UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Resolução de problemas de empacotamento de itens irregulares usando técnicas de programação não-linear

Jeinny Maria Peralta Polo

Tese de Doutorado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Jeinny Maria Peralta Polo

Resolução de problemas de empacotamento de itens irregulares usando técnicas de programação não-linear

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

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Marina Andretta

Coorientador: Prof. Dr. José Fernando da Costa Oliveira

USP – São Carlos
Maio de 2018

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

P778r Polo, Jeinny Maria Peralta
Resolução de problemas de empacotamento de itens
irregulares usando técnicas de programação não-
linear / Jeinny Maria Peralta Polo; orientadora
Marina Andretta; coorientador José Fernando da
Costa Oliveira. -- São Carlos, 2018.
100 p.

Tese (Doutorado - Programa de Pós-Graduação em
Ciências de Computação e Matemática Computacional) --
Instituto de Ciências Matemáticas e de Computação,
Universidade de São Paulo, 2018.

1. Problemas de empacotamento de itens
irregulares. 2. retas separadoras. 3. rotação livre.
4. programação não-linear. 5. parábolas separadoras.
I. Andretta, Marina, orient. II. Oliveira, José
Fernando da Costa , coorient. III. Título.

Jeinny Maria Peralta Polo

**Solving irregular packing problems using non-linear
programming techniques**

Doctoral dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP, in
partial fulfillment of the requirements for the degree of
the Doctorate Program in Computer Science and
Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Marina Andretta

Co-advisor: Prof. Dr. José Fernando da Costa Oliveira

USP – São Carlos

May 2018

AGRADECIMENTOS

Foram muitas as pessoas que estiveram ao meu lado durante essa caminhada. Talvez eu não consiga expressar toda a minha gratidão com palavras...

Meu agradecimento mais profundo só poderia ser dedicado a uma pessoa: ao meu esposo, Alfredo, por estar comigo o tempo todo, nos momentos bons e nos momentos mais difíceis, que não foram raros neste último ano. Por sempre me fazer acreditar que chegaria ao final desta difícil, porém gratificante etapa. Sou grata por cada gesto carinhoso, por cada palavra de apoio. Obrigada meu Amor.

Agradeço ao meu lindo filho, Alejandro, por me dar o seu sorriso todas as manhãs, o que me fez mais forte para lutar.

Agradeço a minha mãe, minha sogra e à tia Mireya por cuidar muito bem do meu filho enquanto eu alcançava meus objetivos.

Agradeço a toda minha família (mãe, pai, irmã, sobrinhos, tios, primos, sogros, cunhadas) pelo apoio incondicional, torcida e confiança que sempre depositam em mim; pelos momentos que não estivemos juntos e souberam entender; mas sobretudo pelos momentos de alegria e descanso de trabalho que sim tivemos. Obrigada!

Agradeço aos todos meus amigos que estavam comigo, mesmo que estivessem a muitos quilômetros de distância. Especialmente ao John, Yaneth, Lina e Oriana, obrigada por todas as conversas, pelo companheirismo e pelo apoio em certos momentos difíceis.

Minha gratidão especial à professora Marina, pela oportunidade de tê-la como orientadora de Mestrado e Doutorado, obrigada por sua condução do trabalho, por todo o apoio, a disponibilidade, pelo saber que transmitiu, pelas opiniões e críticas, e por toda a colaboração ao longo da realização deste trabalho.

Agradeço a meu coorientador, o professor José Fernando, pela ajuda e os excelentes aportes que ajudaram a conduzir bem este trabalho.

Agradeço a todos os professores do LOt pelos ensinamentos que passaram desde o mestrado, os quais foram, são e serão muito importantes para mim e para a minha vida profissional, assim como agradeço aos funcionários, que fazem com que tudo funcione da melhor maneira possível.

Agradeço aos meus companheiros do LOt pelo apoio e toda ajuda que me brindaram.

Agradeço ao Instituto de Ciências Matemáticas e de Computação, da Universidade de

São Paulo, por proporcionar meu aprendizado.

Agradeço à CNPq que financiou este projeto enquanto eu morei no Brasil e que também financiou a compra de um computador, ferramenta essencial a este projeto.

Às demais pessoas que contribuíram direta ou indiretamente na elaboração deste trabalho ou participaram da minha vida, e que, por ventura, eu tenha me esquecido de agradecer.

*“Por vezes sentimos que aquilo que fazemos não é senão uma gota de água no mar.
Mas o mar seria menor se lhe faltasse uma gota.”
(Madre Teresa de Calcuta)*

RESUMO

PERALTA, J. **Resolução de problemas de empacotamento de itens irregulares usando técnicas de programação não-linear**. 2018. 100 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

Os problemas de empacotamento de itens irregulares são problemas de corte e empacotamento, nos quais peças irregulares de menor tamanho (que chamamos de itens) devem ser empacotados inteiramente em uma peça grande (que chamamos de placa), obedecendo a restrições de não-sobreposição e minimizando as dimensões da placa. Para garantir a não-sobreposição, fazemos uso de retas separadoras, quer dizer, retas que separam um item de outro. Apresentamos modelos de programação não-linear para problemas de empacotamentos de itens regulares e irregulares que rotacionam livremente. Os itens podem ser círculos, polígonos convexos e não-convexos. A principal vantagem dos modelos é a simplicidade, já que estes utilizam somente conceitos básicos de geometria. Usamos o algoritmo de programação não-linear IPOPT (um algoritmo de tipo de pontos interiores), que faz parte da COIN-OR, para a resolução dos problemas. Testes computacionais foram executados usando instâncias conhecidas da literatura e os resultados foram comparados com resultados apresentados na literatura, obtidos com outras metodologias que também usam rotações livre, mostrando que nossos modelos são competitivos. Propomos também o uso de parábolas separadoras para a verificação de não-sobreposição na modelagem do problema, o que pode trazer ganhos computacionais e melhor qualidade de soluções.

Palavras-chave: Problemas de empacotamento de itens irregulares, retas separadoras, rotação livre, programação não-linear.

ABSTRACT

PERALTA, J. **Solving irregular packing problems using non-linear programming techniques**. 2018. 100 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2018.

The irregular packing problems are cutting and packing problems, in which smaller irregular pieces (which we call items) should be packaged entirely in one large piece (which we call a plate), obeying non-overlapping constraints and minimizing the dimensions of the plate. To ensure non-overlapping, we make use of separation lines, that is, lines that separate one item from another. We present nonlinear programming models for problems of packing regular and irregular items that rotate freely. The items can be circles, convex and nonconvex polygons. The main advantage of the models is their simplicity, because they use only basic geometry concepts. We use the nonlinear programming algorithm IPOPT (an algorithm of interior points type), which is part of COIN-OR, to solve the problems. Computational tests were performed using known instances of the literature and the results were compared with results presented in the literature, obtained with other methodologies that also use free rotations, showing that our models are competitive. We also propose the use of separating parabola to avoid items overlapping in the models, which could provide greater computational efficiency as well as solutions with better quality.

Keywords: Irregular packing problems, separation lines, free rotation, non-linear programming.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exemplo de leiaute na indústria têxtil. Fonte: (BENNELL; OLIVEIRA, 2009).	28
Figura 2 – Representação 0-1. Fonte: (BENNELL; OLIVEIRA, 2008).	28
Figura 3 – Item da instância <i>Swim</i> .	29
Figura 4 – Círculo aproximado por um polígono.	29
Figura 5 – a) Objetos primários. b) Objetos básicos. Fonte: (STOYAN; PANKRATOV; ROMANOVA, 2016).	30
Figura 6 – Decomposição de itens em objetos básicos. Fonte: (STOYAN; PANKRATOV; ROMANOVA, 2016).	30
Figura 7 – a) Cobertura de círculos. Fonte: (ROCHA <i>et al.</i> , 2015) b) Inscrição de círculos. Fonte:(JONES, 2013)	30
Figura 8 – Exemplo de posição relativa de dois polígonos. Fonte: (BENNELL; OLIVEIRA, 2008).	32
Figura 9 – Interpretação da função-D. Fonte: (BENNELL; OLIVEIRA, 2008).	33
Figura 10 – Geração do NFP_{AB} . a) Movimento do polígono B ao redor do polígono A para formar o NFP. b) Orientação dos polígonos A e B. c) Ordem das direções de A e B. d) NFP_{AB} . Fonte: (BENNELL; OLIVEIRA, 2008)	34
Figura 11 – Exemplo da função Φ para dois polígonos convexos. Fonte: (BENNELL <i>et al.</i> , 2010)	35
Figura 12 – Exemplo de NFR para um polígono e um círculo. Fonte: (MUNDIM; ANDRETTA; QUEIROZ, 2017)	35
Figura 13 – Aproximação da restrição de não-sobreposição usando círculos inscritos. Fonte: (JONES, 2013)	36
Figura 14 – Regra de posicionamento alternativa ao BL. Fonte: (BENNELL; OLIVEIRA, 2009).	38
Figura 15 – Mecanismo para definição de novos ângulos de orientação. Fonte: (ABEYSOORIYA; BENNELL; MARTINEZ-SYKORA, 2018).	38
Figura 16 – Métricas usadas por (DOWSLAND; VAID; DOWSLAND, 2002) para determinar a ordem estática dos itens. Fonte: adaptado de (BENNELL; OLIVEIRA, 2009).	40
Figura 17 – Leiautes resultantes de uma regra de posicionamento BL após movimentos na vizinhança. Fonte: (BENNELL; OLIVEIRA, 2009).	41
Figura 18 – Partição de um polígono não-convexo.	50
Figura 19 – Reta separadora de dois polígonos convexos.	52

Figura 20 – Retas separadoras de dois polígonos, sendo um deles não-convexo.	53
Figura 21 – Reta separadora de um polígono convexo e um círculo.	57
Figura 22 – Sobrepondo-se	58
Figura 23 – Tocando-se	58
Figura 24 – Separados	58
Figura 25 – Exemplo de duas soluções iniciais com o mesmo comprimento e soluções finais com comprimentos diferentes.	65
Figura 26 – Instância Albano	66
Figura 27 – Instância Blaz	66
Figura 28 – Instância Dagli	67
Figura 29 – Instância Jakobs1	67
Figura 30 – Instância Jakobs2	68
Figura 31 – Instância Marques	68
Figura 32 – Instância Poly1a	68
Figura 33 – Instância Poly2a	68
Figura 34 – Instância Poly3a	69
Figura 35 – Instância Poly4a	69
Figura 36 – Instância Poly5a	69
Figura 37 – Instância Poly10a	70
Figura 38 – Instância Poly20a	70
Figura 39 – Instância Shirts	70
Figura 40 – Instância Swim	71
Figura 41 – Instância Trousers	71
Figura 42 – Instância apresentada em (STOYAN; ZLOTNIK; CHUGAY, 2012).	72
Figura 43 – Instância: c1p1-1.	72
Figura 44 – Instância: c1p1-2.	73
Figura 45 – Instância: c1p1-3.	73
Figura 46 – Instância: c1p5a.	73
Figura 47 – Instância: c1p5b.	73
Figura 48 – Instância: c1p6a.	74
Figura 49 – Instância: c3p3.	74
Figura 50 – Instância: c6p3.	74
Figura 51 – Parábola separadora de dois polígonos.	79
Figura 52 – Separação de dois itens, sendo um deles não-convexo.	80
Figura 53 – Separação de dois itens, sendo um deles não-convexo.	80
Figura 54 – Curva separadora de quarto grau para dois polígonos não-convexos.	81
Figura 55 – Sobreposição dos polígonos.	82
Figura 56 – Posição relativa da parábola e as arestas	82
Figura 57 – Vértices em lados diferentes dos pontos de interseção	84

Figura 58 – Vértices no mesmo lado dos pontos de interseção	84
Figura 59 – Elementos de uma parábola.	97

LISTA DE TABELAS

Tabela 1 – Dados das instâncias	63
Tabela 2 – Porcentagens de compactação do comprimento na solução inicial ao comprimento na solução final.	65
Tabela 3 – Comparação dos nossos resultados com aqueles em (STOYAN; PANKRATOV; ROMANOVA, 2016) e (LIAO <i>et al.</i> , 2016).	76
Tabela 4 – Comparação dos nossos resultados com aqueles em (KALLRATH, 2009). .	77

LISTA DE ABREVIATURAS E SIGLAS

BL	<i>Bottom-Left</i>
BRKGA	<i>Biased Random Key Genetic Algorithm</i>
C&P	problemas de corte e empacotamento
IPP	<i>Irregular Packing Problems</i>
ISPP	<i>Irregular Strip Packing Problem</i>
MEP	<i>Minimal Enclosure Problem</i>
MISSP	<i>Model for a Irregular Strip Packing Problem</i>
MMEP	<i>Model for a Minimal Enclosure Problem</i>
NFP	<i>No-Fit Polygon</i>
NFR	<i>No-Fit Raster</i>
PLIM	Programação Linear Inteira Mista
SA	<i>Simulated Annealing</i>
TR	<i>Top Right</i>

SUMÁRIO

1	INTRODUÇÃO	23
2	PROBLEMA DE EMPACOTAMENTO DE ITENS IRREGULARES	27
2.1	Geometria dos problemas de empacotamento de itens irregulares .	27
2.1.1	<i>Representação dos itens</i>	28
2.1.2	<i>Verificação de não-sobreposição</i>	29
2.2	Abordagens para a resolução dos problemas de empacotamento de itens irregulares	36
2.2.1	<i>Heurísticas</i>	37
2.2.1.1	<i>Heurísticas construtivas</i>	37
2.2.1.2	<i>Buscas locais</i>	40
2.2.2	<i>Abordagens exatas</i>	43
2.2.2.1	<i>Programação linear inteira mista</i>	43
2.2.2.2	<i>Programação em lógica com restrições</i>	44
2.2.2.3	<i>Programação não-linear</i>	44
2.2.3	<i>Abordagens exatas combinadas com heurísticas</i>	45
2.3	Problemas de empacotamento de itens irregulares e círculos simul- taneamente	46
3	MODELOS DE PROBLEMAS DE EMPACOTAMENTO DE ITENS IRREGULARES USANDO RETAS SEPARADORAS	49
3.1	Modelos para problemas de empacotamento de itens irregulares considerando rotações livres	50
3.1.1	<i>Representação dos polígonos nos modelos</i>	50
3.1.2	<i>Vértices dos polígonos na forma geral</i>	51
3.1.3	<i>Reta separadora</i>	51
3.1.4	<i>Modelos matemáticos</i>	53
3.1.4.1	<i>Modelo para o problema de empacotamento de itens irregulares em faixa</i> .	54
3.1.4.2	<i>Modelo para o problema de empacotamento de itens irregulares em uma envoltória retangular</i>	56
3.2	Modelos para problemas de empacotamento de itens irregulares e círculos simultaneamente	56
3.2.1	<i>Restrições de contenção</i>	57

3.2.2	<i>Restrições de não-sobreposição</i>	57
4	MÉTODOS DE RESOLUÇÃO E RESULTADOS NUMÉRICOS . . .	61
4.1	Resultados numéricos para o problema de empacotamento de itens irregulares em faixa	62
4.1.1	<i>Solução inicial</i>	63
4.1.2	<i>Comparação dos resultados</i>	66
4.2	Resultados numéricos para o problema de empacotamento de itens irregulares em uma envoltória retangular	69
5	MODELO DE UM PROBLEMA DE EMPACOTAMENTO DE ITENS IRREGULARES USANDO PARÁBOLAS SEPARADORAS	79
5.1	Restrições para garantir a separação de dois polígonos usando pa- rábolas separadoras	81
5.1.1	<i>Posição relativa entre a parábola e as arestas de um polígono</i> . . .	82
5.1.2	<i>Restrições de não-sobreposição</i>	83
6	CONCLUSÕES E TRABALHOS FUTUROS	87
	REFERÊNCIAS	91
APÊNDICE A	PROCEDIMENTO PARA DETERMINAR OS PARÂ- METROS DE UMA PARÁBOLA	97

INTRODUÇÃO

Os problemas de empacotamento de itens irregulares ou *Irregular Packing Problems* (IPP) buscam determinar um arranjo ótimo de objetos menores (que chamamos de itens, sendo pelo menos um deles irregular) dentro de objetos maiores (que chamamos de placas), obedecendo certas restrições e sempre visando minimizar custos, seja maximizando a ocupação de espaços ou minimizando desperdícios.

A definição anterior dos IPP é geral e abrange muitas variantes do problema. Os IPP são problemas de corte e empacotamento (C&P); as variantes dos C&P que definem o tipo do problema foram classificadas em (WÄSCHER; HAUBNER; SCHUMANN, 2007). Os critérios utilizados para classificar os problemas foram: dimensionalidade, tipo de atribuição, homogeneidade / heterogeneidade dos itens, forma dos itens, tamanho, forma, quantidade e qualidade da região e quanto ao objetivo.

Neste trabalho, trataremos em particular de dois tipos de IPP de duas dimensões que obedecem restrições de não-sobreposição entre os itens e restrições de contenção, quer dizer que os itens devem estar inteiramente contidos na placa. Nos dois problemas, os itens podem rotacionar livremente. O primeiro problema é o problema de empacotamento de itens irregulares em faixa ou *Irregular Strip Packing Problem* (ISPP), que é um problema com uma dimensão aberta, quer dizer, a placa tem uma determinada largura, enquanto o comprimento é infinito e tem de ser minimizado. O segundo problema é o problema de empacotamento de itens em uma envoltória retangular ou *Minimal Enclosure Problem* (MEP), que é um problema com duas dimensões abertas, quer dizer, a placa é um retângulo a ser desenhado de forma que a área seja minimizada. Nos problemas, empacotamos simultaneamente polígonos convexos ou não-convexos e círculos.

