO4	29	3,45	TL	29	3,45	TL	29	3,45	TL
RCO5	36	6,20	TL	36	6,11	TL	36	5,78	TL
BLAZEWICZ1	8	0,00	0,17	8	0,00	0,38	8	0,00	0,20
BLAZEWICZ2	14	0,00	4,38	14	0,00	11,34	14	0,00	3,64
BLAZEWICZ3	20	0,00	487,86	20	0,00	1.623,79	20	0,00	45,40
BLAZEWICZ4	28	13,11	TL	27	7,41	TL	27	8,63	TL
BLAZEWICZ5	35	8,57	TL	35	8,57	TL	34	5,88	TL
SHAPES2	14	0,00	0,08	14	0,00	0,65	14	0,00	0,62
SHAPES4	26	11,54	TL	25	8,00	TL	25	4,00	TL
SHAPES5	31	16,13	TL	31	25,36	TL	31	12,79	TL
SHAPES7	48	41,67	TL	-	-	TL	-	-	TL
SHAPES9	-	-	TL	-	-	TL	-	-	TL
SHAPES15	67	40,45	TL	67	40,45	TL	67	40,45	TL

TL - limite de tempo atingido.

Os modelos de coberturas por cliques apresentaram os melhores resultados para a maioria das instâncias, enquanto que o modelo dos pontos apresentou o menor *gap* de otimalidade apenas para a instância SHAPES15. O modelo CC_RLFM_KMK obteve as melhores soluções para todas as instâncias e os menores *gaps* de otimalidade para quase todas (com exceção da SHAPES15). Por outro lado, o modelo CC_RLF_KMK provou a otimalidade de algumas instâncias em menos tempo, por exemplo, RCO4 e BLAZEWICZ4. Observe que ambos os modelos de cobertura por cliques provaram a otimalidade das instâncias RCO4 e BLAZEWICZ4 e obtiveram diversos resultados melhores que os descritos por [Toledo et al. \(2013\)](#).

6.3 Considerações

Neste capítulo, foi proposto o modelo de cobertura por cliques para o problema de empacotamento em faixas de peças irregulares. Foram utilizadas as coberturas triviais do modelo dos pontos ([TOLEDO et al., 2013](#)) e coberturas de vértices baseadas na heurística RLF ([LEIGHTON, 1979](#)) e de arestas baseadas na heurística de [Kellerman \(1973\)](#). Além disso, foi proposta a heurística KMK para o problema de cobertura de arestas por cliques. Esta heurística apresentou resultados significativamente melhores que a heurística KK para todos os grafos avaliados neste trabalho, o que indica que o algoritmo proposto pode ser promissor para outros tipos de grafos.

Os modelos de cobertura por cliques que apresentaram os melhores resultados nos

Tabela 10 – Comparação dos resultados com a literatura.

	Modelo dos pontos			Modelo CC_RLF_KMK			Modelo CC_RLFM_KMK		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,80	8	0,00	0,10	8	0,00	0,12
RCO2	15	0,00	1,65	15	0,00	0,52	15	0,00	0,56
RCO3	22	0,00	79,24	22	0,00	92,34	22	0,00	63,19
RCO4	29	0,00	7.220,93	29	0,00	3.445,00	29	0,00	8.343,11
RCO5	36	5,56	TL	36	4,65	TL	36	2,94	TL
BLAZEWICZ1	8	0,00	0,89	8	0,00	0,12	8	0,00	0,20
BLAZEWICZ2	14	0,00	11,41	14	0,00	6,05	14	0,00	3,64
BLAZEWICZ3	20	0,00	9.069,09	20	0,00	37,42	20	0,00	45,40
BLAZEWICZ4	27	3,70	TL	27	0,00	10.478,17	27	0,00	13.791,11
BLAZEWICZ5	35	14,29	TL	34	5,88	TL	34	5,88	TL
SHAPES2	14	0,00	2,05	14	0,00	0,18	14	0,00	0,62
SHAPES4	26	23,08	TL	25	0,00	14.649,62	25	0,00	7.566,50
SHAPES5	31	25,81	TL	30	6,67	TL	30	6,67	TL
SHAPES7	48	41,63	TL	44	35,02	TL	43	20,93	TL
SHAPES9	54	39,96	TL	54	38,86	TL	51	31,10	TL
SHAPES15	67	40,42	TL	67	40,45	TL	67	40,45	TL

TL - limite de tempo atingido.

testes computacionais foram os modelos CC_RLFM_KMK e CC_RLF_KMK. Ambos os modelos obtiveram um desempenho igual ou superior ao modelo dos pontos para todas as instâncias avaliadas.

No capítulo a seguir serão discutidos aspectos complementares dos modelos dos pontos e de cobertura por cliques, como a adição de desigualdades válidas, o cálculo de um limitante inferior e a disponibilização de uma solução inicial.

7 Desigualdades válidas, limitante inferior e solução inicial

Neste capítulo são discutidos aspectos complementares dos modelos dos pontos e de cobertura por cliques. Na Seção 7.1, são propostas novas desigualdades válidas e um novo limitante inferior para os modelos dos pontos e de cobertura por cliques. Na Seção 7.2, são discutidos os experimentos computacionais realizados para avaliar o modelo. Na Seção 7.3, são apresentadas algumas considerações sobre os tópicos abordados neste capítulo.