As restrições de não-sobreposição entre os itens tornam os IPP de difícil resolução. Na literatura existem ferramentas para solucionar este problema geométrico, estas são, métodos de rasterização, trigonometria direta, polígono de obstrução ou *No-Fit Polygon* (NFP), função Φ

([BENNEL; OLIVEIRA, 2008](#)) e *No-Fit Raster* (NFR) ([MUNDIM; ANDRETTA; QUEIROZ, 2017](#)).

Neste trabalho, representamos os itens irregulares por polígonos e utilizamos trigonometria direta, mais especificamente retas separadoras, para garantir a não-sobreposição dos itens. Uma reta separa dois itens se um item está em um lado da reta e o outro item está no lado oposto. Adicionalmente, apresentamos uma nova ferramenta para garantir que dois itens estejam separados, as parábolas separadoras. Como nos casos das retas, uma parábola separa dois itens se um item está em um lado da parábola e o outro item está no lado oposto.

Devido à dificuldade intrínseca e à importância prática, diversas técnicas de resolução para os IPP têm sido desenvolvidas. A maioria das técnicas são baseadas em heurísticas ([ALBANO; SAPUPPO, 1980](#); [MARQUES; BISPO; SENTIEIRO, 1991](#); [OLIVEIRA; FERREIRA, 1993](#); [GOMES; OLIVEIRA, 2002](#); [EGEBLAD; NIELSEN; ODGAARD, 2007](#); [NIELSEN, 2007](#); [ABEYSOORIYA; BENNEL; MARTINEZ-SYKORA, 2018](#)) e em uma menor quantidade em Programação Linear Inteira Mista (PLIM) ([FISCHETTI; LUZZI, 2009](#); [ALVAREZ-VALDES; MARTINEZ; TAMARIT, 2013](#); [TOLEDO *et al.*, 2013](#)). Adicionalmente, encontramos na recente literatura dois trabalhos em que são apresentadas math-heurísticas. ([CHERRI; CARRAVILLA; TOLEDO, 2016](#); [MARTINEZ-SYKORA *et al.*, 2017](#)). Também, diferentes modelos de programação não-linear ([CHERNOV; STOYAN; ROMANOVA, 2010](#); [JONES, 2013](#); [ROCHA *et al.*, 2015](#); [STOYAN; PANKRATOV; ROMANOVA, 2016](#)) têm sido propostos.

Existem poucos trabalhos na literatura para os IPP que consideram rotação livre dos itens. Estes trabalhos são principalmente modelos de programação não-linear ([CHERNOV; STOYAN; ROMANOVA, 2010](#); [JONES, 2013](#); [ROCHA *et al.*, 2015](#); [STOYAN; PANKRATOV; ROMANOVA, 2016](#)). Porém, encontramos uma math-heurística ([MARTINEZ-SYKORA *et al.*, 2017](#)) e uma heurística ([ABEYSOORIYA; BENNEL; MARTINEZ-SYKORA, 2018](#)) que podem considerar qualquer ângulo de rotação. Em ([MARTINEZ-SYKORA *et al.*, 2017](#)), um modelo de Programação Linear Inteira Mista (PLIM) é executado várias vezes, uma para cada orientação que se queira considerar. Em ([ABEYSOORIYA; BENNEL; MARTINEZ-SYKORA, 2018](#)), um subconjunto de possíveis rotações é avaliado; para a escolha deste subconjunto é elaborado um mecanismo que identifica ângulos promissores de acordo com o arranjo dos itens em cada solução parcial.

Como mencionamos anteriormente, o estudo e resolução de problemas de empacotamento de itens irregulares e círculos simultaneamente também é considerado neste trabalho. Vários modelos não-lineares têm sido empregados com sucesso para modelar o Problema de Empacotamento de Círculos em diversos tipos de placas ([WANG *et al.*, 2002](#); [STOYAN; YASKOV, 2004](#); [BIRGIN; MARTÍNEZ; RONCONI, 2005](#); [MLADENOVIC; PLASTRIA; UROSEVIC, 2005](#)), porém poucos trabalhos tratam com o problema de empacotamento de itens irregulares e círculos simultaneamente ([STOYAN; ZLOTNIK; CHUGAY, 2012](#); [MUNDIM; ANDRETTA; QUEIROZ, 2017](#)). Por outro lado, podemos encontrar na literatura um trabalho em que círculos

e polígonos regulares são empacotados simultaneamente (KALLRATH, 2009).

Desenvolvemos dois modelos de programação não-linear, um para o problema de empacotamento de itens irregulares e círculos em faixa e outro para o problema de empacotamento de itens irregulares e círculos em uma envoltória retangular, em que os itens rotacionam livremente. Fizemos experimentos numéricos para verificar o comportamento dos modelos e comparamos os resultados obtidos com os apresentados em (STOYAN; PANKRATOV; ROMANOVA, 2016; LIAO *et al.*, 2016; STOYAN; ZLOTNIK; CHUGAY, 2012) e (KALLRATH, 2009), que foram obtidos utilizando metodologias que permitem rotações livres. Encontramos sempre dimensões muito próximas, sendo um pouco maiores em todas as instâncias ao comparar com (STOYAN; PANKRATOV; ROMANOVA, 2016) e (STOYAN; ZLOTNIK; CHUGAY, 2012) e um pouco menores na maioria das instâncias ao comparar com (LIAO *et al.*, 2016) e (KALLRATH, 2009). Um modelo de programação não-linear inteiro misto em que uma parábola garante que dois itens estejam separados também é proposto neste trabalho.

As contribuições desta tese são as seguintes: (1) um modelo para o problema de empacotamento de polígonos (convexos ou não-convexos) e círculos em faixa que permite rotações livres dos itens; (2) um modelo para o problema de empacotamento de polígonos (convexos ou não-convexos) e círculos em uma envoltória convexa que permite rotações livres dos itens; (3) criamos restrições simples de não-sobreposição entre os itens usando retas separadoras; (4) apresentamos parábolas separadoras como uma nova ferramenta para garantir que dois polígonos não-convexos, sem necessidade de decomposição em polígonos convexos, estejam separados.

Neste trabalho, os capítulos estão organizados da seguinte maneira: no Capítulo 2, apresentamos uma revisão bibliográfica sobre IPP e problemas de empacotamento de itens irregulares e círculos simultaneamente. No Capítulo 3, apresentamos dois modelos para problemas de empacotamento de itens irregulares e círculos, representados por polígonos, com rotação livre, que devem ser posicionados em uma placa. No primeiro problema, a placa tem largura fixa e comprimento infinito, e o objetivo do modelo é minimizar o comprimento utilizado. No segundo problema, a placa é uma envoltória retangular, e o objetivo do modelo é minimizar a área da envoltória. Retas separadoras são usadas para garantir a não-sobreposição dos itens nestes modelos. No Capítulo 4, apresentamos os métodos de resolução utilizados para resolver os modelos apresentados no Capítulo 3 e os resultados obtidos com estes. No Capítulo 5, um estudo de parábolas separadoras para garantir a não-sobreposição de itens é apresentado. Finalmente, no Capítulo 6, conclusões e sugestões de trabalhos futuros são discutidas.

PROBLEMA DE EMPACOTAMENTO DE ITENS IRREGULARES

Neste capítulo apresentamos uma revisão bibliográfica dos problemas de empacotamento de itens irregulares ou *Irregular Packing Problems* (IPP). Os IPP são problemas de corte e empacotamento que buscam determinar um arranjo ótimo de itens, em que pelo menos um é irregular, sem sobreposição, em uma região (placa). O objetivo é minimizar custos, seja maximizando a ocupação de espaços ou minimizando desperdícios. Porém, considerações práticas podem levar a modelos com diferentes restrições e diferentes objetivos.

Um item é considerado irregular se precisa, no mínimo, de três parâmetros para identificá-lo. Por exemplo, um círculo precisa apenas um único parâmetro, o raio, e um retângulo precisa de dois parâmetros, o seu comprimento e sua largura. As regiões (placas) são geralmente polígonos simples e em alguns casos, polígonos que contêm buracos.

Na prática, os IPP estão presentes em diversos ramos da indústria, em que se necessita cortar ou encaixar múltiplos itens irregulares, como, por exemplo, na indústria têxtil, de móveis e de calçados. A Figura 1 mostra um exemplo de leiaute na indústria têxtil.

2.1 Geometria dos problemas de empacotamento de itens irregulares

Dado que os IPP lidam com itens irregulares (não retangulares, não circulares), é muito importante para a eficácia de qualquer técnica de resolução do problema a representação e manipulação adequada da geometria dos itens. Na Seção 2.1.1 apresentamos as diferentes formas como os itens têm sido representados na literatura e como são representados neste trabalho. Na Seção 2.1.2 apresentamos as ferramentas encontradas na literatura para garantir a não sobreposição dos itens, dependendo da representação dos itens.

Figura 1 – Exemplo de leiaute na indústria têxtil. Fonte: (BENNELL; OLIVEIRA, 2009).

2.1.1 Representação dos itens

Em alguns trabalhos a placa sempre é dividida em áreas discretas e esquemas de codificação são utilizados. Em alguns destes esquemas, como os apresentados em (SEGENREICH; BRAGA, 1986; OLIVEIRA; FERREIRA, 1993), os espaços vazios na placa, sem alocação de itens, são denotados por 0 e utilizam números iguais ou maiores do que 1 para codificar um item (veja um exemplo na Figura 2). Em outros, como em (BABU; BABU, 2001), atribuem 0 ao interior e à fronteira dos itens. Fora dos itens, é atribuído um número maior que 0, sendo 1 para o pixel imediatamente à direita do pixel 0 e, cumulativamente, adiciona 1 por movimento da direita à esquerda. Como estes esquemas não podem representar exatamente a irregularidade dos itens; uma alternativa é representar os itens por polígonos.

Figura 2 – Representação 0-1. Fonte: (BENNELL; OLIVEIRA, 2008).

Na maioria dos algoritmos que resolvem IPP, os itens são polígonos (não regulares) e a representação do item é o mesmo polígono (veja um exemplo na Figura 3, na que é apresentado um item da instância *Swim* que pode ser encontrada em (OLIVEIRA; GOMES; FERREIRA, 2000)). Outros, simplesmente representam os itens por polígonos mais regulares, faltando apenas por representar exatamente, quando existentes, as bordas curvas. O retângulo é o item mais usado para a representação. Quando representamos os itens como polígonos, a quantidade de informação é proporcional ao número de vértices e não depende do tamanho dos itens ou do leiaute.

Figura 3 – Item da instância *Swim*.

Quando os itens incluem bordas curvas, é comum aproximá-los por um polígono envolvente, em que uma série de tangentes à curva formam as bordas poligonais (veja um exemplo de um círculo aproximado por um polígono na Figura 4). Alguns trabalhos permitem que as bordas curvas sejam representadas em sua forma original e, para isto, os itens são representados como a união ou intersecção de objetos chamados primários, ou seja, círculos, retângulos, polígonos regulares, polígonos convexos e o complemento destas formas (STOYAN *et al.*, 2001; STOYAN *et al.*, 2004; STOYAN; PANKRATOV; ROMANOVA, 2016) (veja Figuras 5 e 6).

Figura 4 – Círculo aproximado por um polígono.

Em outros trabalhos, a representação dos itens é feita pela inscrição ou cobertura de alguns círculos em cada item (JONES, 2013; ROCHA *et al.*, 2015), veja um exemplo na Figura 7.

Neste trabalho, representaremos os itens irregulares como polígonos convexos ou como a união de vários polígonos convexos. Cada círculo é representado por ele mesmo.

2.1.2 Verificação de não-sobreposição

A restrição principal dos IPP é a não-sobreposição dos itens entre si, entretanto, é muito complexo para um programa computacional saber se dois itens estão sobrepostos, estão tocando-se ou estão separados. Existem técnicas conhecidas na literatura para resolver esta questão (BENNEL; OLIVEIRA, 2008), entre estes, pode-se citar: rasterização, trigonometria direta,

Figura 5 – a) Objetos primários. b) Objetos básicos. Fonte: (STOYAN; PANKRATOV; ROMANOVA, 2016).

Figura 6 – Decomposição de itens em objetos básicos. Fonte: (STOYAN; PANKRATOV; ROMANOVA, 2016).

Figura 7 – a) Cobertura de círculos. Fonte: (ROCHA *et al.*, 2015) b) Inscrição de círculos. Fonte: (JONES, 2013)

No-Fit Polygon (NFP) e função Φ . Um conceito mais recente que também lida com esta questão é o *No-Fit Raster* (NFR) (MUNDIM; ANDRETTA; QUEIROZ, 2017).

Nos casos em que a região é dividida em áreas discretas, a rasterização é utilizada para determinar a sobreposição, verificando apenas as células da grade.

A vantagem da rasterização é que calcular quanto o item deve ser deslocado, a fim de eliminar infactibilidade ou onde alocar dois itens de modo que eles estejam em contato, é apenas uma questão de contagem de células na direção desejada.

Ao representar os itens por polígonos, a rasterização não pode ser utilizada, já que a factibilidade ou a qualidade de posicionamento não está implícita pela representação e, como resultado, outras técnicas de avaliação devem ser aplicadas, como por exemplo trigonometria direta ou NFP. Outra desvantagem da rasterização é que esta consome muita memória no armazenamento da matriz, enquanto que o gasto em outras técnicas, como trigonometria direta ou NFP, é menor.

Na trigonometria direta, os itens são representados como uma série de vértices e arestas. Usando esta técnica, é preciso avaliar o cruzamento das arestas para verificar se há ou não sobreposição entre os itens. Para isto, existem testes bem conhecidos. Embora esses testes sejam computacionalmente mais complexos em comparação com os métodos de rasterização, o tempo de execução não pode ser comparado diretamente. Para o método de rasterização, o tempo para verificar a viabilidade é quadrático no tamanho da grade, enquanto que para a trigonometria direta é exponencial no número de arestas dos polígonos.

A Figura 8 ilustra exemplos das posições relativas de dois polígonos. As figuras 8c e 8d são os dois exemplos nos quais há sobreposição. A primeira pode ser identificada através da análise direta das arestas e a outra identificando que um ou mais vértices de um polígono estão dentro do outro.

Em (BENNELL; OLIVEIRA, 2008) são sugeridos os seguintes testes, de forma hierárquica, para verificar a sobreposição de dois polígonos utilizando trigonometria direta.

Teste 1: Os retângulos envolventes dos polígonos se sobrepõem?

Não: os polígonos não se sobrepõem (Figura 8a).

Sim: aplicar Teste 2.

Teste 2: Para cada par de arestas de diferentes polígonos, seus respectivos retângulos envolventes se sobrepõem?

Não para todas as arestas: os polígonos não se sobrepõem (Figura 8b).

Sim para algumas arestas: para estas arestas, aplicar Teste 3.

Teste 3: Para cada par de arestas de diferentes polígonos, a análise das arestas indica intersecção?

Não para todas as arestas: aplicar Teste 4.

Sim para um par de arestas: como um par de arestas se cruza, os polígonos se sobrepõem

Figura 8 – Exemplo de posição relativa de dois polígonos. Fonte: (BENNEL; OLIVEIRA, 2008).

(Figura 8c).

Teste 4: Para cada polígono, algum vértice se encontra dentro do outro polígono?

Não para ambos polígonos: os polígonos não se sobrepõem (Figura 8e).

Sim para algum polígono: os polígonos se sobrepõem (Figura 8d).

Verificar se dois retângulos se sobrepõem é trivial, já que é suficiente comparar as coordenadas dos lados dos retângulos. Em (PREPARATA; SHAMOS, 1985) foi descrito um teste de ponto de inserção simples, que pode ser usado como Teste 4. Para o Teste 3, (MAHADEVAN, 1984) descreve como a posição relativa de duas arestas pode ser determinada usando a função-D (*D-function*). A função-D, proposta por (KONOPASEK, 1981), definida a seguir,

$$D_{ABP} = ((X_A - X_B)(Y_A - Y_P) - (Y_A - Y_B)(X_A - X_P)),$$

fornece a posição relativa de um ponto P , com coordenadas (X_P, Y_P) em relação à aresta orientada AB (Figura 9), que vai do ponto (X_A, Y_A) ao ponto (X_B, Y_B) . A função-D é baseada na equação da distância de um ponto a uma reta. Se $D_{ABP} > 0$, o ponto P está à esquerda da aresta AB ; se $D_{ABP} < 0$, o ponto P está à direita da aresta AB ; se $D_{ABP} = 0$, o ponto P está na aresta AB . Se supõe que a origem do sistema de coordenadas, em que as coordenadas (X_i, Y_i) são definidas, é o canto inferior-esquerdo, ou seja, a coordenada x aumenta para a direita e a coordenada y aumenta para cima.

A vantagem da função-D é que ela oferece uma abordagem precisa, no sentido de que os polígonos são exatamente representados. No entanto, do ponto de vista do esforço computacional, os cálculos devem ser realizados com unidades de ponto flutuante, que é muito mais lento do que tratar com posições de memória e somar números inteiros (rasterização). Além disso, cada

Figura 9 – Interpretação da função-D. Fonte: (BENNELL; OLIVEIRA, 2008).

vez que a posição de um polígono é alterada, a viabilidade da posição deve ser verificada. Como resultado, todos estes cálculos têm de ser repetidos a partir do início durante a busca do arranjo ótimo. Assim, esta abordagem não é interessante para algoritmos que resolvem IPP baseados em qualquer processo de busca iterativa, uma vez que o tempo gasto em avaliar a sobreposição pode restringir o tempo e, por conseguinte, o tempo da busca.

A ferramenta geométrica mais utilizada para verificar se dois polígonos estão sobrepostos, tocando-se ou separados, é o NFP. Em essência, o NFP é um polígono resultante dos mesmos dois polígonos a comparar.

O NFP de dois polígonos A e B , denotado por NFP_{AB} , é o polígono resultante de uma operação de deslizamento entre A e B , em que cada um tem um papel específico na operação. Ambos os polígonos tem orientação fixa. A tem posição fixa, em que se supõe que a origem é $(0,0)$, e B é um polígono que tem um ponto de referência e se move em torno do perímetro de A . O NFP_{AB} é obtido traçando-se o movimento do ponto de referência do polígono B ao redor de A . O movimento é feito de modo que A e B estão sempre se tocando, mas nunca se sobrepondo. O conjunto de pontos gerados forma um caminho fechado que é o NFP_{AB} . A Figura 10 mostra este deslizamento do polígono B sobre o polígono A e o NFP resultante. A orientação das arestas do polígono A (polígono fixo) é anti-horário e do polígono B (polígono orbital) é horário.

Claramente, enquanto A é fixado em $(0,0)$, se B é posicionado de modo que seu ponto de referência está dentro de NFP_{AB} , então A e B estão se sobrepondo; se o ponto de referência de B está na fronteira de NFP_{AB} , então A e B estão se tocando. Assim, o interior do NFP_{AB} representa todas as posições em que, se o ponto de referência de B é alocado, há intersecção de A e B . Se a origem do polígono A não é restrita em $(0,0)$, por exemplo, se é movida à posição (x,y) , então a posição do ponto de referência de B deve ser mudada para $(-x, -y)$ antes de testar a sua posição relativa com NFP_{AB} . Portanto, podemos verificar se dois polígonos se sobrepõem verificando se o resultado da subtração da posição de origem de A do ponto de referência de B está dentro do NFP_{AB} .

Uma das vantagens desta técnica é que a geração destes polígonos é feita uma única vez, antes do processamento da resolução. Assim, é mais eficiente do que a trigonometria direta, especialmente quando é utilizada uma busca iterativa, e também, compartilha o benefício de precisão que se tem ao usar as arestas originais do polígono. No entanto, a desvantagem

Figura 10 – Geração do NFP_{AB} . a) Movimento do polígono B ao redor do polígono A para formar o NFP. b) Orientação dos polígonos A e B. c) Ordem das direções de A e B. d) NFP_{AB} . Fonte: (BENNELL; OLIVEIRA, 2008)

significativa desta abordagem deve-se às dificuldades em desenvolver um gerador robusto de NFPs para polígonos não-convexos gerais.

Quando os itens incluem bordas curvas e estas são representados na sua forma original, as funções Φ foram usadas para garantir a não-sobreposição dos itens. As funções Φ foram concebidas e aplicadas por (STOYAN *et al.*, 2001; STOYAN *et al.*, 2004; STOYAN; PANKRATOV; ROMANOVA, 2016). A função Φ é uma expressão matemática que representa a posição relativa de dois itens. Especificamente, o valor da função Φ é maior do que 0 se os itens estão separados, igual a 0 se as suas fronteiras estão se tocando e menor do que 0 se sobrepõem-se. Quando a função Φ é normalizada, seu valor é a distância Euclideana entre os dois itens; caso contrário, é uma estimativa desta distância. (STOYAN *et al.*, 2001) obteve a função Φ para objetos primários, ou seja, círculos, retângulos, polígonos regulares, polígonos convexos e o complemento destas formas. A obtenção das funções é baseada em trigonometria e é feita manualmente, veja um exemplo na Figura 11.

A falta de um processo algorítmico para gerar funções Φ para itens arbitrários e que rotacionem livremente pode explicar porque essa abordagem não tem sido mais amplamente adotada.

Recentemente, um novo conceito, NFR, foi introduzido em (MUNDIM; ANDRETTA;

Figura 11 – Exemplo da função Φ para dois polígonos convexos. Fonte: (BENNELL *et al.*, 2010)

QUEIROZ, 2017). NFR pode ser usado para verificar a sobreposição entre itens com quaisquer formas genéricas. Nesta ferramenta, uma grade discreta de pontos é usada para representar a placa e NFR consiste em todos os pontos da grade em que o ponto de referência do polígono pode ser posicionado sem gerar sobreposição com os itens que foram empacotados anteriormente. O cálculo do NFR pode ser feito usando trigonometria direta, NFP e/ou funções Φ . Veja um exemplo de NFR na Figura 12.

Figura 12 – Exemplo de NFR para um polígono e um círculo. Fonte: (MUNDIM; ANDRETTA; QUEIROZ, 2017)

A vantagem desta técnica é que para verificar a não-sobreposição dos itens não necessita envolver equações não-lineares difíceis como na função Φ e nas técnicas de trigonometria direta. A desvantagem desta ferramenta é o tamanho da grade adotada, já que deve procurar ter a distância entre dois pontos subsequentes tão pequena quanto possível.

Nos trabalhos em que a representação dos itens é feita pela inscrição ou cobertura de alguns círculos, modelos matemáticos foram criados, em que as restrições de não-sobreposição dos itens são relaxadas e substituídas por restrições de não-sobreposição dos círculos (veja um exemplo na Figura 13).

Figura 13 – Aproximação da restrição de não-sobreposição usando círculos inscritos. Fonte: (JONES, 2013)

Unicamente as funções Φ e a cobertura ou inscrição de círculos permitem a rotação livre dos itens. Neste trabalho apresentamos outras duas alternativas para lidar com as restrições de não-sobreposição e que permitem rotações livres, as retas e as parábolas separadoras.

Depois de revisar as técnicas conhecidas na literatura para tratar a questão de não-sobreposição dos itens e a forma como os itens são representados, a seguir, estudaremos algumas das abordagens utilizadas para a resolução dos IPP.

2.2 Abordagens para a resolução dos problemas de empacotamento de itens irregulares

Várias abordagens têm sido propostas para a resolução dos IPP, nesta seção apresentamos um breve resumo dos principais trabalhos que apresentam métodos de resolução para os IPP, estes métodos contornam o problema desde diferentes perspectivas e portanto apresentam algumas variações, como por exemplo, de dimensões abertas ou fechadas, de várias ou uma única placa, com ou sem defeitos, que permitem ou não rotações livres, entre outros. Dividimos as abordagens em heurísticas (Seção 2.2.1), exatas (Seção 2.2.2) e math-heurísticas (Seção 2.2.3).

2.2.1 Heurísticas

Os métodos de resolução são baseados, predominantemente, em heurísticas (ALBANO; SAPUPPO, 1980; MARQUES; BISPO; SENTIEIRO, 1991; OLIVEIRA; FERREIRA, 1993; GOMES; OLIVEIRA, 2002; EGEBLAD; NIELSEN; ODGAARD, 2007; NIELSEN, 2007; ABEYSOORIYA; BENNELL; MARTINEZ-SYKORA, 2018), devido ao problema ser NP-completo (FOWLER; PATERSON; TANIMOTO, 1981).

As heurísticas usadas para a resolução destes problemas podem trabalhar com soluções parciais, construindo o leiaute final item a item (heurísticas construtivas), ou com soluções completas, em que mudanças são feitas a fim de encontrar melhorias.

2.2.1.1 Heurísticas construtivas

Uma solução parcial é o leiaute dado por uma heurística construtiva antes de encontrar o leiaute final. Heurísticas construtivas muitas vezes podem produzir soluções de qualidade razoável, com baixo custo computacional. Além disso, a viabilidade é garantida, uma vez que cada item é alocado em uma posição factível na placa e não é movido. A qualidade da solução nas heurísticas construtivas depende de duas características principais: primeiro, a ordem em que são posicionados os itens; segundo, a regra de posicionamento adotada. Adicionalmente, o desempenho da heurística pode ser melhorado, permitindo sequenciamento dinâmico dos itens na evolução da solução parcial e/ou *backtracking*. Outra questão que deve ser considerada em uma heurística construtiva é a sua capacidade de preencher buracos. Os buracos podem estar nos itens ou podem ser porções da placa, cercados por itens já alocados.

Regra de posicionamento

Independentemente de qual item estamos posicionando, uma posição na placa deve ser encontrada. A regra de posicionamento mais popular e amplamente utilizada é inferior esquerda ou *Bottom-Left* (BL), em que se escolhe uma sequência para alocar os itens no ponto da placa mais abaixo e à esquerda possível. Esta regra pode ser igualmente aplicada a outras direções, como superior direita ou *Top Right* (TR). O item na sua posição final é obrigado a estar dentro da placa e não deve se sobrepor a itens previamente alocados. Esta estratégia de posicionamento foi aplicada pela primeira vez a IPP por (ART, 1996) e ainda os mesmos princípios são aplicados em mais recentes abordagens.

(OLIVEIRA; GOMES; FERREIRA, 2000) sugerem uma regra de posicionamento alternativa ao BL. Neste caso, as posições dos itens não são fixas na placa. Em vez disso, um leiaute flutuante é utilizado, em que as posições relativas dos itens são fixas e a largura da solução parcial é restrita a ser menor ou igual à largura da placa. A solução parcial é descrita por sua fronteira, na qual é adicionado o próximo item (Figura 14). (OLIVEIRA; GOMES; FERREIRA,

Figura 14 – Regra de posicionamento alternativa ao BL. Fonte: (BENNEL; OLIVEIRA, 2009).

2000) investigam três regras de posicionamento para o crescimento da solução, com o objetivo de manter o leiaute mais compacto possível ao tentar evitar um aumento no comprimento. Estas são:

- a minimização da área do retângulo envolvente da solução parcial recém-gerada;
- a minimização do comprimento do retângulo envolvente da solução parcial recém-gerada;
- a maximização da sobreposição entre o retângulo envolvente da solução parcial atual e o retângulo envolvente do item a ser posicionado, sem permitir a sobreposição entre os próprios itens.

Cada item adicionado à solução parcial é fundido com os itens já posicionados e eventuais buracos entre os itens são considerados como desperdícios.

Figura 15 – Mecanismo para definição de novos ângulos de orientação. Fonte: (ABEYSOORIYA; BENNEL; MARTINEZ-SYKORA, 2018).

Recentemente, em (ABEYSOORIYA; BENNELL; MARTINEZ-SYKORA, 2018), foi proposta uma heurística construtiva, baseada na heurística que foi proposta originalmente em (OLIVEIRA; GOMES; FERREIRA, 2000) e depois melhorada em (BENNELL; SONG, 2010), na qual é permitido que os itens adotem qualquer ângulo de orientação. Foram avaliadas três regras de posicionamento, BL, posicionamento de comprimento mínimo e posicionamento de máxima utilização, para determinar a posição de cada item. Para garantir a não-sobreposição dos itens, os autores utilizaram NFP, porém como qualquer orientação é permitida, os NFPs são calculados durante o algoritmo. Para alocar um item, o algoritmo gera o NFP do próximo item a ser posicionado com um único polígono que representa todos os outros itens que já foram posicionados. Este polígono é criado juntando sequencialmente todos os itens posicionados. Um mecanismo para identificar um subconjunto de ângulos promissores de acordo com o arranjo dos itens na solução parcial atual foi elaborado nesse trabalho. Neste mecanismo, um item é posicionado temporariamente com um ângulo de 0° , 90° , 180° ou 270° , e cada ponto ou borda de toque define dois novos ângulos de orientação (veja Figura 15).

Ordem de posicionamento

A ordem em que os itens são posicionados na placa tem uma influência significativa sobre a qualidade da solução. Definir esta ordem é difícil, pois formas irregulares não têm características comuns que podem dar uma visão sobre como os itens podem se encaixar melhor. Ao considerar a ordem, é necessário decidir entre a aleatorização, ordenação de acordo com uma regra fixa, seleção dinâmica e se o *backtracking* é permitido.

A abordagem mais simples é a aleatorização. É usualmente utilizada para gerar soluções iniciais para algoritmos que trabalham com soluções completas. Tal abordagem produz soluções altamente variáveis e só deve ser considerada sob várias execuções. Se a abordagem para a resolução consiste na construção de uma única solução usando uma heurística construtiva, podemos empregar uma ordem com melhor raciocínio que a aleatorização. Sequências pré-definidas podem ser utilizadas. (DOWSLAND; VAID; DOWSLAND, 2002) usam oito ordenações estáticas, ilustradas na Figura 16. Todas estas regras têm em comum a estratégia de tentar alocar primeiro os itens “difíceis”. Dependendo dos critérios, os itens “difíceis” podem ser os maiores, os mais compridos, os mais irregulares ou simplesmente aqueles que estão em quantidades maiores.

A seleção dinâmica permite que todos os tipos de itens que estejam disponíveis sejam o próximo item a adicionar à solução parcial. Um tipo de item só é excluído se todos os itens desse tipo já foram posicionados. Cada solução parcial é avaliada usando um critério dado e o item que gera a melhor solução parcial é escolhido.

Construir uma solução item por item é uma maneira rápida de gerar soluções razoáveis. No entanto, a fim de encontrar soluções com melhor qualidade, são necessários métodos mais sofisticados. Isto pode ser conseguido trabalhando com soluções completas que são melhoradas

Figura 16 – Métricas usadas por (DOWSLAND; VAID; DOWSLAND, 2002) para determinar a ordem estática dos itens. Fonte: adaptado de (BENNEL; OLIVEIRA, 2009).

de forma iterativa.

2.2.1.2 Buscas locais

Como os IPP são problemas combinatórios e NP-completos, abordagens do tipo busca local são um conjunto de candidatos óbvios para sua resolução. As principais meta-heurísticas foram aplicadas a IPP, entre estas Busca Tabú (BENNEL; DOWSLAND, 1999), *Simulated Annealing* (SA) (OLIVEIRA; FERREIRA, 1993; HECKMANN; T.LENGAUER, 1995) e Algoritmos Genéticos (JAKOBS, 1996; BABU; BABU, 2001).

Há muitas características a considerar quando se trabalha com busca local, entre estas, a função objetivo, o conjunto de soluções factíveis, os movimentos na vizinhança e, sobretudo, o mecanismo de busca que controla a geração e aceitação de movimentos. No entanto, a forma como se representa o problema é a característica que define a escolha da estratégia de busca. As duas principais formas de representar o problema são: como uma sequência de itens que é decodificada por meio de uma regra de posicionamento e como o próprio leiaute.

Representação do problema como uma sequência de itens

Dada a sequência de itens, gerar o leiaute é possível e necessário para avaliar a qualidade da solução. Isso tende a incorrer em um maior custo computacional do que a avaliação de um movimento dentro do próprio leiaute. Além disso, a qualidade da solução está sujeita às limitações da regra de posicionamento adotado. Um dos principais benefícios é que a regra de posicionamento garante factibilidade.

Figura 17 – Leiautes resultantes de uma regra de posicionamento BL após movimentos na vizinhança. Fonte: (BENNEL; OLIVEIRA, 2009).

Troca e inserção são as estratégias de busca na vizinhança mais comuns nas heurísticas de busca local. Na troca, a posição de dois itens na sequência é intercambiada e, na inserção, um item é removido de sua posição atual e inserido em outro lugar na sequência. Para os problemas de empacotamento, um outro movimento é acrescentado, a troca de orientação do item. Estes três movimentos são ilustrados na Figura 17.

No exemplo da Figura 17, cada movimento significa mudanças no leiaute resultante. Conceitualmente, a troca é mais suscetível a gerar pequenas mudanças no leiaute, já que a substituição de um item por outro deixa todos os outros itens da sequência na mesma posição, enquanto que a inserção vai perturbar o leiaute de todos os itens que estão após a inserção ou remoção de um item em uma posição diferente da sequência. Assim, inserção pode fornecer uma maior diversificação. A troca de orientação pode ser utilizada quando mais do que uma orientação é permitida e o algoritmo não tem um mecanismo para testar diferentes orientações. Este movimento é utilizado em conjunto com a troca e a inserção e, provavelmente, favorecerá estes movimentos.

Representação do problema pelo próprio leiaute

Implementações que trabalham com o próprio leiaute muitas vezes permitem sobreposição entre os itens, a fim de proporcionar outras possibilidades para a busca. Assim, convergir para uma solução factível pode ser problemático. Alguns autores propõem “pós-otimização” ou processos de reparação para eliminar a sobreposição, mas isto, muitas vezes, afeta consideravelmente a qualidade da solução (BENNEL; OLIVEIRA, 2009).