7.1 Desigualdades válidas e limitante inferior

No modelo dos pontos, além das desigualdades válidas clássicas do politopo do conjunto estável (REBENNACK; REINELT; PARDALOS, 2012), podemos considerar simultaneamente as restrições:

$$(c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t, \quad (3.33)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t, \quad \forall t \in \mathcal{T}, \quad (3.34)$$

para gerar as seguintes desigualdades válidas:

$$\sum_{d \in \mathcal{IFP}_t} (c_d g_x + x_t^M) \delta_t^d \leq q_t \max_{d \in \mathcal{IFP}_t} \{(c_d g_x + x_t^M) \delta_t^d\} \leq q_t z, \quad \forall t \in \mathcal{T} \quad (7.1)$$

Seja $\underline{z}$ um limitante inferior para o problema. Então, podemos reescrever as desigualdades válidas anteriores, como:

$$\sum_{d \in \mathcal{IFP}_t} \max\{\underline{z}, (c_d g_x + x_t^M)\} \delta_t^d \leq q_t \underline{z}, \quad \forall t \in \mathcal{T}. \quad (7.2)$$

Observe que as desigualdades válidas (7.2) agregam várias informações, como a demanda, os coeficientes associados à função objetivo e o valor de um limitante inferior do problema, o que pode melhorar o valor obtido pela relaxação linear dos modelos, mas pode tornar sua resolução mais difícil.

Analogamente, podemos obter desigualdades válidas para qualquer clique obtido do grafo de conflitos do modelo de cobertura por cliques, inclusive para os cliques triviais do modelo dos pontos:

$$\sum_{(d,t) \in K} (c_d g_x + x_t^M) \delta_t^d \leq z, \quad \forall K \in \mathcal{E}, \quad (7.3)$$

$$\sum_{(d,t) \in K} \max\{\underline{z}, (c_d g_x + x_t^M)\} \delta_t^d \leq z, \quad \forall K \in \mathcal{E}, \quad (7.4)$$

$$\sum_{(d,t) \in K} \max\{\underline{z}, (c_d g_x + x_t^M)\} \delta_t^d \leq z, \quad \forall K \in \mathcal{V}. \quad (7.5)$$

Note que as restrições (7.5) podem substituir as restrições (6.1) no modelo de cobertura por cliques. Como o modelo dos pontos é um caso particular do modelo de cobertura por cliques, estas substituições são equivalentes a alterar o valor do coeficiente associado à variável binária de cada uma das restrições (3.33) para o valor máximo entre o limitante inferior e o coeficiente antigo.

Além disso, é importante ressaltar que as desigualdades válidas (7.2), (7.4) e (7.5) podem ser utilizadas em um método iterativo para melhorar o limitante inferior do problema (Algoritmo 5). Em cada iteração, resolvemos a relaxação linear do modelo de cobertura por cliques com o acréscimo das desigualdades válidas e, caso um critério de parada não tenha sido atingido, são geradas novas desigualdades válidas para o problema a partir do valor do novo limitante inferior. Neste trabalho, foram utilizados os seguintes critérios: número máximo de iterações e alteração mínima no valor da função objetivo.

Algoritmo 5: Método iterativo para calcular um limitante inferior.

Entrada: Modelo de cobertura por cliques $\mathcal{M}$;

Limitante inferior $\underline{z}$.

Saída: Novo limitante inferior $\underline{z}'$.

```

1 while critério de parada não for satisfeito do
2   Adicionar desigualdades válidas (7.2), (7.4) e/ou (7.5) ao modelo  $\mathcal{M}$ ;
3    $\underline{z}' \leftarrow$  relaxação linear do modelo  $\mathcal{M}$ ;
4   ► Atualização do valor do limitante inferior
5   if  $\underline{z} < \underline{z}'$  then
6      $\underline{z} \leftarrow \underline{z}'$ ;
7   end
8 return  $\underline{z}'$ ;
```

7.2 Experimentos computacionais

Os experimentos descritos a seguir foram realizados no mesmo ambiente computacional descrito na Seção 6.2, utilizando o tempo limite de execução de 18.000 segundos. Os modelos comparados nestes experimentos são o modelo dos pontos e o modelo de cobertura por cliques CC_RLFM_KMK, uma vez que este apresentou, de forma geral, os melhores resultados no Capítulo 6.

Na Subseção 7.2.1, são avaliadas as desigualdades válidas descritas neste capítulo. Na Subseção 7.2.2, são apresentados os limitantes inferiores obtidos pelo processo iterativo proposto. Na Subseção 7.2.3, são realizados os testes finais com o valor do novo limitante inferior e com a disponibilização de uma solução inicial.

7.2.1 Avaliação das desigualdades válidas

Nesta seção são avaliadas as desigualdades válidas (7.2) e (7.5), utilizando o limitante inferior trivial (Equação 4.14). As desigualdades válidas (7.4) não são avaliadas devido ao grande aumento no número de restrições. O tempo limite de execução utilizado para todos os testes foi de 18.000 segundos.

Na Tabela 11, são reportados os resultados obtidos quando são adicionadas as desigualdades válidas (7.2) e (7.5) ao modelo dos pontos. Ao adicionar apenas as desigualdades válidas (7.5), o modelo dos pontos apresentou resultados muito semelhantes para quase todas as instâncias, reduzindo o *gap* de otimalidade para a instância SHAPES7 e aumentando para a SHAPES5. Por outro lado, a adição das desigualdades (7.2) e (7.5) proporcionou alguns resultados melhores, com destaque para as instâncias BLAZEWCZ3 e SHAPES4/7/15. No entanto, nesta configuração não foi encontrada nenhuma solução factível para a instância SHAPES9.