Podemos caracterizar as abordagens para resolução dos IPP, ao se trabalhar com soluções completas representadas pelo próprio leiaute, dependendo do espaço solução, que pode ser contínuo (infinito) ou discreto (finito). A placa, no caso mais geral, é uma área contínua, e

portanto, o espaço de solução para métodos que buscam sobre o próprio leiaute é contínuo. Métodos que discretizam o espaço solução buscam gerar procedimentos de busca mais rápidos, enquanto tentam não eliminar a solução ótima global, mantendo o ponto de posicionamento mais promissor. Um método comum é impor uma grade sobre a placa e só pesquisar sobre os pontos da grade. Por isso, o espaço solução contém somente aquelas soluções que alocam o ponto de referência de cada item em um ponto de grade.

A vizinhança, agora, é definida levando em consideração se o método permite ou não sobreposição e se o espaço solução é contínuo ou discreto. Pesquisadores usam duas estruturas principais. A primeira é baseada em mover um único item ou um conjunto limitado de itens. O movimento de um único item pode ser caracterizado como um movimento de inserção, em que um item é removido de sua localização atual e inserido em um novo local da placa. Se mais de um item é movido, pode-se caracterizar um movimento de troca, em que as posições de dois ou mais itens são intercambiadas. Outros tipos de movimentos são usualmente derivados da natureza geométrica dos IPP, tais como rotação e simetria.

A segunda estrutura é conhecida como compactação e separação, e permite mover todos os itens ao mesmo tempo quando se deslocam de uma solução para outra, seja compactando ou separando os itens. Embora esta seja uma abordagem poderosa, só faz a busca em uma área local do espaço solução. Assim, outros mecanismos são utilizados quando é necessária uma pesquisa mais ampla.

Alguns pesquisadores que trabalharam com a primeira estrutura e que empregaram uma grade, relatam sucesso razoável (OLIVEIRA; FERREIRA, 1993; BENNELL; DOWSLAND, 2001). (GOMES; OLIVEIRA, 1999) estenderam o trabalho de (OLIVEIRA; FERREIRA, 1993), removendo a necessidade de uma grade de aproximação dos itens e da placa.

Os algoritmos que usam compactação e separação, resolvem o problema de uma maneira muito diferente de todas as abordagens anteriores. O problema é modelado e resolvido como um problema de programação linear, no qual a função objetivo consiste em minimizar comprimento e as restrições impedem ou removem a sobreposição, além de garantir a permanência dos itens dentro da placa (MILENKOVIC; DANIELS; LI, 1992; STOYAN; NOVOZHILOVA; KARTASHOV, 1996; BENNELL; DOWSLAND, 2001).

Por último, um sistema visual para o ISPP com rotação livre que visa minimizar o desperdício foi encontrado na literatura (LIAO *et al.*, 2016). Nesse artigo é usado o algoritmo de empacotamento de elástico para resolver o problema. Este algoritmo simula um elástico de borracha envolvendo os itens irregulares que devem ser empacotados. No artigo é descrito o processo de analogia da seguinte forma: um elástico é esticado em torno dos itens e a energia elástica é considerada alta inicialmente. Os itens são compactados porque o elástico esticado tende a diminuir a energia até que a força resultante seja zero. Para diminuir o comprimento da placa e obter uma alta densidade de empacotamento, a força resultante do elástico de borracha é aplicada nos itens quando transladados e rotacionados. Para fazer uma compactação adequada

sem interseção, a força do elástico de borracha, que é variável, é definida junto com um intervalo de tempo, também variável, de acordo com o alongamento do elástico a cada iteração. O leiaute é construído atendendo a não-sobreposição entre os itens e a não-sobreposição dos itens com a fronteira da placa. A relação entre a área total de todos os itens e a área da placa, denominada densidade de empacotamento, deve ser a maior possível.

2.2.2 Abordagens exatas

Abordagens exatas para a resolução dos IPP têm sido propostas em menor quantidade. Na literatura encontramos alguns modelos de programação linear inteira mista, assim como alguns modelos de programação não-linear.

2.2.2.1 Programação linear inteira mista

Modelos de Programação Linear Inteira Mista (PLIM) têm sido propostos para representar os IPP. A seguir, apresentamos um breve resumo dos trabalhos em que modelos de PLIM têm sido propostos.

- O modelo proposto em (FISCHETTI; LUZZI, 2009) utiliza variáveis contínuas para definir o posicionamento dos itens. Definem o conceito de fatias associadas ao complemento de um NFP e utiliza variáveis binárias para assegurar a não-sobreposição entre dois itens.
- Em (ALVAREZ-VALDES; MARTINEZ; TAMARIT, 2013) é feita uma revisão do modelo proposto em (FISCHETTI; LUZZI, 2009). Além disso, é proposto um método *branch-and-bound* para a resolução do problema. Neste artigo, o NFP de itens não-convexos é uma região que pode ser um polígono junto com, se necessário, alguns pontos, alguns segmentos ou outros polígonos.
- No modelo proposto em (TOLEDO *et al.*, 2013), a placa é representada por uma malha de pontos. Cada item é definido por um conjunto de vértices, sendo um deles o ponto de referência. Os pontos de referência dos itens só podem ser alocados sobre os pontos da malha. No modelo é usado o NFP para garantir a não-sobreposição dos itens.
- Em (LEAO *et al.*, 2016) foi proposto um modelo semi-contínuo no qual os itens podem ser alocados continuamente ao longo do eixo dos x e em posições discretas ao longo do eixo dos y , isto é, o ponto de referência dos itens pode ser alocado somente em linhas horizontais da placa. Estas linhas são paralelas e estão igualmente distribuídas, isto é, a distância entre as linhas no eixo dos y é um múltiplo constante. Esta distância deve ser cuidadosamente definida, uma vez que a qualidade da solução depende disso. No modelo é usado o NFP para garantir a não-sobreposição dos itens.

- Dois modelos robustos para um IPP com itens convexos ou não-convexos foram propostos em (CHERRI *et al.*, 2016). No primeiro modelo (DTM) foi usada trigonometria direta e no segundo modelo (NFP-CM) foi usado o NFP para garantir a não-sobreposição dos itens.
- Um grande número de soluções simétricas podem ser geradas com o modelo NFP-CM proposto em (CHERRI *et al.*, 2016). Em (RODRIGUES; CHERRI; MUNDIM, 2017) restrições de ruptura de simetria são propostas para melhorar este modelo.

2.2.2.2 Programação em lógica com restrições

Em (CARRAVILLA; RIBEIRO; OLIVEIRA, 2003) foi apresentada uma abordagem baseada em enumeração implícita e programação de lógica de restrição para a resolução de um IPP. Neste artigo, os itens podem ser convexos e não-convexos e é usado o NFP para garantir a não-sobreposição dos itens.

2.2.2.3 Programação não-linear

Modelos de programação não-linear também têm sido propostos para representar os IPP. Em alguns destes modelos a rotação livre é permitida. Um breve resumo destes modelos é apresentado a seguir.

- Em (CHERNOV; STOYAN; ROMANOVA, 2010) foi apresentado um modelo para um ISPP e algoritmos práticos para resolver este problema. Neste artigo, é mostrado como as funções Φ e a programação matemática melhoram o desempenho dos algoritmos de corte. As funções Φ apresentadas nestes artigos podem ser aplicadas a itens convexos e não-convexos, e a itens que podem ter furos. Os itens são *phi-objects*, que são objetos gerais de duas e três dimensões. Para resolver o problema, (CHERNOV; STOYAN; ROMANOVA, 2010) aplicaram uma modificação do método Zoutendijk de direções factíveis (ZOUTENDIJK, 1960; ZOUTENDIJK, 1970) combinado com o conceito de desigualdades ε -active (STOYAN; CHUGAY, 2008).
- Em (JONES, 2013) é apresentado um algoritmo exato para resolver o IPP. Neste artigo, a placa e os itens podem ser polígonos não-convexos. Além disso, os itens podem ter furos. Os itens podem ser transladados e rotacionados em uma placa com áreas proibidas. A ideia-chave é inscrever círculos em cada item e, em seguida, relaxar as restrições de não-sobreposição dos itens, substituindo-as por restrições de não-sobreposição dos círculos inscritos.
- Uma ideia parecida à apresentada em (JONES, 2013) é usada em (ROCHA *et al.*, 2015). Neste artigo, é apresentada uma abordagem para o ISPP que compacta os itens grandes em uma primeira fase, enquanto que em uma segunda fase, aloca os itens pequenos restantes entre os itens grandes, compactando todos os itens. Os itens rotacionam livremente.

- Em (STOYAN; PANKRATOV; ROMANOVA, 2016) foi fornecido um modelo de programação não-linear que emprega funções Φ . Neste artigo, os itens são delimitados por arcos circulares e/ou segmentos de linha; dois tipos de placas são considerados, retangulares e circulares. As restrições no modelo incluem: os itens estarem inteiramente contidos em uma placa, não-sobreposição de itens, distâncias mínimas permitidas entre objetos serem respeitadas, áreas proibidas não conterem itens e outras restrições tecnológicas específicas. Neste artigo é proposto um algoritmo para gerar soluções iniciais viáveis para itens irregulares, baseado em transformações homotéticas de círculos circunscritos em torno dos itens. Um algoritmo que gera sistemas de desigualdades com funções suaves que descrevem sub-regiões viáveis do problema também é proposto em (STOYAN; PANKRATOV; ROMANOVA, 2016). Para resolver o problema, desenvolveram um algoritmo de compactação para buscar soluções ótimas locais.

2.2.3 Abordagens exatas combinadas com heurísticas

Poucas math-heurísticas, abordagens exatas combinadas com heurísticas, têm sido desenvolvidas (GOMES; OLIVEIRA, 2006; CHERRI; CARRAVILLA; TOLEDO, 2016; MARTINEZ-SYKORA *et al.*, 2017). Estas permitem a computação de soluções de alta qualidade em tempos computacionais mais curtos. A seguir, apresentamos um breve resumo de cada um destes trabalhos

- Em (GOMES; OLIVEIRA, 2006) modelos de programação linear, derivados da aplicação de algoritmos de compactação e separação, são resolvidos para gerar vizinhanças durante o processo de busca e encontrar soluções ótimas locais. (GOMES; OLIVEIRA, 2006) usa a meta-heurística *Simuleating Annealing* (SA) para orientar a busca pelo espaço solução.
- Em (CHERRI; CARRAVILLA; TOLEDO, 2016) é apresentado um método que possui três fases: construção, melhoria e compactação, nas quais modelos exatos de PLIM da literatura são usados para resolver os sub-problemas. O modelo apresentado em (TOLEDO *et al.*, 2013) é usado na primeira e na segunda fase, combinado com uma heurística inspirada na abordagem *relax-and-fix* e uma heurística de vizinhança variável, respectivamente. Na terceira fase é usado um modelo baseado em um modelo apresentado em (GOMES; OLIVEIRA, 2006).
- Em (MARTINEZ-SYKORA *et al.*, 2017) é apresentado um procedimento construtivo que permite ou a rotação livre ou um conjunto finito de rotações dos itens, o que não é permitido nas outras abordagens que combinam heurísticas com métodos exatos. Este procedimento lida com a atribuição de itens às placas e o arranjo dos itens atribuídos a uma placa. Modelos de PLIM e heurísticas são usados para atribuir os itens às placas. Depois de atribuídos os itens, para cada rotação, de um conjunto de rotações promissoras previamente determinado, resolve-se um modelo de PLIM para determinar se o item encaixa na placa.

2.3 Problemas de empacotamento de itens irregulares e círculos simultaneamente

Nos processos de produção, como fabricação de vestuário, fabricação de calçados, fabricação de móveis, entre outros, é necessária a otimização de leiautes de itens de diferentes formas. Esses itens podem ser representados por círculos ou polígonos. Na literatura, existem muitos trabalhos que lidam com o empacotamento de polígonos, como visto até o momento neste capítulo. Também existem muitos trabalhos que lidam com o problema de empacotamento de círculos, por exemplo, (SZABÓ *et al.*, 2007) e (CASTILLO; KAMPAS; PINTÉR, 2008) entre muitos outros. Porém poucos trabalhos lidam com o problema de empacotamento de círculos e de polígonos simultaneamente (KALLRATH, 2009; STOYAN; ZLOTNIK; CHUGAY, 2012; MUNDIM; ANDRETTA; QUEIROZ, 2017).

Em (KALLRATH, 2009) e (STOYAN; ZLOTNIK; CHUGAY, 2012), foram propostos modelos de programação não-linear em que os itens podem ser transladados e rotacionados livremente. Em (MUNDIM; ANDRETTA; QUEIROZ, 2017), duas heurísticas baseadas em BL e o conceito de NFR foram propostos. Nestes, os itens não podem ser rotacionados livremente.

Em (KALLRATH, 2009) foi desenvolvido um modelo para dois casos de empacotamento de círculos e polígonos convexos. No primeiro caso, o objetivo é empacotar todos os itens em um retângulo de área mínima. No segundo caso, o objetivo é empacotar os itens em retângulos estocados de dimensões geométricas conhecidas. Para círculos, as únicas variáveis são as coordenadas do centro. Os polígonos são descritos pelas coordenadas de seus vértices, sendo todas variáveis. Assim, o uso de restrições que asseguram que as variáveis que representam os vértices reproduzam os objetos geométricos originais é necessário. Além destas restrições, o modelo possui restrições de não-sobreposição e restrições que garantem que todos os itens não excedam os limites da placa. A ideia de reta separadora é usada neste artigo para garantir a não-sobreposição dos itens.

Em (STOYAN; ZLOTNIK; CHUGAY, 2012) foi proposto um modelo para o problema de empacotamento de círculos e polígonos convexos e não-convexos em uma faixa com regiões proibidas. O objetivo do problema é maximizar a utilização do espaço. Os autores garantem que os itens não se sobrepõem usando a função Φ . A abordagem de solução inclui os seguintes métodos: um método de otimização por grupos de variáveis para construir soluções iniciais, uma modificação do método de direção viável Zoutendijk (ZOUTENDIJK, 1960; ZOUTENDIJK, 1970) para procurar mínimos locais e uma busca especial não-exaustiva de mínimos locais para encontrar uma aproximação de um mínimo global. A construção de soluções iniciais é feita pela aproximação por retângulos de uma mesma largura da placa, dos círculos e dos polígonos. A busca por mínimos locais é realizada em sequências de sub-regiões que são especificadas por sistemas de desigualdade não-lineares. A busca não-exaustiva de mínimos locais é feita pelo método de busca em vizinhança decremental (STOYAN; YASKOV, 2010), que baseia-se na

escolha de sub-regiões nas quais a probabilidade de encontrar melhores mínimos locais é maior do que em outras sub-regiões.

Em (MUNDIM; ANDRETTA; QUEIROZ, 2017) foram propostas duas heurísticas para o problema de empacotamento de círculos e polígonos não-convexos em uma placa bidimensional. O objetivo do problema é minimizar uma ou ambas as dimensões da placa. Para o desenvolvimento das heurísticas, a sequência em que os itens são empacotados deve ser determinada e isso é feito por um *Biased Random Key Genetic Algorithm* (BRKGA).

Depois de ter apresentado uma revisão bibliográfica sobre IPP, no seguinte capítulo, apresentamos dois modelos. O primeiro modelo é para um problema de empacotamento de itens irregulares e círculos em faixa que tem como objetivo minimizar o comprimento utilizado. O segundo modelo é para o problema de empacotamento de itens irregulares e círculos em uma envoltória retangular a ser desenhada com o objetivo de minimizar a área da envoltória. Retas separadoras são usadas para garantir a não-sobreposição dos itens nestes modelos.

MODELOS DE PROBLEMAS DE EMPACOTAMENTO DE ITENS IRREGULARES USANDO RETAS SEPARADORAS

Neste capítulo, apresentamos dois modelos de programação não-linear para problemas de empacotamentos de itens irregulares ou *Irregular Packing Problems* (IPP). Solucionar um IPP é posicionar n itens, neste caso círculos e polígonos convexos ou não-convexos que podem rotacionar livremente, obedecendo às restrições dos itens estarem inteiramente contidos na placa e às restrições de não-sobreposição, de modo a minimizar o espaço utilizado da placa. Para garantir a não-sobreposição usamos trigonometria direta, em particular usamos retas separadoras, uma técnica similar à utilizada em (KALLRATH, 2009), mas com um número significativamente menor de variáveis. Os detalhes das retas separadoras podem ser vistos na Seção 3.1.3.

A diferença entre os dois problemas está na placa e, portanto, a diferença entre os modelos esta associada à objetivo. No primeiro problema, a placa é uma faixa de largura fixa e de comprimento ilimitado. Este problema é denominado problema de empacotamento de itens irregulares em faixa ou *Irregular Strip Packing Problem* (ISPP). No segundo problema, a placa é uma envoltória retangular a ser produzida. Este problema é denominado problema de empacotamento de itens irregulares em uma envoltória retangular ou *Minimal Enclosure Problem* (MEP).

As coordenadas de um vértice l pertencente à partição de um polígono não-convexo P_i são dadas por

$$(x_{i,j}^l, y_{i,j}^l),$$

com $i = 1, \dots, n$, $j = 1, \dots, p_i$ e $l = 1, \dots, v_{i,j}$, sendo $v_{i,j}$ o número de vértices do polígono convexo $P_{i,j}$.

Podemos lidar com os problemas com os n polígonos não-convexos da mesma forma que lidamos com os problemas com N polígonos convexos, com $N = \sum_{i=1}^n p_i$. Unicamente devemos garantir que as translações e rotações sejam iguais para todos os polígonos pertencentes à partição de um polígono não-convexo.

Um ponto de referência é usado para representar um polígono que tenha sido submetido a translações e/ou rotações, pois, podemos escrever todos os outros vértices do polígono em termos deste ponto, como pode ser visto na Seção 3.1.2. Escolhemos o primeiro vértice como o ponto de referência de um polígono.

3.1.2 Vértices dos polígonos na forma geral

Usaremos a seguinte notação: (x_i^l, y_i^l) são as coordenadas do vértice l do polígono P_i na posição original e $(\bar{x}_i^l, \bar{y}_i^l)$ são as coordenadas deste vértice depois de ter sido submetido a translações e/ou rotações.

Na representação dos polígonos usada nos modelos, assumimos que $(x_i^1, y_i^1) = (0, 0)$ para todo i . Assim, se trasladamos e rotacionamos um polígono P_i em torno do seu ponto de referência, usando uma matriz de rotação, podemos calcular as coordenadas de qualquer vértice l em forma geral da seguinte maneira:

$$(\bar{x}_i^l, \bar{y}_i^l) = (x_i^l \cos \theta_i - y_i^l \sin \theta_i + \bar{x}_i^1, x_i^l \sin \theta_i + y_i^l \cos \theta_i + \bar{y}_i^1),$$

sendo θ o ângulo de rotação do polígono P_i . Consideramos que os ângulos positivos representam a rotação no sentido anti-horário.

Quando lidamos com polígonos não-convexos, devemos garantir que as translações e rotações sejam iguais para todos os polígonos pertencentes à partição do polígono não-convexo.

3.1.3 Reta separadora

Para garantir a não-sobreposição dos polígonos, fazemos uso de retas separadoras (veja a Figura 19). A ideia de retas separadoras foi apresentada em (KALLRATH, 2009), porém a representação que usamos no nosso modelo é diferente. Em (KALLRATH, 2009), um polígono é caracterizado por todos os seus vértices e implicitamente descrito por seu centro, as distâncias dos vértices para o centro e a orientação. A não-sobreposição de dois polígonos convexos é imposta pela condição de que todos os vértices de um polígono estão em um lado ou sobre a reta

separadora e todos os vértices do outro polígono estão no lado oposto ou sobre a reta separadora. Uma reta separadora é dada pela equação vetorial e envolve as seguintes variáveis: um ponto, o vetor diretor e um número real. Para garantir que todos os vértices de um polígono estejam de um lado da reta e todos os vértices do outro polígono estejam do outro lado da reta é usada a condição de que todos os vetores que conetam os vértices de um polígono com a reta separadora sejam paralelos ao vetor normal da reta separadora, e todos os vetores que conetam os vértices do outro polígono com a reta separadora sejam anti-paralelos ao vetor normal da reta separadora, quer dizer, que estejam na mesma direção mas com sentido oposto. Note que esta condição envolve outras variáveis, tais como, vetor normal à reta separadora, vetores que conetam os vértices com a reta separadora e distância dos vértices para a reta separadora. Para impor que um polígono não se sobreponha a um círculo, é exigido que todos os vetores que conetam os vértices do polígono com a reta separadora sejam paralelos ao vetor normal da reta separadora e o vetor que conecta o centro do círculo com a reta separadora seja anti-paralelo ao vetor normal da reta separadora. No modelo que é proposto em (KALLRATH, 2009), uma reta que separa um círculo de um polígono é sempre uma reta tangente ao círculo.

Nos modelos que estamos propondo neste trabalho, uma reta separadora é dada pela equação geral da reta e envolve unicamente três variáveis: as coordenadas de um ponto de referência e a orientação. A reta dada pela equação $y = c_{i,r}x + d_{i,r}$ separa dois polígonos P_i e P_r se

$$\begin{cases} \bar{y}_i^l - c_{i,r}\bar{x}_i^l - d_{i,r} \leq 0, & \forall l \in \{1, \dots, v_i\}, \\ \bar{y}_r^l - c_{i,r}\bar{x}_r^l - d_{i,r} \geq 0, & \forall l \in \{1, \dots, v_r\}, \end{cases} \quad (3.1)$$

ou

$$\begin{cases} \bar{y}_i^l - c_{i,r}\bar{x}_i^l - d_{i,r} \geq 0, & \forall l \in \{1, \dots, v_i\}, \\ \bar{y}_r^l - c_{i,r}\bar{x}_r^l - d_{i,r} \leq 0, & \forall l \in \{1, \dots, v_r\}. \end{cases} \quad (3.2)$$

Figura 19 – Reta separadora de dois polígonos convexos.

Para cada par de polígonos P_i e P_r , com $i, r \in \{1, 2, \dots, n\}$, usamos uma reta separadora. Quando estamos lidando com polígonos não-convexos, devemos ter retas separando cada par de polígonos $P_{i,j}$, $P_{r,s}$, pertencentes à partição de polígonos P_i , P_r , respectivamente, com $i \neq r$ (isto é, não temos retas separando os polígonos pertencentes à partição de um polígono não-convexo), $j \in 1, \dots, p_i$ e $s \in 1, \dots, p_r$, (veja Figura 20).

Figura 20 – Retas separadoras de dois polígonos, sendo um deles não-convexo.

Em seguida, apresentamos a forma geral de uma reta separadora que passa por dois pontos, digamos (x_k, y_k) e (x_{k+1}, y_{k+1}) :

$$(y - y_k)(x_{k+1} - x_k) - (y_{k+1} - y_k)(x - x_k) = 0.$$

Como os polígonos, as retas separadoras também podem rotacionar e se transladar, desde que continuem sendo retas separadoras. Se rotacionamos e transladamos uma reta, usando uma matriz de rotação, podemos reescrever o ponto $(\bar{x}_{k+1}, \bar{y}_{k+1})$, em função de $(\bar{x}_k, \bar{y}_k)$ (ao que chamamos de ponto de referência da reta separadora), assim

$$\begin{aligned} (\bar{x}_{k+1}, \bar{y}_{k+1}) &= (x_{k+1} - x_k) \cos \alpha_{i,j,r,s} + (y_k - y_{k+1}) \sin \alpha_{i,j,r,s} + \bar{x}_k, \\ &\quad (x_{k+1} - x_k) \sin \alpha_{i,j,r,s} + (y_{k+1} - y_k) \cos \alpha_{i,j,r,s} + \bar{y}_k, \end{aligned}$$

sendo $\alpha_{i,j,r,s}$ o ângulo de rotação da reta que separa os polígonos $P_{i,j}$ e $P_{r,s}$.

Substituindo $(\bar{x}_k, \bar{y}_k)$ e $(\bar{x}_{k+1}, \bar{y}_{k+1})$ na equação da reta separadora, podemos obter a seguinte equação:

$$\begin{aligned} (y - \bar{y}_k)[(x_{k+1} - x_k) \cos \alpha_{i,j,r,s} + (y_k - y_{k+1}) \sin \alpha_{i,j,r,s}] - \\ (x - \bar{x}_k)[(x_{k+1} - x_k) \sin \alpha_{i,j,r,s} + (y_{k+1} - y_k) \cos \alpha_{i,j,r,s}] = 0, \end{aligned}$$

que é a restrição que usamos nos nossos modelos.

3.1.4 Modelos matemáticos

Tanto os polígonos como as retas separadoras podem rotacionar, assim, os modelos matemáticos dos problemas apresentados aqui são não-lineares e não-convexos.

Lembre que, para que uma reta seja separadora de dois polígonos, esta deve satisfazer (3.1) ou (3.2). Aqui, sem perda de generalidade, para garantir a não-sobreposição dos polígonos, usaremos nos nossos modelos o conjunto de restrições (3.1) para cada par de polígonos $P_{i,j}$ e $P_{r,s}$, com

$$c_{i,j,r,s} = \frac{(x_{k+1} - x_k) \sin \alpha_{i,j,r,s} + (y_{k+1} - y_k) \cos \alpha_{i,j,r,s}}{(x_{k+1} - x_k) \cos \alpha_{i,j,r,s} + (y_k - y_{k+1}) \sin \alpha_{i,j,r,s}} \quad (3.3)$$

e

$$d_{i,j,r,s} = \bar{y}_k - c_{i,j,r,s} \bar{x}_k, \quad (3.4)$$

sendo (x_k, y_k) e (x_{k+1}, y_{k+1}) os dois vértices de um dos polígonos pelos quais passa a reta separadora e $\alpha_{i,j,r,s}$ o ângulo de rotação da reta que separa $P_{i,j}$ de $P_{r,s}$. Usamos a notação $(\bar{x}_k, \bar{y}_k)$ para o ponto de referência da reta separadora. Note que as coordenadas deste ponto de referência são também os valores dos parâmetros de translação, que, a partir de agora, escreveremos $(\bar{x}_{i,j,r,s}, \bar{y}_{i,j,r,s})$, para a reta que separa $P_{i,j}$ de $P_{r,s}$.

3.1.4.1 Modelo para o problema de empacotamento de itens irregulares em faixa

Nesta seção apresentamos um modelo para o problema de empacotamento de n itens irregulares em faixa de largura fixa e comprimento ilimitado, com o objetivo de minimizar o comprimento usado da faixa. A função objetivo é dada por:

$$W = \max\{\bar{x}_{i_l}\}, \quad l = 1, 2, \dots, v_i \text{ e } i = 1, 2, \dots, n, \quad (3.5)$$

com v_i o número de vértices do polígono P_i .

Neste caso, L é a largura (fixa) da faixa em que os polígonos são posicionados. Um modelo geral para o problema, que chamamos de *Model for a Irregular Strip Packing Problem* (MISSP), é dado por:

Minimizar W

$$\text{sujeito a } 0 \leq \bar{y}_{i,j}^l \leq L, \quad i = 1, \dots, n, \quad (3.6a)$$

$$j = 1, \dots, p_i,$$

$$l = 1, \dots, v_{i,j},$$

$$0 \leq \bar{x}_{i,j}^l \leq W, \quad i = 1, \dots, n, \quad (3.6b)$$

$$j = 1, \dots, p_i,$$

$$l = 1, \dots, v_{i,j},$$

$$\bar{y}_{i,j}^l - c_{i,j,r,s} \bar{x}_{i,j}^l - d_{i,j,r,s} \leq 0, \quad i = 1, \dots, n-1, \quad (3.6c)$$

$$r = i+1, \dots, n,$$

$$j = 1, \dots, p_i,$$

$$s = 1, \dots, p_r,$$

$$l = 1, \dots, v_{i,j},$$

$$\bar{y}_{r,s}^l - c_{i,j,r,s} \bar{x}_{r,s}^l - d_{i,j,r,s} \geq 0, \quad i = 1, \dots, n-1, \quad (3.6d)$$

$$r = i+1, \dots, n,$$

$$j = 1, \dots, p_i,$$

$$s = 1, \dots, p_r,$$

$$l = 1, \dots, v_{r,s}.$$

Lembrando que os vértices de um polígono transladado e rotacionado $P_{i,j}$ são dados por $(\bar{x}_{i,j}^l, \bar{y}_{i,j}^l)$, para $i = 1, \dots, n$, $j = 1, \dots, p_i$ e $l = 1, \dots, v_{i,j}$, em que

$$\bar{x}_{i,j}^l = x_{i,j}^l \cos \theta_i - y_{i,j}^l \sin \theta_i + \bar{x}_i^1,$$

e

$$\bar{y}_{i,j}^l = x_{i,j}^l \sin \theta_i + y_{i,j}^l \cos \theta_i + \bar{y}_i^1,$$

com $(\bar{x}_i^1, \bar{y}_i^1)$ o ponto de referência do polígono P_i e θ_i o ângulo de rotação.

As restrições (3.6a) e (3.6b), garantem que os polígonos estejam inteiramente dentro da faixa, a largura L é um parâmetro fixo e o comprimento W é uma variável; $\bar{x}_{i,j}^l$ e $\bar{y}_{i,j}^l$ dependem do ponto de referência e do ângulo de rotação do polígono, $(\bar{x}_i^1, \bar{y}_i^1)$ e θ_i , respectivamente, os quais são variáveis. As restrições (3.6c) e (3.6d), em conjunto, garantem a não-sobreposição dos polígonos convexos $P_{i,j}$ e $P_{r,s}$. Nessas restrições, $c_{i,j,r,s}$ e $d_{i,j,r,s}$ dependem do ponto de referência

e do ângulo de rotação da reta separadora, $(\bar{x}_{i,j,r,s}, \bar{y}_{i,j,r,s})$ e $\alpha_{i,j,r,s}$, respectivamente, os quais são variáveis.

O domínio das variáveis é definido da seguinte forma: $L \in \mathbb{R}^+$, $W \in \mathbb{R}^+$, $\bar{x}_i^1 \in \mathbb{R}^+$, $\bar{y}_i^1 \in \mathbb{R}^+$, $\theta_i \in [-2\pi, 2\pi]$, $\bar{x}_{i,j,r,s} \in \mathbb{R}^+$, $\bar{y}_{i,j,r,s} \in \mathbb{R}^+$ e $\alpha_{i,j,r,s} \in [-2\pi, 2\pi]$, para $i = 1, \dots, n$, $r = i + 1, \dots, n$, $j = 1, \dots, p_i$ e $s = 1, \dots, p_r$.

3.1.4.2 Modelo para o problema de empacotamento de itens irregulares em uma envoltória retangular

Nesta seção apresentamos um modelo para o problema de empacotamento de n itens irregulares em uma envoltória retangular a ser desenhada, com o objetivo de minimizar a área da envoltória retangular. A função objetivo é dada por:

$$z = W \times L. \quad (3.7)$$

Um modelo geral para o problema que chamamos de *Model for a Minimal Enclosure Problem* (MMEP), é dado por:

$$\begin{aligned} &\text{Minimizar } W \times L \\ &\text{sujeito a (3.6a) – (3.6d).} \end{aligned} \quad (3.8)$$

Note que, neste modelo, tanto a largura L , como o comprimento W , são variáveis.

Como dito anteriormente, a diferença entre dois modelos está na função objetivo.

3.2 Modelos para problemas de empacotamento de itens irregulares e círculos simultaneamente

Aos modelos apresentados nas Seções 3.1.4.1 e 3.1.4.2, podemos acrescentar o empacotamento de círculos. Usamos retas separadoras, apresentadas na Seção 3.1.3, para assegurar a não-sobreposição entre polígonos e círculos. Um círculo está em um lado da reta se o seu centro estiver naquele lado da reta e se a distância do centro à reta for maior ou igual ao seu raio. Diferentemente do que é apresentado em (KALLRATH, 2009), nos modelos que estamos propondo neste trabalho, uma reta que separa um círculo de um polígono não é necessariamente tangente ao círculo.

Ao acrescentar círculos aos problemas, é necessário acrescentar as variáveis referentes aos círculos nos modelos, quer dizer, as coordenadas do seu centro (x_i^c, y_i^c) .

A seguir, apresentamos as restrições que acrescentamos nos modelos ao acrescentar o empacotamento de círculos aos problemas. As restrições que garantem que todos os círculos

estejam inteiramente contidos na placa são apresentadas na Seção 3.2.1 e, restrições que garantem que os itens não se sobreponham são apresentadas na Seção 3.2.2.

3.2.1 Restrições de contenção

Temos de garantir que todos os círculos estejam inteiramente dentro da placa. Para isso, exigimos que:

$$r_i - x_i^c \leq 0, \quad (3.9)$$

$$r_i + x_i^c \leq W, \quad (3.10)$$

$$r_i - y_i^c \leq 0, \quad (3.11)$$

e,

$$r_i + y_i^c \leq L, \quad (3.12)$$

sendo (x_i^c, y_i^c) as coordenadas do centro do círculo C_i e r_i seu raio, para $i = 1, \dots, n_c$, com n_c o número de círculos a serem empacotados. Lembre que, no modelo MISPP apresentado na Seção 3.1.4.1, a largura L da faixa é fixa; e, no modelo MMEP apresentado na Seção 3.1.4.2, a largura L da envoltória retangular é variável.

3.2.2 Restrições de não-sobreposição

Nos modelos MISPP e MMEP, as retas separadoras são usadas para garantir a não-sobreposição.

Figura 21 – Reta separadora de um polígono convexo e um círculo.

Uma reta separa um círculo C_i de um polígono P_j se

- todos os vértices do polígono P_j estiverem em um lado (ou sobre) da reta separadora, e
- o centro do círculo C_i estiver no outro lado da reta separadora e distante de pelo menos o raio r_i de C_i .

Veja um exemplo de uma reta separadora de um polígono e um círculo na Figura 21.

As restrições que garantem a não-sobreposição entre um círculo C_i e um polígono P_j são dadas por

$$\bar{y}_j^l - c_{i,j}\bar{x}_j^l - d_{i,j} \geq 0, \quad l = 1, \dots, v_j, \quad (3.13)$$

$$y_i^c - c_{i,j}x_i^c - d_{i,j} \leq 0, \quad (3.14)$$

e

$$\Delta_{i,j} \geq r_i, \quad (3.15)$$

com $\Delta_{i,j} = \frac{|c_{i,j}x_i^c - y_i^c + d_{i,j}|}{\sqrt{c_{i,j}^2 + 1}}$ a distância do centro do círculo C_i à reta que separa C_i de P_j . Os valores de $c_{i,j}$ e $d_{i,j}$ são dados nas equações (3.3) e (3.4), respectivamente.