Na Tabela 12, são reportados os resultados obtidos quando são adicionadas as desigualdades válidas (7.2) e (7.5) ao modelo de cobertura por cliques. Para esse modelo, a adição das desigualdades válidas (7.5) apresentou bons resultados, uma vez que houve alterações significativas no tempo necessário para provar a otimalidade para algumas instâncias (RCO4 e SHAPES4) e foram obtidas soluções melhores para as instâncias SHAPES5/9. Ao contrário das desigualdades (7.5), a adição das desigualdades (7.2) apresentou resultados piores para quase todas as instâncias, aumentando o tempo computacional para provar a otimalidade, bem como a porcentagem do *gap* de otimalidade. Apesar disso, esta configuração obteve uma solução melhor para a instância SHAPES7, com comprimento de placa 42.

Na Tabela 13, são comparados os modelos dos pontos e de cobertura por cliques com adição das desigualdades válidas (7.5), uma vez que estas configurações apresentaram os melhores resultados nos testes anteriores. O modelo de cobertura por cliques obteve as

Tabela 11 – Avaliação das desigualdades válidas para o modelo dos pontos.

	Sem desigualdades			Desigualdades (7.5)			Desigualdades (7.2) e (7.5)		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,80	8	0,00	0,80	8	0,00	0,76
RCO2	15	0,00	1,65	15	0,00	1,67	15	0,00	2,56
RCO3	22	0,00	79,24	22	0,00	76,30	22	0,00	73,51
RCO4	29	0,00	7.220,93	29	0,00	7.185,35	29	3,45	TL
RCO5	36	5,56	TL	36	5,56	TL	36	10,69	TL
BLAZEWICZ1	8	0,00	0,89	8	0,00	0,91	8	0,00	1,00
BLAZEWICZ2	14	0,00	11,41	14	0,00	11,95	14	0,00	12,75
BLAZEWICZ3	20	0,00	9.069,09	20	0,00	9.042,90	20	0,00	3.500,39
BLAZEWICZ4	27	3,70	TL	27	3,70	TL	28	20,70	TL
BLAZEWICZ5	35	14,29	TL	35	14,29	TL	35	21,02	TL
SHAPES2	14	0,00	2,05	14	0,00	2,08	14	0,00	2,16
SHAPES4	26	23,08	TL	26	23,08	TL	25	16,00	TL
SHAPES5	31	25,81	TL	31	29,03	TL	31	28,06	TL
SHAPES7	48	41,63	TL	45	33,33	TL	43	31,02	TL
SHAPES9	54	39,96	TL	54	39,96	TL	-	-	TL
SHAPES15	67	40,42	TL	67	40,42	TL	67	38,85	TL

TL - limite de tempo atingido.

Tabela 12 – Avaliação das desigualdades válidas para o modelo de cobertura por cliques.

	Sem desigualdades			Desigualdades (7.5)			Desigualdades (7.2) e (7.5)		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,12	8	0,00	0,15	8	0,00	0,11
RCO2	15	0,00	0,56	15	0,00	0,55	15	0,00	0,57
RCO3	22	0,00	63,19	22	0,00	79,12	22	0,00	70,73
RCO4	29	0,00	8.343,11	29	0,00	5.237,85	29	0,00	13.466,08
RCO5	36	2,94	TL	36	4,20	TL	36	4,31	TL
BLAZEWICZ1	8	0,00	0,20	8	0,00	0,20	8	0,00	0,20
BLAZEWICZ2	14	0,00	3,64	14	0,00	3,67	14	0,00	3,51
BLAZEWICZ3	20	0,00	45,50	20	0,00	45,41	20	0,00	421,27
BLAZEWICZ4	27	0,00	13.791,11	27	0,00	13.734,89	27	3,70	TL
BLAZEWICZ5	34	5,88	TL	34	5,88	TL	34	10,51	TL
SHAPES2	14	0,00	0,62	14	0,00	0,17	14	0,00	0,17
SHAPES4	25	0,00	7.566,50	25	0,00	6.736,26	25	20,21	TL
SHAPES5	30	6,67	TL	29	3,45	TL	30	23,80	TL
SHAPES7	43	20,93	TL	43	18,60	TL	42	28,18	TL
SHAPES9	51	31,10	TL	50	32,55	TL	53	35,09	TL
SHAPES15	67	40,45	TL	67	38,13	TL	67	37,38	TL

TL - limite de tempo atingido.

melhores soluções e menor *gap* de otimalidade para todas as instâncias avaliadas. Além disso, esse modelo provou a otimalidade em menor tempo para quase todas as instâncias, com exceção da instância RCO3.

Tabela 13 – Avaliação dos modelos com adição das desigualdades válidas (7.5).

	Modelo dos pontos			Modelo CC_RLFM_KMK		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,80	8	0,00	0,15
RCO2	15	0,00	1,67	15	0,00	0,55
RCO3	22	0,00	76,30	22	0,00	79,12
RCO4	29	0,00	7.185,35	29	0,00	5.237,85
RCO5	36	5,56	TL	36	4,20	TL
BLAZEWICZ1	8	0,00	0,91	8	0,00	0,20
BLAZEWICZ2	14	0,00	11,95	14	0,00	3,67
BLAZEWICZ3	20	0,00	9.042,90	20	0,00	45,41
BLAZEWICZ4	27	3,70	TL	27	0,00	13.734,89
BLAZEWICZ5	35	14,29	TL	34	5,88	TL
SHAPES2	14	0,00	2,08	14	0,00	0,17
SHAPES4	26	23,08	TL	25	0,00	6.736,26
SHAPES5	31	29,03	TL	29	3,45	TL
SHAPES7	45	33,33	TL	43	18,60	TL
SHAPES9	54	39,96	TL	50	32,55	TL
SHAPES15	67	40,42	TL	67	38,13	TL

TL - limite de tempo atingido.