Figura 22 – Sobrepondo-se

Figura 23 – Tocando-se

Figura 24 – Separados

Nas restrições de não-sobreposição entre círculos, o uso de retas separadoras não é necessário, veja Figuras 22, 23 e 24. Garantimos que não há sobreposição entre dois círculos ao impor que a distância entre seus centros seja maior ou igual à soma de seus raios. Portanto, para dois círculos C_i e C_j , $i = 1, \dots, n_{c-1}$ e $j = i + 1, \dots, n_c$, a restrição é dada por

$$(x^{c_i} - x^{c_j})^2 + (y^{c_i} - y^{c_j})^2 \geq (r_i + r_j)^2. \quad (3.16)$$

Com essas restrições podemos completar os modelos MISPP e MMEP e concluir que o modelo para o problema de empacotamento de itens irregulares e círculos em faixa, ISPP, é dado pelo modelo MISPP, acrescentando as restrições (3.9) - (3.16), quer dizer, é dado por:

$$\begin{aligned} &\text{Minimizar } W \\ &\text{sujeito a (3.6a) – (3.6d),} \\ &\quad (3.9) – (3.16); \end{aligned} \tag{3.17}$$

e, o modelo para o problema de empacotamento de itens irregulares e círculos em uma envoltória retangular, MEP, é dado pelo modelo MMEP, acrescentando as restrições (3.9) - (3.16), quer dizer, é dado por:

$$\begin{aligned} &\text{Minimizar } W \times L \\ &\text{sujeito a (3.6a) – (3.6d),} \\ &\quad (3.9) – (3.16). \end{aligned} \tag{3.18}$$

Esses modelos mais gerais os chamamos pelos mesmos nomes, MISPP e MMEP, significa que, a partir de agora, MISPP é o modelo (3.17) e MMEP é o modelo (3.18). Lembrando que no MISPP a largura L é fixa, enquanto no modelo MMEP a largura L é variável.

Depois de ter apresentado dois modelos para problemas de empacotamento de itens irregulares e círculos, em que retas separadoras são usadas para garantir a não-sobreposição dos itens, no próximo capítulo apresentamos os métodos de resolução para resolver estes modelos e os resultados obtidos com estes.

MÉTODOS DE RESOLUÇÃO E RESULTADOS NUMÉRICOS

No Capítulo 3, foram apresentados dois modelos para a resolução do IPP. O primeiro modelo é para um ISPP, em que os itens são círculos e polígonos convexos ou não-convexos que podem rotacionar livremente. O segundo modelo é para um problema em que polígonos irregulares, convexos ou não-convexos, são empacotados juntamente a círculos em uma envoltória retangular a ser produzida. Neste modelo, a rotação livre dos polígono também é permitida. Retas separadoras são usadas em ambos modelos para garantir a não-sobreposição dos itens.

Ao considerar rotações livres dos polígonos e das retas separadoras nos modelos, estes se tornam modelos não-lineares. Os detalhes dos experimentos realizados com o primeiro e o segundo modelo são apresentados nas Seções 4.1 e 4.2, respectivamente.

Todos os experimentos numéricos foram realizados em um computador com processador Intel Core I7-4510U @ 2.1GHz e 8 GB de memória. Utilizamos um código para programação não-linear para resolver os problemas modelados, IPOPT (Wächter; Biegler, 2006), que é parte da COIN-OR (Wächter; Biegler, 2015).

IPOPT é a implementação de um método de barreira ou pontos interiores para problemas de otimização não-linear de grande porte do tipo

$$\begin{array}{ll} \text{Minimizar} & f(v) \\ \text{sujeita a} & g^L \leq g(v) \leq g^U, \\ & v^L \leq v \leq v^U, \end{array}$$

com $v \in \mathbb{R}^n$ e f, g continuamente diferenciável. Os detalhes deste método podem ser encontrados em (Wächter; Biegler, 2006; Nocedal; Wächter; Waltz, 2009). IPOPT pode ser executado em diferentes linguagens de programação, usamos a linguagem de programação C junto com a versão do Ipopt-3.12.3 e compilamos os códigos no sistema operacional

Ubuntu 12.04.

Apesar de nos modelos MISSP e MMEP o empacotamento de círculos e polígonos irregulares ser considerado, para o modelo MISSP a maioria dos testes computacionais foram executados usando instâncias clássicas da literatura em que os itens são unicamente polígonos, apenas uma única instância em que os itens são polígonos e círculos foi testada; para modelo MMEP, os testes computacionais foram executados usando instâncias apresentadas na literatura que empacotam círculos e polígonos simultaneamente. Os resultados obtidos para ambos os modelos foram comparados com outros apresentados na literatura que também permitem rotações livres.

4.1 Resultados numéricos para o problema de empacotamento de itens irregulares em faixa

Nesta seção, apresentamos os resultados obtidos na resolução do problema de empacotamento de itens irregulares em faixa. Para a resolução do problema usamos IPOPT, para o qual uma solução inicial é necessária. Apresentamos uma breve explicação da solução inicial utilizada na Seção 4.1.1. Na Seção 4.1.2, são apresentadas as tabelas resumindo os resultados obtidos com IPOPT para as instâncias em que os itens são unicamente polígonos, e as comparações destes resultados com duas metodologias da literatura, (STOYAN; PANKRATOV; ROMANOVA, 2016; LIAO *et al.*, 2016), que também permitem rotações livres.

Além dos parâmetros padrão do IPOPT, usamos a estratégia de atualização adaptativa para parâmetros de barreira. O tempo de execução foi muito grande ao usarmos a Hessiana do Lagrangiano, portanto, sempre usamos uma opção dada no IPOPT para aproximar a Hessiana com memória limitada, o que torna o tempo de execução mais curto, sem afetar a qualidade da solução. O tempo máximo de execução foi limitado a uma hora.

Usamos a Biblioteca Geométrica Computacional *Computational Geometry Algorithms Library* (CGAL) para particionar os polígonos não-convexos, em particular, usamos a implementação do algoritmo de programação dinâmica de Greene (FOGEL; TEILLAUD, 2014).

Para testar o modelo MISSP, usaremos as instâncias usadas nas duas abordagens com as quais comparamos os resultados, disponíveis em (ESICUP, 2015).

As características mais importantes dessas instâncias são apresentadas na Tabela 1. Os nomes das instâncias são apresentados na primeira coluna. Na segunda e terceira coluna, o número de polígonos convexos e não-convexos são apresentados, respectivamente. O número total de polígonos, após a decomposição de polígonos não-convexos em polígonos convexos, é apresentado na quarta coluna. Na quinta coluna, o número total de vértices é apresentado. O número de variáveis e o número de restrições são apresentados na sexta e sétima coluna, respectivamente. A largura da faixa é apresentada na oitava coluna.

Tabela 1 – Dados das instâncias

Instância	pol. convexos	pol. não-convexos	total pol. convexos	vértices	variáveis	restrições	largura da faixa
albano	10	14	52	220	3907	11700	4900
blaz	16	12	48	216	3385	10792	15
dagli	21	9	51	228	3790	11988	60
jakobs1	15	10	35	146	1831	5468	40
jakobs2	14	11	42	158	2590	7367	40
marques	10	14	50	214	3628	11002	104
poly1a	10	5	22	81	712	1968	40
poly2a	20	10	44	162	2875	7500	40
poly3a	30	15	66	243	6490	16596	40
poly4a	40	20	88	324	11557	29256	40
poly5a	50	25	110	405	18076	45480	40
poly10a	100	50	220	810	72451	180060	40
poly20a	200	100	440	1620	290101	716520	40
shirts	60	39	169	739	42583	126236	40
swim	6	42	291	1446	123787	415845	5752
trousers	48	16	104	468	16045	49476	79

4.1.1 Solução inicial

IPOPT foi desenhado para encontrar ótimos locais. Tendo em conta que o modelo não-linear proposto é não-convexo, ao resolvê-lo, podemos encontrar muitos pontos estacionários com diferentes valores na função objetivo. Esses pontos estacionários dependem da solução inicial.

Geramos uma solução inicial usando um algoritmo *Bottom Left* (BL). O algoritmo BL é uma heurística de passagem única que, tendo um conjunto de itens e uma ordem determinada, aloca os itens em uma faixa, o mais à esquerda e mais abaixo possível.

Passamos uma lista dos polígonos ordenados aleatoriamente ao algoritmo. Esta lista é representada por uma sequência, e para decodificar a sequência, usamos a técnica apresentada em (MUNDIM; ANDRETTA; QUEIROZ, 2017). Para evitar sobreposições, o algoritmo usa NFR, conceito também introduzido em (MUNDIM; ANDRETTA; QUEIROZ, 2017). Em NFR, para representar a faixa é utilizada uma grade discreta de pontos. Em (MUNDIM; ANDRETTA; QUEIROZ, 2017) é utilizada a seguinte regra para definir o cálculo da grade: para instâncias com uma largura menor do que 2500, é usada uma escala de 1; para instâncias com uma largura entre 2500 e 5000, é usada uma escala de 0.1; e, para instâncias com largura superior a 5000, é usado 0.01 como escala. Neste trabalho, como em (MUNDIM; ANDRETTA; QUEIROZ, 2017), para as instâncias com largura da faixa menor do que 2500, a escala usada é 1.0, exceto na única instância em que alguns dos seus vértices não são definidos em pontos inteiros, Dagli,

para esta instância usamos 0.5; nas instâncias Albano e Swim, (casos em que a largura da faixa é maior do que 2500), usamos 0.02 e 0.00005, respectivamente. Ao aumentar o tamanho da grade (ou diminuir a escala), o tempo de execução diminui, mas, poderíamos encontrar soluções infactíveis. O tamanho da grade usado nas instâncias Dagli, Albano e Swim, é maior ao sugerido em (MUNDIM; ANDRETTA; QUEIROZ, 2017), porém, sempre encontramos soluções iniciais factíveis em tempos computacionais razoáveis.

Embora no modelo permitamos a rotação livre, o algoritmo BL não permite a rotação livre dos polígonos, portanto, usamos quatro ângulos de rotação predefinidos (0° , 90° , 180° e 270°) em todos os casos. Para cada instância, executamos o algoritmo BL 1000 vezes, então escolhemos o leiaute com o menor comprimento obtido dessas 1000 execuções como uma solução inicial.

Uma solução inicial é o leiaute com o comprimento mais curto, entre 1000 execuções do algoritmo BL aplicado à lista aleatória de itens. Como a lista dos polígonos é aleatória, se executarmos o algoritmo mais 1000 vezes, o leiaute com o menor comprimento pode ser diferente do encontrado anteriormente. Portanto, para analisar a eficácia do nosso modelo, para cada instância, consideramos 10 soluções iniciais, obtidas da seguinte forma: executamos o algoritmo 10000 vezes e, a cada 1000 execuções escolhemos o leiaute com o menor comprimento. Executamos o IPOPT para resolver o modelo com cada uma dessas 10 soluções iniciais.

Quando no leiaute inicial os polígonos têm espaço suficiente para se mover, o modelo pode mudar o leiaute mais profundamente. Por outro lado, quando o leiaute inicial já está bastante apertado, o modelo executa uma fase de compactação, sem alterações profundas no leiaute. Para mostrar o quanto nosso modelo reduz o comprimento da solução inicial, apresentamos a Tabela 2, na qual as porcentagens de compactação, medidas pela redução do comprimento do leiaute da solução inicial para o leiaute de comprimento mínimo da solução final, são apresentadas (segunda coluna); as porcentagens mínimas, médias e máximas de compactação, em relação às 10 soluções iniciais diferentes utilizadas, também são apresentadas na terceira, quarta e quinta coluna, respectivamente. Observe que essas porcentagens são maiores em casos com itens de tamanhos semelhantes, como em todas as variações da instância poly, e que, para casos com um número menor de itens, a compactação é maior. Note também que a compactação é menor nas instâncias com itens grandes, tais como albano, marques, shirts ou trousers.

O comprimento da solução inicial não tem um impacto determinante no comprimento da solução final, o leiaute da solução inicial é o que determina o comprimento da solução final. Duas soluções iniciais com o mesmo comprimento podem produzir duas soluções finais com comprimentos diferentes, desde que os leiautes nas soluções iniciais sejam diferentes. Veja um exemplo na Figura 25. Porém, em nossos experimentos, observamos que em 10 das 16 instâncias testadas, a solução inicial mais curta produziu a solução final mais curta.

As retas separadoras são determinadas em uma fase de pré-processamento. Uma reta separadora não necessariamente passa por um lado de um dos dois polígonos que está separando;

Tabela 2 – Porcentagens de compactação do comprimento na solução inicial ao comprimento na solução final.

Instância	Compc.	Min. Compc.	Média Compc.	Max. Compc.
albano	2,60%	0,00%	2,20%	4,55%
blaz	4,06%	1,80%	6,27%	11,84%
dagli	8,17%	2,50%	5,07%	8,17%
jakobs1	13,33%	0,00%	4,00%	13,33%
jakobs2	7,14%	0,00%	5,26%	8,90%
marques	3,80%	0,00%	1,58%	3,80%
poly1a	26,23%	15,90%	20,71%	26,89%
poly2a	20,71%	11,25%	16,75%	20,71%
poly3a	18,71%	7,29%	15,08%	18,71%
poly4a	17,75%	13,34%	15,68%	17,99%
poly5a	16,86%	13,21%	15,46%	16,98%
poly10a	15,78%	7,27%	14,46%	15,97%
poly20a	14,18%	2,05%	10,38%	14,24%
shirts	2,81%	0,56%	2,10%	3,70%
swim	9,41%	4,30%	6,74%	10,09%
trousers	2,59%	1,24%	2,69%	4,15%

(a) Leiaute solução inicial 1

(b) Leiaute solução final 1

(c) Leiaute solução inicial 2

(d) Leiaute solução final 2

Figura 25 – Exemplo de duas soluções iniciais com o mesmo comprimento e soluções finais com comprimentos diferentes.

no entanto, na solução inicial, todas as retas separadoras têm uma rotação de 0° e passam por um lado de um dos dois polígonos que está separando. Visitamos todos os lados dos polígonos e verificamos se a reta que passa por cada lado é uma reta separadora. Note que, se os polígonos são convexos e não se sobrepõem, pelo menos um dos lados dos polígonos determina uma reta separadora. Escolhemos a primeira reta separadora encontrada. Foi notado nos experimentos que, quando mais de um lado dos polígonos determina uma reta separadora, os resultados podem ser diferentes dependendo da reta separadora usada.

4.1.2 Comparação dos resultados

Resolvemos o modelo que representa o problema de empacotamento de itens irregulares em faixa apresentado neste trabalho para as instâncias da Tabela 1. Usamos 10 soluções iniciais diferentes e comparamos nossos resultados com alguns resultados da literatura que também permitem rotações livres, (STOYAN; PANKRATOV; ROMANOVA, 2016) e (LIAO *et al.*, 2016). Esses resultados estão resumidos na Tabela 3, na qual são apresentadas o mínimo (segunda coluna), a média (terceira coluna) e o máximo (quarta coluna) dos comprimentos obtidos. O tempo médio usado para construir as soluções iniciais (quinta coluna) e o tempo médio (sexta coluna) que foi gasto para resolver as instâncias também são apresentados na Tabela 3. O comprimento da faixa e o tempo relatado em (STOYAN; PANKRATOV; ROMANOVA, 2016) estão nas sétima e oitava colunas, e o comprimento da faixa e o tempo relatado em (LIAO *et al.*, 2016) estão na nona e décima coluna, respectivamente.

Para todas as instâncias testadas em (STOYAN; PANKRATOV; ROMANOVA, 2016), pode-se observar que o comprimento da melhor solução que obtivemos é ligeiramente maior que o comprimento relatado por eles. No entanto, ao comparar com (LIAO *et al.*, 2016) podemos ver que, na maioria dos casos, o comprimento mínimo obtido usando nosso modelo é menor. Nas restantes instâncias, o comprimento é muito próximo. Por outro lado, note que nosso modelo encontra boas soluções para problemas com um grande número de polígonos, embora o número de variáveis, de restrições e o tempo computacional cresçam drasticamente. As soluções obtidas podem não ser ótimos locais para os casos em que o tempo máximo de execução foi atingido.

Os leiautes do comprimento mínimo obtido e a solução inicial usada para encontrá-los, para cada instância, podem ser vistos nas Figuras 26 - 41.

Figura 26 – Instância Albano

Figura 27 – Instância Blaz

Figura 28 – Instância Dagli

Figura 29 – Instância Jakobs1

Apenas testamos uma instância para o problema de empacotamento de itens irregulares em faixa em que círculos e polígonos são empacotados simultaneamente. Esta instância foi apresentada em (STOYAN; ZLOTNIK; CHUGAY, 2012), sendo que os itens devem ser posicionados em uma faixa que pode ter regiões proibidas (apenas um exemplo sem regiões proibidas foi apresentado nesse artigo). A instância consiste em 44 polígonos, sendo alguns não-convexos, e 5 círculos de tamanhos diferentes. O valor da largura é $L=25$. Para modelar esta instância, foram usadas 19335 variáveis e 54860 restrições.