7.2.2 Cálculo do limitante inferior

Para calcular um limitante inferior melhor para os modelos, foi utilizado o método iterativo descrito no Algoritmo 5 da Seção 7.1 com as desigualdades válidas (7.2) e (7.5), utilizando o limitante inferior trivial. O modelo adotado foi o CC_RLFM_KMK, uma vez que este é baseado em cliques maximais que fornecem uma relaxação linear mais forte. Em cada iteração do método, é resolvida a relaxação linear do modelo e caso haja alteração no valor da função objetivo, apenas as restrições (7.2) são atualizadas. O critério de parada adotado foi a alteração mínima da função objetivo $\epsilon = 10^{-6}$ ou 1.000 iterações.

A Tabela 14 está organizada conforme descrito a seguir. Na primeira coluna são descritas as instâncias e na segunda e terceira, os limitante inferior trivial e o obtido por esse método. Na quarta e quinta colunas são apresentados os limitantes superiores da melhor solução conhecida e da placa inicial. Por fim, as duas últimas colunas indicam o número de iterações até a convergência e o tempo de execução do método em segundos. Para todas as instâncias, exceto SHAPES2, o limitante inferior obtido pelo método é melhor que o trivial. Além disso, para as menores instâncias (RCO1, BLAZEWICZ1 e SHAPES2), o valor do limitante inferior é igual a solução ótima do problema. O tempo de execução do algoritmo foi inferior a cinco minutos para as instâncias dos grupos RCO e BLAZEWICZ. No entanto, algumas instâncias do grupo SHAPES apresentaram tempo de execução superiores a dez minutos (SHAPES7-15), com destaque para a instância SHAPES15 cujo tempo foi superior a uma hora.

Tabela 14 – Cálculo do limitante inferior.

Instância	Limitante inferior		Limitante superior		Iterações	Tempo(s)
	Trivial	Método	Melhor	Placa		
RCO1	6,30	8,00	8*	8	51	0,37
RCO2	12,60	14,99	15*	15	IL	3,63
RCO3	18,90	21,00	22*	22	233	5,04
RCO4	25,20	27,00	29*	29	93	11,15
RCO5	31,50	34,00	36	37	331	41,33
BLAZEWICZ1	5,40	8,00	8*	8	465	0,66
BLAZEWICZ2	10,80	13,00	14*	15	88	2,52
BLAZEWICZ3	16,20	19,00	20*	22	149	19,92
BLAZEWICZ4	21,60	25,00	27*	28	229	40,08
BLAZEWICZ5	27,00	31,00	34	37	310	248,70
SHAPES2	14,00	14,00	14*	14	1	0,02
SHAPES4	16,00	22,00	25*	27	166	51,95
SHAPES5	20,00	25,00	29	32	41	114,83
SHAPES7	28,00	34,00	42	48	510	949,49
SHAPES9	32,40	38,00	50	54	213	1.186,31
SHAPES15	39,90	46,77	67	67	IL	3.841,52

IL - limite de iterações atingido.

* - solução ótima.

7.2.3 Avaliação do limitante inferior e solução inicial

Na última etapa de testes, foram avaliados os efeitos de alterar o limitante inferior e informar uma solução inicial (Apêndice A.1) para o *solver*. Além disso, foram adicionadas as desigualdades válidas (7.5) a partir dos limitantes inferiores descritos na Tabela 14. O tempo limite de execução utilizado para todos os testes foi de 18.000 segundos.

Na Tabela 15, são reportados os resultados obtidos com a adição desigualdades válidas (7.5), o uso do novo limitante inferior e a disponibilização de uma solução inicial para o modelo dos pontos. Ao adicionar apenas o novo limitante inferior, o modelo dos pontos apresentou resultados melhores em função objetivo para as instâncias BLAZEWICZ5 e SHAPES, mas não obteve uma solução factível para a instância SHAPES9. Outro ponto positivo desta configuração é a redução do *gap* de otimalidade para instâncias grandes (RCO5, BLAZEWICZ5 e SHAPES4-7/15) e do tempo para provar a otimalidade para a instância BLAZEWICZ3. A disponibilização de uma solução inicial não provocou grandes alterações nesse modelo, com exceção do tempo para provar a otimalidade para as instâncias RCO4 e BLAZEWICZ3 e a redução do *gap* de otimalidade para a instância SHAPES9.

Por fim, na Tabela 16 são reportados os resultados obtidos com a adição desigualdades válidas (7.5), o uso do novo limitante inferior e a disponibilização de uma solução inicial para o modelo de cobertura por cliques. Ao adicionar apenas o novo limitante inferior, esse modelo conseguiu reduzir o *gap* de otimalidade de algumas instâncias grandes, como RCO5,

Tabela 15 – Avaliação do uso do novo limitante inferior e solução inicial para o modelo dos pontos.

	Desigualdades (7.5)			Desigualdades (7.5) e limitante inferior			Desigualdades (7.5), limitante inferior e solução inicial		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,80	8	0,00	0,57	8	0,00	0,49
RCO2	15	0,00	1,67	15	0,00	1,92	15	0,00	1,84
RCO3	22	0,00	76,30	22	0,00	212,57	22	0,00	119,29
RCO4	29	0,00	7.185,35	29	0,00	12.271,11	29	0,00	8.082,14
RCO5	36	5,56	TL	36	2,78	TL	36	2,78	TL
BLAZEWICZ1	8	0,00	0,91	8	0,00	1,68	8	0,00	1,76
BLAZEWICZ2	14	0,00	11,95	14	0,00	25,08	14	0,00	21,89
BLAZEWICZ3	20	0,00	9.042,90	20	0,00	795,94	20	0,00	1.466,62
BLAZEWICZ4	27	3,70	TL	27	3,70	TL	27	3,70	TL
BLAZEWICZ5	35	14,29	TL	34	5,88	TL	34	5,88	TL
SHAPES2	14	0,00	2,08	14	0,00	2,17	14	0,00	0,89
SHAPES4	26	23,08	TL	25	8,00	TL	26	11,54	TL
SHAPES5	31	29,03	TL	31	16,13	TL	32	15,63	TL
SHAPES7	45	33,33	TL	48	29,17	TL	48	29,17	TL
SHAPES9	54	39,96	TL	-	-	TL	54	29,63	TL
SHAPES15	67	40,42	TL	67	30,20	TL	67	30,20	TL

TL - limite de tempo atingido.

SHAPES9/15, mas não provou a otimalidade para a instância SHAPES4. Analogamente ao modelo dos pontos, a disponibilização de uma solução inicial não provocou grandes alterações no modelo de cobertura por cliques, com exceção do tempo para provar a otimalidade para as instâncias RCO4 e SHAPES4. No entanto, nesta configuração não foi possível provar a otimalidade para a instância BLAZEWICZ4.

Analisando as Tabelas 15 e 16, podemos concluir que o modelo de cobertura por cliques usando apenas as desigualdades válidas (7.5) obteve os melhores resultados em função objetivo para todas as instâncias. Além disso, este modelo encontrou os menores valores de *gap* de otimalidade para quase todas as instâncias.

7.3 Considerações

Neste capítulo, foram propostas desigualdades válidas para os modelos dos pontos e de cobertura por cliques e um método iterativo para calcular um limitante inferior. Os resultados obtidos demonstraram que a adição das desigualdades válidas (7.5) pode reduzir significativamente o tempo necessário para provar a otimalidade de algumas instâncias. Além disso, as desigualdades válidas (7.2) e (7.5) foram utilizadas de forma eficiente para calcular um limitante inferior para os modelos dos pontos e de cobertura por cliques, o que pode tornar as relaxações lineares dos modelos mais fortes.

Tabela 16 – Avaliação do uso do novo limitante inferior e solução inicial para o modelo de cobertura por cliques.

	Desigualdades (7.5)			Desigualdades (7.5) e limitante inferior			Desigualdades (7.5), limitante inferior e solução inicial		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0,00	0,15	8	0,00	0,07	8	0,00	0,06
RCO2	15	0,00	0,55	15	0,00	11,97	15	0,00	12,44
RCO3	22	0,00	79,12	22	0,00	140,10	22	0,00	87,23
RCO4	29	0,00	5.237,85	29	0,00	5.495,15	29	0,00	2.890,03
RCO5	36	4,20	TL	36	2,78	TL	36	2,78	TL
BLAZEWICZ1	8	0,00	0,20	8	0,00	0,16	8	0,00	0,13
BLAZEWICZ2	14	0,00	3,67	14	0,00	4,51	14	0,00	8,29
BLAZEWICZ3	20	0,00	45,41	20	0,00	393,95	20	0,00	386,48
BLAZEWICZ4	27	0,00	13.734,89	27	0,00	15.388,00	27	3,70	TL
BLAZEWICZ5	34	5,88	TL	34	5,88	TL	34	5,88	TL
SHAPES2	14	0,00	0,17	14	0,00	0,17	14	0,00	0,08
SHAPES4	25	0,00	6.736,26	25	4,00	TL	25	0,00	3.939,39
SHAPES5	29	3,45	TL	30	6,67	TL	30	6,67	TL
SHAPES7	43	18,60	TL	44	20,45	TL	44	20,45	TL
SHAPES9	50	32,55	TL	50	24,00	TL	54	29,63	TL
SHAPES15	67	38,13	TL	67	30,20	TL	67	30,10	TL

TL - limite de tempo atingido.

Diversas comparações foram feitas entre o modelo de cobertura por cliques e o modelo dos pontos, considerando a adição de desigualdades válidas, o emprego de um limitante inferior melhor e a disponibilização de uma solução inicial. Em todos os testes realizados, o modelo de cobertura por cliques demonstrou-se igual ou superior ao modelo dos pontos para todas as instâncias, considerando o valor da função objetivo. O modelo de cobertura por cliques também obteve *gaps* de otimalidade menores e provou otimalidade em menor tempo que o modelo dos pontos para quase todas as instâncias avaliadas.

8 Considerações finais e trabalhos futuros

Neste trabalho foram propostos modelos matemáticos para o problema de empacotamento em faixas de peças irregulares. Todos os modelos propostos foram baseados no modelo dos pontos (TOLEDO et al., 2013). Inicialmente, foram apresentados dois modelos reduzidos para o problema. Estes modelos tiveram um desempenho melhor que o modelo dos pontos para instâncias com poucas repetições de peças, mas devido à baixa qualidade das suas relaxações lineares, não conseguiram provar a otimalidade para instâncias com várias repetições de peças.

Com o objetivo de reduzir o número de restrições e aumentar a qualidade da relaxação linear do modelo dos pontos, foram estudados modelos baseados em cobertura por cliques. Para calcular estas coberturas, foram empregadas heurísticas baseadas na heurística RLF (LEIGHTON, 1979) e de Kellerman (1973). Além disso, foi proposta a heurística KMK para o problema de cobertura de arestas por cliques. Esta heurística apresentou resultados significativamente melhores que outras heurísticas clássicas da literatura para todos os grafos avaliados neste trabalho, o que indica que este algoritmo pode ser promissor em outras áreas.

Outras contribuições relevantes deste trabalho são a introdução de desigualdades válidas para os modelos dos pontos e de cobertura por cliques e a criação de um método iterativo para calcular seu limitante inferior, que podem tornar as relaxações lineares dos modelos mais fortes.