Da mesma forma como nas instâncias em que os itens são unicamente polígonos, usamos 10 soluções iniciais diferentes, cada solução inicial foi gerada usando um algoritmo BL, como apresentado na Seção 4.1.1, em que os círculos foram representados pela sua envoltória retangular.

Em (STOYAN; ZLOTNIK; CHUGAY, 2012), um comprimento de 33,90 em 7200.00 segundos de execução foi alcançado, o comprimento mínimo que nós conseguimos foi de 35,44 em 130,34 segundos. A média e o máximo dos comprimentos obtidos foram 36,19 e 36,96, respectivamente. O tempo médio para construir a solução inicial foi de 197,99 segundos e o tempo médio que foi gasto para resolver a instância foi de 883,01 segundos. O leiaute do comprimento mínimo obtido e da solução inicial usada para encontrá-lo são exibidos na Figura 42.

(a) Leiaute da solução inicial

(b) Leiaute da solução final

Figura 30 – Instância Jakobs2

(c) Leiaute da solução inicial

(d) Leiaute da solução final

Figura 31 – Instância Marques

(a) Leiaute da solução inicial

(b) Leiaute da solução final

Figura 32 – Instância Poly1a

(c) Leiaute da solução inicial

(d) Leiaute da solução final

Figura 33 – Instância Poly2a

Figura 34 – Instância Poly3a

Figura 35 – Instância Poly4a

Figura 36 – Instância Poly5a

4.2 Resultados numéricos para o problema de empacotamento de itens irregulares em uma envoltória retangular

Nesta seção, apresentamos os resultados obtidos na resolução do problema de empacotamento de itens irregulares em uma envoltória retangular. Nossos resultados são comparados com os resultados apresentados em (KALLRATH, 2009), nos quais são empacotados círculos e polígonos simultaneamente.

Em (KALLRATH, 2009), como no nosso modelo, os itens devem ser posicionados em uma envoltória retangular. A envoltória retangular está sujeita a limitantes inferiores e superiores da largura e do comprimento.

Testamos todas as instâncias com círculos e polígonos convexos que foram apresentadas

(a) Leiaute da solução inicial

(b) Leiaute da solução final

Figura 37 – Instância Poly10a

(a) Leiaute da solução inicial

(b) Leiaute da solução final

Figura 38 – Instância Poly20a

(a) Leiaute da solução inicial

(b) Leiaute da solução final

Figura 39 – Instância Shirts

em (KALLRATH, 2009) e que possuem dados disponíveis. Usamos os mesmos limitantes, estes são $\bar{W} = 4$ para o limitante superior da largura e $\bar{L} = 8$ para o limitante superior do comprimento

Figura 40 – Instância Swim

Figura 41 – Instância Trousers

para todas as instâncias, exceto para *clp1-3*, para o qual temos $\bar{W} = 2.5$ e $\bar{L} = 3$, e para *c6p3*, para o qual temos $\bar{W} = 6$ e $\bar{L} = 10$.

Os resultados obtidos ao resolver essas instâncias usando nosso modelo e aqueles apresentados em (KALLRATH, 2009) estão resumidos na Tabela 4. Os nomes das instâncias são apresentados na primeira coluna. O número de círculos e polígonos são apresentados na segunda e terceira coluna, respectivamente. Da quarta à sétima coluna, os dados e resultados encontrados em (KALLRATH, 2009) são apresentados. Da coluna oitava à décima primeira, nossos dados e resultados são apresentados. Nestes, n é o número de variáveis do modelo, m é o número de restrições do modelo; CPU é o tempo de execução em segundos; e a é a área da envoltória retangular. Em todos os casos, o IPOPT parou em soluções ótimas locais. A solução obtida pelo modelo proposto e a solução inicial usada para encontrá-la, para cada instância, são exibidos das Figuras 43 até a 50.

Podemos observar que o número de variáveis e restrições no nosso modelo é significativa-

(a) Leiaute solução inicial

(b) Leiaute solução final

Figura 42 – Instância apresentada em (STOYAN; ZLOTNIK; CHUGAY, 2012).

(a) Leiaute solução inicial

(b) Leiaute solução final

Figura 43 – Instância: c1p1-1.

Figura 44 – Instância: c1p1-2.

Figura 45 – Instância: c1p1-3.

Figura 46 – Instância: c1p5a.

Figura 47 – Instância: c1p5b.

mente menor do que o número de variáveis e restrições do modelo apresentado em (KALLRATH, 2009). O tempo de execução é consideravelmente reduzido, o que pode ser atribuído, não só à

Figura 48 – Instância: c1p6a.

Figura 49 – Instância: c3p3.

Figura 50 – Instância: c6p3.

redução de variáveis e restrições no modelo, mas também às diferenças no sistema operacional, arquitetura de hardware, entre outros. Também podemos observar nos nossos resultados que a área é menor ou igual àqueles obtidos em (KALLRATH, 2009) na maioria dos casos.

O alto número de variáveis no modelo utilizado em (KALLRATH, 2009) se deve ao fato que todos os vértices do polígono são tratados como variáveis. Além disso, à equação vetorial da reta é usada para modelar as retas separadoras, o que, por sua vez, implica o emprego de muitas variáveis, como o vetor diretor, vetor normal, vetores que conectam a reta separadora com os vértices e a distância dos vértices para a reta separadora, entre outras. Nos nossos modelos, apenas o ponto de referência e o ângulo de rotação para cada polígono e para cada reta separadora

são variáveis.

No próximo capítulo, apresentamos uma nova ferramenta para garantir que dois itens estejam separados, as parábolas separadoras.

Tabela 3 – Comparação dos nossos resultados com aqueles em (STOYAN; PANKRATOV; ROMANOVA, 2016) e (LIAO *et al.*, 2016).

Instância	Nossa abordagem					Melhores resultados em (STOYAN; PANKRATOV; ROMANOVA, 2016)		Melhores resultados em (LIAO <i>et al.</i> , 2016)	
	Solução mínima	Solução média	Solução máxima	SI tempo médio (s)	IPOPT tempo médio (s)	Comprimento	Tempo(s)	Comprimento	Tempo(s)
albano	10355,80	10601,03	10849,99	158,68	178,56	10032,24	124,39		
blaz	27,82	29,02	30,44	1,37	49,52	25,41	25,42	28,27	56,86
dagli	60,60	61,96	63,36	404,13	135,10	56,90	139,00	59,24	132,58
jakobs1	12,99	13,49	14,00	4,68	19,30			13,19	48,46
jakobs2	26,00	27,29	30,00	17,60	18,41			24,25	53,67
marques	84,65	86,68	91,00	66,74	58,11			84,93	118,12
poly1a	14,01	14,53	15,01	6,23	19,07			13,90	27,59
poly2a	26,16	27,30	27,92	10,68	87,53			26,67	61,23
poly3a	39,01	40,29	44,50	15,44	773,32			39,48	149,66
poly4a	50,99	52,50	53,72	20,59	1621,54			51,13	210,74
poly5a	63,66	65,05	66,31	26,21	1773,85			65,64	287,32
poly10a	127,16	129,46	140,01	58,25	2350,17	126,29	618,80		
poly20a	254,86	268,06	294,81	151,38	3484,92	251,04	1209,17		
shirts	62,19	64,43	65,62	11,55	1808,73	5661,95	431,97	65,06	340,89
swim	6011,93	6311,16	6526,38	449,85	3600,00				
trousers	249,35	258,03	261,94	118,00	603,36			251,94	265,48

Tabela 4 – Comparação dos nossos resultados com aqueles em (KALLRATH, 2009).

			Nossa abordagem				Resultados em (KALLRATH, 2009)			
Instância	C	P	n	m	CPU	a	n	m	CPU	a
c1p1-1	1	1	10	26	0,01	1,00	47	54	0,10	1,00
c1p1-2	1	1	10	26	0,02	1,50	47	54	42,00	1,50
c1p1-3	1	1	10	26	0,01	1,50	47	54	368,00	2,50
c1p5a	1	2	19	80	0,89	11,70	767	799	2000,00	15,90
c1p5b	1	5	64	284	0,06	19,73	791	838	1800,00	18,55
c1p6a	1	6	85	376	0,05	20,06	1110	1161	1800,00	20,37
c3p3	3	3	53	177	0,03	16,86	489	516	2000,00	16,75
c6p3	6	3	86	219	0,04	33,19	628	661	33000,00	35,10

MODELO DE UM PROBLEMA DE EMPACOTAMENTO DE ITENS IRREGULARES USANDO PARÁBOLAS SEPARADORAS

Até o momento, usamos retas separadoras para garantir a não-sobreposição dos itens. Com o objetivo de ir além das retas separadoras, visando ter melhores resultados no caso em que não só polígonos convexos devem ser empacotados, neste capítulo estudamos uma outra ferramenta para garantir a não-sobreposição de itens, isto é, estudamos curvas separadoras, mais especificamente, parábolas separadoras. Uma parábola separa dois itens se um item está “dentro” da parábola e o outro item está “fora” da parábola, como pode ser visto na Figura 51. Dizemos que um item está “dentro” da parábola se todos os seus pontos interiores estão no lado da parábola em que está o foco. Assim, um item está “fora” da parábola se todos os seus pontos interiores estão no outro lado da parábola.

Figura 51 – Parábola separadora de dois polígonos.

No caso de empacotar unicamente polígonos convexos, as retas separadoras têm melhor desempenho ao ser comparadas com as parábolas separadoras, já que a forma curva das

parábolas não deixará encostar totalmente um polígono em outro. Porém, quando queremos separar dois itens, sendo um deles não-convexo (como por exemplo um polígono não-convexo sem decomposição em convexos ou um item não-convexo delimitado por arcos circulares), as parábolas nos permitiriam encontrar soluções que as retas não permitem. Veja um exemplo nas Figuras 52 e 53.

Figura 52 – Separação de dois itens, sendo um deles não-convexo.

Figura 53 – Separação de dois itens, sendo um deles não-convexo.

Pode-se observar que precisaríamos aumentar o grau da curva separadora na medida que aumentam as não-convexidades para conseguir um arranjo ótimo dos itens; por exemplo, para dois polígonos não-convexos, um com uma não-convexidade e outro com duas não-convexidades, uma parábola não seria suficiente para conseguir um arranjo ótimo, neste caso precisaríamos de uma curva de quarto grau. Veja um exemplo na Figura ??.

Ao aumentar o grau da curva, determinar se a curva separa dois itens se torna mais complexo, e, por isso, nos limitamos neste trabalho ao estudo de parábolas separadoras, deixando o estudo de curvas separadoras de maior grau como trabalho futuro. Deixaremos também como trabalho futuro o estudo de parábolas separadoras para itens delimitados por arcos circulares; trataremos unicamente com parábolas separando polígonos.

Figura 54 – Curva separadora de quarto grau para dois polígonos não-convexos.

5.1 Restrições para garantir a separação de dois polígonos usando parábolas separadoras

Daqui em diante, assumimos que o polígono P_i é o polígono que está do lado de “fora” da parábola e o polígono P_j é aquele que está do lado de “dentro”. A determinação do qual polígono estará por “dentro” e qual por “fora” da parábola deverá ser feita em uma fase de pré-processamento. Muitos detalhes devem ser levados em consideração nesta fase, por exemplo, se desejamos empacotar dois polígonos, sendo um deles não-convexo e o outro convexo, é preferível que o polígono convexo esteja do lado de “dentro” da parábola, ou, por exemplo, que o polígono com menos vértices esteja do lado de fora. O desenvolvimento desta fase o deixamos para trabalho futuro. Algumas soluções podem ser perdidas ao fixar que um polígono esteja do lado de “dentro” e o outro de lado de “fora”, mas, permitir que durante a resolução o polígono mude de lado implicaria no uso de muitas outras variáveis.

Como nas retas separadoras, uma parábola separa dois polígonos, P_i e P_j , se todos os vértices do polígono P_i estão ou sobre a parábola ou a um lado da parábola, e todos os vértices do polígono P_j estão ou sobre a parábola ou do outro lado da parábola. Porém, que todos os vértices do polígono P_i estejam por fora da parábola e todos os vértices do polígono P_j estejam por dentro não é suficiente para garantir que a parábola seja separadora, como pode ser visto na Figura 55, na qual isto é obedecido e ainda assim a parábola não separa os dois polígonos e temos sobreposição.

Para garantir que uma parábola seja separadora, além dos vértices do polígono P_i estarem por fora e os vértices do polígono P_j estarem por dentro da parábola, não devemos ter nenhum ponto da parábola no interior do polígono P_i . Para isto, vamos verificar a interseção da parábola com cada uma das retas que contém as arestas do polígono P_i .

Figura 55 – Sobreposição dos polígonos.

5.1.1 Posição relativa entre a parábola e as arestas de um polígono

As possíveis posições relativas de uma parábola e a reta que contém uma aresta são (veja a Figura 56):

1. A reta é secante. Neste caso, sua interseção é de dois pontos.
2. A reta é tangente à parábola. Neste caso, sua interseção é um ponto.
3. A reta apenas corta em um único ponto se a reta for paralela ao eixo de simetria da parábola.
4. A reta não corta a parábola, isto é, está fora da parábola. Neste caso, não há um ponto comum para ambas.

Figura 56 – Posição relativa da parábola e as arestas

Uma parábola é uma curva cônica representada pela equação geral de segundo grau

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0, \quad (5.1)$$

com A, B, C, D, E e F os parâmetros da parábola. Estes parâmetros se podem determinar a partir de dois pontos, o vértice e o foco, (v_x, v_y) e (f_x, f_y) , respectivamente. Os detalhes do procedimento para calcular estes parâmetros são apresentados no Apêndice A.

A equação da reta que contém a aresta r que vai do vértice $(x_{i,r}, y_{i,r})$ ao vértice $(x_{i,r+1}, y_{i,r+1})$ está dada por:

$$(y_{i,r} - y_{i,r+1})x + (x_{i,r+1} - x_{i,r})y + x_{i,r}(y_{i,r+1} - y_{i,r}) - y_{i,r}(x_{i,r+1} - x_{i,r}) = 0 \quad (5.2)$$

Para determinar as coordenadas dos pontos de interseção da parábola e da reta, resolvemos o sistema não-linear formado pelas equações (5.1) e (5.2), o que nos levará a uma equação de segundo grau, $ax^2 + bx + c = 0$. Ao resolver esta equação, resultados diferentes são obtidos dependendo do valor do discriminante, $\Delta = b^2 - 4ac$.

- Se o discriminante é negativo, $\Delta < 0$, a equação não possui soluções reais. Quer dizer, o sistema não possui solução, portanto a reta não corta a parábola e é dito estar fora dela.
- Se o discriminante for nulo, $\Delta = 0$, a equação tem duas soluções reais iguais, ou seja, a reta corta a parábola em um único ponto. Neste caso, ou a reta é tangente à parábola ou a reta é paralela ao eixo de simetria da parábola.
- Se o discriminante for positivo, $\Delta > 0$, a equação tem duas soluções reais e diferentes, ou seja, a reta possui dois pontos comuns com a parábola. Então é dito que a reta é secante à parábola.

5.1.2 Restrições de não-sobreposição

Como as arestas são do polígono P_i e os vértices deste polígono estão por fora da parábola, podemos garantir que os polígonos estão separados se para cada uma das retas que contém uma aresta:

- ou está fora, ou é tangente, ou é paralela ao eixo de simetria da parábola;
- ou é secante à parábola e $(x_{i,r}, y_{i,r})$ e $(x_{i,r+1}, y_{i,r+1})$ estão a um mesmo lado dos pontos de interseção, (p_x^1, p_y^1) e (p_x^2, p_y^2) .

Se $(x_{i,r}, y_{i,r})$ e $(x_{i,r+1}, y_{i,r+1})$ estão de lados diferentes dos pontos de interseção, (p_x^1, p_y^1) e (p_x^2, p_y^2) , então temos sobreposição, veja as Figuras 57 e 58.

Figura 57 – Vértices em lados diferentes dos pontos de interseção

Figura 58 – Vértices no mesmo lado dos pontos de interseção

Assim, uma parábola separa dois polígonos P_i e P_j (considerando que, se $Ax^2 + Bxy + Cy^2 + Dx + Ey + F < 0$ então (x, y) está do lado de fora da parábola) se

$$\begin{cases} Ax_{i,r}^2 + Bx_{i,r}y_{i,r} + Cy_{i,r}^2 + Dx_{i,r} + Ey_{i,r} + F \leq 0, & \forall r \in \{1, \dots, m_i\}, \\ Ax_{j,s}^2 + Bx_{j,s}y_{j,s} + Cy_{j,s}^2 + Dx_{j,s} + Ey_{j,s} + F \geq 0, & \forall s \in \{1, \dots, m_j\}, \end{cases} \quad (5.3)$$

$$\delta_r \Delta_r \geq 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4a)$$

$$\delta_r [x_{i,r} - p_x^1 - \lambda_r^1 (p_x^2 - p_x^1)] = 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4b)$$

$$\delta_r [y_{i,r} - p_y^1 - \lambda_r^1 (p_y^2 - p_y^1)] = 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4c)$$

$$\delta_r [x_{i,r+1} - p_x^1 - \lambda_r^2 (p_x^2 - p_x^1)] = 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4d)$$

$$\delta_r [y_{i,r+1} - p_y^1 - \lambda_r^2 (p_y^2 - p_y^1)] = 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4e)$$

$$\lambda_r^1 \lambda_r^2 > 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4f)$$

$$(1 - \delta_r) \Delta_r \leq 0, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4g)$$

$$\delta_r \in \{0, 1\}, \quad \forall r \in \{1, \dots, m_i\}, \quad (5.4h)$$

$$\lambda_r^1, \lambda_r^2 \in \mathbb{R}, \quad \forall r \in \{1, \dots, m_i\} \quad (5.4i)$$

e

$$(v_x - f_x)^2 + (v_y - f_y)^2 > 0, \quad (5.5)$$

sendo m_i e m_j o numero de vértices do polígono P_i e P_j , respectivamente. Nas restrições (5.4d) e (5.4e), se $r = m_i$, então $r + 1 = 1$.