O modelo de cobertura por cliques CC_RLFM_KMK obteve um desempenho igual ou superior ao modelo dos pontos para todas as instâncias avaliadas, considerando o valor da função objetivo e o *gap* de otimalidade. Este desempenho se manteve praticamente inalterado, mesmo considerando diversas configurações, como a adição de desigualdades válidas, o emprego de um limitante inferior melhor e a disponibilização de uma solução inicial. Deve-se destacar também que o modelo CC_RLFM_KMK provou otimalidade em menor tempo para quase todas as instâncias avaliadas.

É importante ressaltar que o modelo de cobertura por cliques conseguiu provar a otimalidade para instâncias com até 28 peças (RCO4 e BLAZEWICZ4) e encontrar melhores soluções que as obtidas por Toledo et al. (2013) para instâncias grandes (RCO5, BLAZEWICZ5 e SHAPES5-9), o que demonstra sua relevância para a literatura.

As propostas apresentadas sugerem os seguintes trabalhos futuros:

- **Avaliação de outras instâncias** - avaliar os modelos dos pontos e de cobertura por cliques para outras instâncias clássicas da literatura;

- **Heurísticas construtivas para o problema de cobertura de arestas por cliques** - propor algoritmos baseados na heurística proposta por [Gramm et al. \(2006\)](#) e utilizá-los para construir coberturas de arestas com o objetivo de reduzir o tempo computacional da fase de pré-processamento;
- **Desigualdades válidas do politopo do conjunto estável** - estudar o impacto da adição das desigualdades válidas clássicas associadas ao politopo do conjunto estável ([REBENNACK; REINELT; PARDALOS, 2012](#)) na relaxação linear do modelo de cobertura por cliques;
- **Branch-and-cut** - desenvolver um algoritmo *branch-and-cut* considerando as desigualdades válidas introduzidas neste trabalho e as clássicas do politopo do conjunto estável.

Referências

- ÁLVAREZ-VALDÉS, R.; MARTÍNEZ, A.; TAMARIT, J. A branch & bound algorithm for cutting and packing irregularly shaped pieces. *International Journal of Production Economics*, v. 145, p. 463 – 477, 2013.
- BABU, A. R.; BABU, N. R. A generic approach for nesting of 2-d parts in 2-d sheets using genetic and heuristic algorithms. *Computer-Aided Design*, v. 33, p. 879 – 891, 2001.
- BENNELL, J. A.; DOWSLAND, K. A. Hybridising tabu search with optimisation techniques for irregular stock cutting. *Management Science*, v. 47, p. 1160–1172, 2001.
- BENNELL, J. A.; OLIVEIRA, J. F. The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, v. 184, p. 397 – 415, 2008.
- BENNELL, J. A.; OLIVEIRA, J. F. A tutorial in irregular shape packing problems. *Journal of the Operational Research Society*, v. 60, p. S93–S105, 2009.
- BENNELL, J. A.; SONG, X. A beam search implementation for the irregular shape packing problem. *Journal of Heuristics*, v. 16, p. 167–188, 2010.
- BŁAŻEWICZ, J.; HAWRYLUK, P.; WALKOWIAK, R. Using a tabu search approach for solving the two-dimensional irregular cutting problem. *Annals of Operations Research*, v. 41, p. 313–325, 1993.
- BOLLOBÁS, B.; THOMASON, A. Random graphs of small order. In: KAROŃSKI, M.; RUCIŃSKI, A. (Ed.). *Random Graphs '83 Based on lectures presented at the 1st Poznań Seminar on Random Graphs*. New York: North-Holland, 1985, (North-Holland Mathematics Studies, v. 118). p. 47 – 97.
- BOMZE, I. M.; BUDINICH, M.; PARDALOS, P. M.; PELILLO, M. The maximum clique problem. In: DU, D.-Z.; PARDALOS, P. (Ed.). *Handbook of Combinatorial Optimization: Supplement Volume A*. Boston: Kluwer Academic Publishers, 1999. p. 1–74.
- BRÉLAZ, D. New methods to color the vertices of a graph. *Communications of the ACM*, v. 22, p. 251–256, 1979.
- BROWN, J. R. Chromatic scheduling and the chromatic number problem. *Management Science*, v. 19, p. 456–463, 1972.
- BURKE, E.; HELLIER, R.; KENDALL, G.; WHITWELL, G. A new bottom-left-fill heuristic algorithm for the two-dimensional irregular packing problem. *Operations Research*, v. 54, p. 587–601, 2006.
- BURKE, E. K.; HELLIER, R. S. R.; KENDALL, G.; WHITWELL, G. Irregular packing using the line and arc no-fit polygon. *Operations Research*, v. 58, p. 948–970, 2010.
- CARRAVILLA, M. A.; RIBEIRO, C.; OLIVEIRA, J. F. Solving nesting problems with non-convex polygons by constraint logic programming. *International Transactions in Operational Research*, v. 10, p. 651–663, 2003.

- CULBERSON, J. C.; LUO, F. Exploring the k-colorable landscape with iterated greedy. In: JOHNSON, D. S.; TRICK, M. A. (Ed.). *Cliques, Coloring, and Satisfiability: 2nd DIMACS Implementation Challenge, 1993*. Boston: American Mathematical Society, 1996, (DIMACS Series in Discrete Mathematics and Theoretical Computer Science). p. 245–284.
- DOWSLAND, K. A.; DOWSLAND, W. B. Solution approaches to irregular nesting problems. *European Journal of Operational Research*, v. 84, p. 506 – 521, 1995. Cutting and Packing.
- DOWSLAND, K. A.; VAID, S.; DOWSLAND, W. B. An algorithm for polygon placement using a bottom-left strategy. *European Journal of Operational Research*, v. 141, p. 371 – 381, 2002.
- EGEBLAD, J.; NIELSEN, B. K.; ODGAARD, A. Fast neighborhood search for two- and three-dimensional nesting problems. *European Journal of Operational Research*, v. 183, p. 1249 – 1266, 2007.
- ELKERAN, A. A new approach for sheet nesting problem using guided cuckoo search and pairwise clustering. *European Journal of Operational Research*, v. 231, p. 757 – 769, 2013.
- ERDŐS, P.; GOODMAN, A. W.; PÓSA, L. The representation of a graph by set intersections. *Canadian Journal of Mathematics*, v. 18, p. 106–112, 1966.
- FISCHER, A. D.; DAGLI, C. H. Employing subgroup evolution for irregular-shape nesting. *Journal of Intelligent Manufacturing*, v. 15, p. 187–199, 2004.
- FISCHETTI, M.; LUZZI, I. Mixed-integer programming models for nesting problems. *Journal of Heuristics*, v. 15, p. 201–226, 2009.
- FOWLER, R. J.; PATERSON, M. S.; TANIMOTO, S. L. Optimal packing and covering in the plane are np-complete. *Information Processing Letters*, v. 12, p. 133 – 137, 1981.
- GALINIER, P.; HERTZ, A. A survey of local search methods for graph coloring. *Computers & Operations Research*, v. 33, p. 2547 – 2562, 2006.
- GOMES, A. M.; OLIVEIRA, J. F. A 2-exchange heuristic for nesting problems. *European Journal of Operational Research*, v. 141, p. 359 – 370, 2002.
- GOMES, A. M.; OLIVEIRA, J. F. Solving irregular strip packing problems by hybridising simulated annealing and linear programming. *European Journal of Operational Research*, v. 171, p. 811 – 829, 2006.
- GRAMM, J.; GUO, J.; HÜFFNER, F.; NIEDERMEIER, R. Data reduction, exact, and heuristic algorithms for clique cover. In: *2006 Proceedings of the Eighth Workshop on Algorithm Engineering and Experiments (ALENEX)*. Philadelphia: Society for Industrial and Applied Mathematics, 2006. p. 86–94.
- GRAMM, J.; GUO, J.; HÜFFNER, F.; NIEDERMEIER, R. Data reduction and exact algorithms for clique cover. *Journal of Experimental Algorithmics*, v. 13, p. 2:2.2–2:2.15, 2008.
- HANSEN, P.; LABBÉ, M.; SCHINDL, D. Set covering and packing formulations of graph coloring: Algorithms and first polyhedral results. *Discrete Optimization*, v. 6, p. 135 – 147, 2009.

- HOOVER, D. N. Complexity of graph covering problems for graphs of low degree. *Journal of Combinatorial Mathematics and Combinatorial Computing*, v. 11, p. 187–208, 1992.
- HSU, W.-L.; TSAI, K.-H. Linear time algorithms on circular-arc graphs. *Information Processing Letters*, v. 40, p. 123 – 129, 1991.
- IMAMICHI, T.; YAGIURA, M.; NAGAMOCCHI, H. An iterated local search algorithm based on nonlinear programming for the irregular strip packing problem. *Discrete Optimization*, v. 6, p. 345 – 361, 2009.
- JAKOBS, S. On genetic algorithms for the packing of polygons. *European Journal of Operational Research*, v. 88, p. 165 – 181, 1996.
- JOHNSON, D. S.; ARAGON, C. R.; MCGEOCH, L. A.; SCHEVON, C. Optimization by simulated annealing: an experimental evaluation; part ii, graph coloring and number partitioning. *Operations research*, v. 39, p. 378–406, 1991.
- KARP, R. Reducibility among combinatorial problems. In: MILLER, R.; THATCHER, J.; BOHLINGER, J. (Ed.). *Complexity of Computer Computations*. New York: Springer US, 1972, (The IBM Research Symposia Series). p. 85–103.
- KELLERMAN, E. Computation of masks for use in determining keyword conflict. *IBM Technical Disclosure Bulletin*, v. 16, p. 544–546, 1973.
- KOU, L. T.; STOCKMEYER, L. J.; WONG, C. K. Covering edges by cliques with regard to keyword conflicts and intersection graphs. *Communications of the ACM*, v. 21, p. 135–139, 1978.
- LEIGHTON, F. T. A graph coloring algorithm for large scheduling problems. *Journal of Research of the National Bureau of Standards*, v. 84, p. 489–506, 1979.
- LEUNG, S. C.; LIN, Y.; ZHANG, D. Extended local search algorithm based on nonlinear programming for two-dimensional irregular strip packing problem. *Computers & Operations Research*, v. 39, p. 678 – 686, 2012.
- LI, Z.; MILENKOVIC, V. Compaction and separation algorithms for non-convex polygons and their applications. *European Journal of Operational Research*, v. 84, p. 539 – 561, 1995.
- MA, S.; WALLIS, W. D.; WU, J. Clique covering of chordal graphs. *Utilitas Mathematica*, v. 36, p. 151–152, 1989.
- MALAGUTI, E.; TOTH, P. A survey on vertex coloring problems. *International Transactions in Operational Research*, v. 17, p. 1–34, 2010.
- MEHROTRA, A.; TRICK, M. A. A column generation approach for graph coloring. *informatics Journal on Computing*, v. 8, p. 344–354, 1996.
- MÉNDEZ-DÍAZ, I.; ZABALA, P. A branch-and-cut algorithm for graph coloring. *Discrete Applied Mathematics*, v. 154, p. 826 – 847, 2006.
- MÉNDEZ-DÍAZ, I.; ZABALA, P. A cutting plane algorithm for graph coloring. *Discrete Applied Mathematics*, v. 156, p. 159 – 179, 2008.

MONSON, S. D.; PULLMAN, N. J.; REES, R. A survey of clique and biclique coverings and factorizations of $(0, 1)$ -matrices. *Bulletin of the ICA*, v. 14, p. 17–86, 1995.

OLIVEIRA, J.; FERREIRA, J. Algorithms for nesting problems. In: VIDAL, R. (Ed.). *Applied Simulated Annealing*. Berlin: Springer Berlin Heidelberg, 1993, (Lecture Notes in Economics and Mathematical Systems, v. 396). p. 255–273.

OLIVEIRA, J. F.; GOMES, A. M.; FERREIRA, J. S. Topos - a new constructive algorithm for nesting problems. *OR-Spektrum*, v. 22, p. 263–284, 2000.

ORLIN, J. Contentment in graph theory: Covering graphs with cliques. *Indagationes Mathematicae (Proceedings)*, v. 80, p. 406 – 424, 1977.

PARDALOS, P.; XUE, J. The maximum clique problem. *Journal of Global Optimization*, v. 4, p. 301–328, 1994.

PARDALOS, P. M.; MAVRIDOU, T.; XUE, J. The graph coloring problem: A bibliographic survey. In: DU, D.-Z.; PARDALOS, P. (Ed.). *Handbook of Combinatorial Optimization*. Boston: Kluwer Academic Publishers, 1998. v. 2, p. 331–395.

PULLMAN, N. Clique covering of graphs IV. algorithms. *SIAM Journal on Computing*, v. 13, p. 57–75, 1984.

PULLMAN, N. J. Clique coverings of graphs — a survey. In: CASSE, L. R. A. (Ed.). *Combinatorial Mathematics X*. Berlin: Springer Berlin Heidelberg, 1983, (Lecture Notes in Mathematics, v. 1036). p. 72–85.

REBENNACK, S.; REINELT, G.; PARDALOS, P. M. A tutorial on branch and cut algorithms for the maximum stable set problem. *International Transactions in Operational Research*, v. 19, p. 161–199, 2012.

RIBEIRO, C.; CARRAVILLA, M. A.; OLIVEIRA, J. F. Applying constraint logic programming to the resolution of nesting problems. *Pesquisa Operacional*, v. 19, p. 239–247, 1999.

ROBERTS, F. S. Applications of edge coverings by cliques. *Discrete Applied Mathematics*, v. 10, p. 93 – 109, 1985.

SATO, A. K.; MARTINS, T. C.; TSUZUKI, M. S. G. An algorithm for the strip packing problem using collision free region and exact fitting placement. *Computer-Aided Design*, v. 44, p. 766 – 777, 2012.

STOYAN, Y.; NOVOZHILOVA, M.; KARTASHOV, A. Mathematical model and method of searching for a local extremum for the non-convex oriented polygons allocation problem. *European Journal of Operational Research*, v. 92, p. 193 – 210, 1996.

STOYAN, Y.; SCHEITHAUER, G.; GIL, N.; ROMANOVA, T. ϕ -functions for complex 2d-objects. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, v. 2, p. 69–84, 2004.

STOYAN, Y.; TERNO, J.; SCHEITHAUER, G.; GIL, N.; ROMANOVA, T. ϕ -functions for primary 2d-objects. *Studia Informatica Universalis*, v. 2, p. 1–32, 2001.

- TOLEDO, F. M.; CARRAVILLA, M. A.; RIBEIRO, C.; OLIVEIRA, J. F.; GOMES, A. M. The dotted-board model: A new MIP model for nesting irregular shapes. *International Journal of Production Economics*, v. 145, p. 478 – 487, 2013.
- UMETANI, S.; YAGIURA, M.; IMAHORI, S.; IMAMICHI, T.; NONOBE, K.; IBARAKI, T. Solving the irregular strip packing problem via guided local search for overlap minimization. *International Transactions in Operational Research*, v. 16, p. 661–683, 2009.
- WÄSCHER, G.; HAUSSNER, H.; SCHUMANN, H. An improved typology of cutting and packing problems. *European Journal of Operational Research*, v. 183, p. 1109 – 1130, 2007.
- WELSH, D. J. A.; POWELL, M. B. An upper bound for the chromatic number of a graph and its application to timetabling problems. *The Computer Journal*, v. 10, p. 85–86, 1967.
- WU, Q.; HAO, J.-K. A review on algorithms for maximum clique problems. *European Journal of Operational Research*, v. 242, p. 693 – 709, 2015.

APÊNDICE A – Soluções

A seguir são ilustradas as soluções iniciais utilizadas nos experimentos computacionais da Subseção 7.2.2 e as melhores soluções obtidas neste trabalho. Os números em parênteses indicam o valor da função objetivo e o asterisco destaca que a solução é ótima.

A.1 Soluções iniciais

Figura 25 – Soluções iniciais utilizadas para o grupo RCO.

BLAZEWCZ1 (8*)

BLAZEWCZ2 (15)

BLAZEWCZ3 (22)

BLAZEWCZ4 (28)

BLAZEWCZ5 (37)

Figura 26 – Soluções iniciais utilizadas para o grupo BLAZEWCZ.

Figura 27 – Soluções iniciais utilizadas para o grupo SHAPES.

A.2 Melhores soluções

RCO1 (8*)

RCO2 (15*)

RCO3 (22*)

RCO4 (29*)

RCO5 (36)

Figura 28 – Melhores soluções obtidas para o grupo RCO.

Figura 29 – Melhores soluções obtidas para o grupo BLAZEWCZ.

Figura 30 – Melhores soluções obtidas para o grupo SHAPES.