O conjunto de restrições (5.3) garante que os vértices do polígono P_i estejam por fora da parábola e que os vértices do polígono P_j estejam por dentro da parábola.

O conjunto de restrições (5.4) garante que não há pontos da parábola no interior do polígono P_i .

No conjunto de restrições (5.4), podemos ter as seguintes possibilidades:

- Se $\delta_r = 1$, então $\Delta_r > 0$. Quer dizer que a reta que contém a aresta r é secante à parábola, restrição (5.4a), pelo que devemos garantir que os vértices $(x_{i,r}, y_{i,r})$ e $(x_{i,r+1}, y_{i,r+1})$ estão a um mesmo lado dos pontos de interseção (p_x^1, p_y^1) e (p_x^2, p_y^2) , restrições (5.4b) - (5.4f).
- $\delta_r = 1$ e $\Delta_r = 0$ é inviável, devido a que neste caso teríamos $(p_x^1, p_y^1) = (p_x^2, p_y^2) = (x_{i,r}, y_{i,r}) = (x_{i,r+1}, y_{i,r+1})$, o que não é possível porque todos os vértices são diferentes.
- $\delta_r = 1$ e $\Delta_r < 0$ é inviável, devido à restrição (5.4a).
- Se $\delta_r = 0$, então $\Delta_r \leq 0$, quer dizer que a reta que contém a aresta r ou está fora, ou é tangente, ou é paralela ao eixo de simetria da parábola, restrição (5.4g).
- $\delta_r = 0$ e $\Delta_r > 0$ é inviável devido à restrição (5.4g).

A restrição (5.4f) nos impede ter uma solução na qual um vértice seja igual a um ponto de interseção (se λ_r^1 ou λ_r^2 fossem nulos não poderíamos garantir que os vértices estejam a um mesmo lado dos pontos de interseção), porém poderíamos ter uma solução na qual um vértice e um ponto de interseção estejam o suficientemente próximos.

A restrição (5.5) garante que o vértice e o foco da parábola sejam pontos diferentes, já que se o vértice e o foco é o mesmo ponto não teríamos uma parábola, teríamos simplesmente um ponto.

As restrições (5.3), (5.4) e (5.5) fazem parte do que seria o primeiro modelo não-linear inteiro misto para o problema de empacotamento de polígonos irregulares. Deve-se notar que o modelo lida com polígonos irregulares sem fazer diferença se são convexos ou não-convexos, já que não é preciso decompor os não-convexos em convexos.

Se desejamos empacotar n polígonos, para cada par de polígonos P_i e P_j , com $i, j \in \{1, 2, \dots, n\}$, usamos uma parábola separadora. As parábolas separadoras iniciais são determinadas em uma fase de pré-processamento, na qual devemos estabelecer, para cada par de polígonos, qual será o polígono que vai estar por fora da parábola e qual por dentro. As variáveis necessárias para cada parábola que separa o polígono P_i do polígono P_j são:

- 4 variáveis contínuas: as coordenadas do vértice e do foco;

- 2 variáveis contínuas para cada aresta do polígono P_i ;
- uma variável binária para cada aresta do polígono P_i .

Com a variação do vértice e do foco conseguimos variação da curvatura da parábola, assim como translação e rotação da parábola. Os discriminantes e os pontos de interseção são calculados a partir das outras variáveis.

Poucos trabalhos tem sido desenvolvidos em que os itens sejam representados exatamente quando têm bordas curvas. Com o estudo de parábolas separadoras, e no futuro de curvas separadoras, esperamos poder construir modelos para problemas de empacotamentos de itens irregulares representados exatamente e que rotacionem livremente.

Até o momento, desenvolvemos restrições para o empacotamento de polígonos, com as quais podemos empacotar polígonos não-convexos sem a necessidade de decomposição em polígonos convexos. O seguinte passo será construir um modelo para um problema em específico de empacotamento de itens irregulares e realizar experimentos numéricos para verificar a efetividade do modelo. A inclusão do empacotamento de círculos no problema deverá ser feita para começar a conduzir nosso trabalho ao empacotamento de itens irregulares com bordas curvas representados exatamente.

As parábolas ou curvas separadoras não deixam encostar totalmente um polígono em outro, pelo que se espera que no caso de empacotamento de unicamente polígonos, as retas separadoras tenham melhor desempenho. As retas, parábolas e curvas separadoras terão de ser combinadas em um modelo só, para conseguir abranger o empacotamento de todo tipo de itens sem perder soluções ótimas.

CONCLUSÕES E TRABALHOS FUTUROS

Neste trabalho, estudamos o problema de empacotamento de itens irregulares. Dois modelos são propostos, no primeiro modelo círculos e polígonos convexos ou não-convexos, que podem rotacionar livremente, são empacotados em uma faixa, com o objetivo de minimizar o comprimento da faixa. No segundo modelo círculos e polígonos convexos ou não-convexos, que também podem rotacionar livremente, são empacotados simultaneamente em uma envoltória retangular a ser desenhada, com o objetivo de minimizar a área da envoltória.

Nos modelos que representam os problemas de empacotamento, as principais restrições são as de contenção, quer dizer, as restrições que garantem que os itens estejam inteiramente contidos na placa, e as restrições de não-sobreposição, que é uma das maiores dificuldades devido aos itens serem irregulares. Retas separadoras são usadas nos dois modelos para garantir a não-sobreposição dos itens.

Como um ponto relevante deste trabalho, destacamos o uso da equação geral da reta para modelar as retas separadoras, permitindo-nos usar apenas três variáveis para cada reta separadora: as coordenadas do ponto de referência e o ângulo de rotação. Isso estabelece uma marcada diferença com outros modelos apresentados na literatura, por exemplo, o apresentado em (KALLRATH, 2009), que possui um maior número de variáveis. O alto número de variáveis no modelo apresentado em (KALLRATH, 2009) é devido a todos os vértices dos polígonos serem tratados como variáveis, além do fato que, para modelar as retas separadoras, é usada a equação vetorial da reta, o que implica o uso de muitas variáveis. De fato, para uma instância com dois polígonos convexos de 5 e 6 vértices usaria:

- Para as retas separadoras: onze variáveis para as distâncias dos vértices até a reta separadora, duas variáveis para o vetor base, vinte e duas variáveis para os vetores que conectam a reta separadora com os vértices, onze variáveis auxiliares necessárias para calcular esses vetores, duas variáveis para o vetor diretor e, duas variáveis para o vetor normal.

- Para os polígonos: duas variáveis para o ângulo de orientação dos polígonos, duas variáveis para o raio do menor círculo que contém cada polígono, vinte e duas variáveis para as coordenadas dos vértices e quatro variáveis para as coordenadas centrais dos polígonos.
- Outros: uma variável para a área da envoltória retangular, duas variáveis para as extensões da envoltória retangular, uma variável para a função objetivo.

Em vez disso, para uma instância com dois polígonos convexos e usando a equação geral da reta, unicamente precisamos de 9 variáveis para modelar a separação dos dois polígonos, uma simplificação significativa que levaria a um melhor desempenho do método de solução sem deixar de garantir uma boa solução para o problema.

Para testar nossos modelos, usamos um código para programação não-linear, IPOPT (Wächter; Biegler, 2006), que depende substancialmente de uma solução inicial. Para o primeiro modelo, utilizamos um algoritmo de BL para construir as soluções iniciais. Para o segundo modelo, utilizamos uma solução inicial aleatória.

Para testar a eficácia do primeiro modelo, comparamos nossos resultados com os apresentados em (Stoyan; Pankratov; Romanova, 2016; Liao *et al.*, 2016; Stoyan; Zlotnik; Chugay, 2012), que foram obtidos utilizando metodologias que permitem rotações livres. Nas instâncias utilizadas para comparar nossos resultados com os apresentados em (Stoyan; Pankratov; Romanova, 2016; Liao *et al.*, 2016) os itens são unicamente polígonos. Apenas uma instância em que círculos e polígonos são empacotados simultaneamente foi testada, apresentada em (Stoyan; Zlotnik; Chugay, 2012). O comprimento relatado em (Stoyan; Zlotnik; Chugay, 2012) é menor, mas muito próximo do que encontramos. Os comprimentos relatados em (Stoyan; Pankratov; Romanova, 2016) são menores, mas muito próximos dos que encontramos. Por outro lado, os comprimentos relatados em (Liao *et al.*, 2016) são maiores na maioria dos casos. Nos outros, eles estão muito próximos. Portanto, a eficácia deste modelo é verificada, bem como a qualidade das soluções iniciais construídas usando um algoritmo de BL. No entanto, acreditamos que esses resultados poderiam ser melhorados usando outro algoritmo para construir as soluções iniciais.

Para testar a eficácia do segundo modelo, comparamos nossos resultados com os apresentados em (Kallrath, 2009), nos quais as retas separadoras também são usadas, mas estas são modeladas usando a equação vetorial da reta. As áreas relatadas em (Kallrath, 2009) são iguais ou muito próximas às encontradas por nós, algumas um pouco menores outras um pouco maiores, mas o tempo de execução é sempre reduzido; com as reduções do número de variáveis, restrições e tempo computacional, conseguimos estender o modelo a polígonos não-convexos e resolver problemas maiores. Testamos sete instâncias das apresentadas em (Kallrath, 2009), todas aquelas para polígonos e círculos que tinham os dados disponíveis.

Neste trabalho também foi apresentado um conjunto de restrições que garante que dois polígonos irregulares convexos ou não-convexos estejam separados por uma parábola e que

permite a rotação livre da parábola e dos polígonos.

Nas restrições que modelam a separação de dois itens usando uma parábola não precisamos particionar um polígono não-convexo em polígonos convexos como é preciso ao modelar a separação usando uma reta.

A forma curva das parábolas não deixará encostar totalmente um polígono em outro, pelo que se espera que, ao empacotar unicamente polígonos, as retas separadoras tenham melhor desempenho que as parábolas separadoras.

Com o uso de retas separadoras podemos não conseguir um acoplamento ótimo de itens não-convexos se a não-convexidade é representada por um arco circular. O intuito das parábolas separadoras, além de poder empacotar polígonos não-convexos sem necessidade de particioná-los, é poder empacotar este tipo de itens delimitados por arcos circulares.

Trabalharemos em um método que construa soluções iniciais para os modelos formulados neste trabalho, com as quais acreditamos que os resultados poderiam ser melhorados.

Desenvolveremos um modelo para o problema de empacotamento de itens irregulares, representados por polígonos, que rotacionem livremente. Testes numéricos deverão ser feitos para verificar a efetividade das parábolas separadoras.

Trabalharemos em um modelo que use parábolas separadoras que inclua o empacotamento de círculos, para conduzir nosso trabalho ao desenvolvimento de um modelo para o problema do empacotamento de itens irregulares delimitados por arcos circulares.

Pretendemos realizar o estudo de curvas separadoras de grau maior do que dois para, finalmente, desenvolver um modelo que combine retas e curvas separadoras.

REFERÊNCIAS

ABEYSOORIYA, R. P.; BENNELL, J. A.; MARTINEZ-SYKORA, A. Jostle heuristics for the 2d-irregular shapes bin packing problems with free rotation. **International Journal of Production Economics**, v. 195, p. 12 – 26, 2018. Citado nas páginas 13, 24, 37, 38 e 39.

ALBANO, A.; SAPUPPO, A. Optimal allocation of two-dimensional irregular shapes using heuristic search methods. **IEEE Transactions on Systems, Man and Cybernetics**, v. 10, p. 242–248, 1980. Citado nas páginas 24 e 37.

ALVAREZ-VALDES, R.; MARTINEZ, A.; TAMARIT, J. M. A branch and bound algorithm for cutting and packing irregularly shaped pieces. **International Journal of Production Economics**, v. 145, p. 463–477, 2013. Citado nas páginas 24 e 43.

ART, R. C. **An approach to the two-dimensional, irregular cutting stock problem**. 1996. Citado na página 37.

BABU, A. R.; BABU, N. R. A generic approach for nesting of 2-D parts in 2-D sheets using genetic and heuristic algorithms. **Computers-Aided Design**, v. 33, p. 879–891, 2001. Citado nas páginas 28 e 40.

BENNELL, J.; SCHEITHAUER, G.; STOYAN, Y.; ROMANOVA, T. Tools of mathematical modelling of arbitrary object packing problems. **Ann Oper Res**, v. 179, p. 343–368, 2010. Citado nas páginas 13 e 35.

BENNELL, J. A.; DOWSLAND, K. A. A tabu thresholding implementation for the irregular stock cutting problem. **International Journal of Production Research**, v. 37, p. 4259–4275, 1999. Citado na página 40.

_____. Hybridising tabu search with optimisation techniques for irregular stock cutting. **Management Science**, v. 47, p. 1160–1172, 2001. Citado na página 42.

BENNELL, J. A.; OLIVEIRA, J. F. The geometry of nesting problems: A tutorial. **European Journal of Operational Research**, v. 184, p. 397–415, 2008. Citado nas páginas 13, 24, 28, 29, 31, 32, 33 e 34.

_____. A tutorial in irregular shape packing problems. **Journal of Operational Research Society**, v. 60, p. S93–S105, 2009. Citado nas páginas 13, 28, 38, 40 e 41.

BENNELL, J. A.; SONG, X. A beam search implementation for the irregular shape packing problem. **Journal of Heuristics**, v. 16, p. 167–188, Apr 2010. ISSN 1572-9397. Disponível em: <<https://doi.org/10.1007/s10732-008-9095-x>>. Citado na página 39.

BIRGIN, E. G.; MARTÍNEZ, J. M.; RONCONI, D. P. Optimizing the packing of cylinders into a rectangular container: A nonlinear approach. **European Journal of Operational Research**, v. 160, p. 19–33, 2005. Citado na página 24.

- 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. Citado na página 44.
- CASTILLO, I.; KAMPAS, F. J.; PINTÉR, J. D. Solving circle packing problems by global optimization: Numerical results and industrial applications. **European Journal of Operational Research**, v. 191, p. 786–802, 2008. Citado na página 46.
- CHERNOV, N.; STOYAN, Y.; ROMANOVA, T. Mathematical model and efficient algorithms for object packing problem. **Computational Geometry: Theory and Applications**, v. 43, p. 535–553, 2010. Citado nas páginas 24 e 44.
- CHERRI, L.; CARRAVILLA, M.; TOLEDO, F. A model-based heuristic for the irregular strip packing problem. **Pesquisa Operacional**, v. 36, p. 447–468, 2016. Citado nas páginas 24 e 45.
- CHERRI, L. H.; MUNDIM, L. R.; ANDRETTA, M.; TOLEDO, F. M.; OLIVEIRA, J. F.; CARRAVILLA, M. A. Robust mixed-integer linear programming models for the irregular strip packing problem. **European Journal of Operational Research**, v. 253, p. 570 – 583, 2016. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221716301370>>. Citado na página 44.
- 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. Citado nas páginas 13, 39 e 40.
- 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. 1294–1266, 2007. Citado nas páginas 24 e 37.
- ESICUP. **EURO special interest group on cutting and packing**. 2015. [Http://paginas.fe.up.pt/esicup/datasets](http://paginas.fe.up.pt/esicup/datasets). Citado na página 62.
- FISCHETTI, M.; LUZZI, I. Mixed-integer programming models for nesting problems. **Journal of Heuristics**, v. 15, p. 201–226, 2009. Citado nas páginas 24 e 43.
- FOGEL, E.; TEILLAUD, M. The computational geometry algorithms library cgal. **ACM Commun. Comput. Algebra**, ACM, New York, NY, USA, v. 47, p. 85–87, jan. 2014. ISSN 1932-2240. Disponível em: <<http://doi.acm.org/10.1145/2576802.2576806>>. Citado na página 62.
- FOWLER, R. J.; PATERSON, M. S.; TANIMOTO, S. L. Optimal packing and covering in the plane are NP-complete. **Inform Process Lett**, v. 12, p. 133–137, 1981. Citado na página 37.
- GOMES, A. M.; OLIVEIRA, J. F. Nesting irregular shapes with simulated annealing. In: **Ribeiro C (ed). Extended Abstracts of MIC1999-III Metaheuristics Internacional Conference**. Angra dos Reis, Brazil: [s.n.], 1999. p. 19–22. Citado na página 42.
- _____. A 2-exchange heuristic for nesting problems. **European Journal of Operational Research**, v. 141, p. 359–370, 2002. Citado nas páginas 24 e 37.
- _____. Solving irregular strip packing problems by hybridising simulated annealing and linear programming. **European Journal of Operational Research**, v. 171, p. 811–829, 2006. Citado na página 45.

- HECKMANN, R.; TLENGAUER. A simulated annealing approach to the nesting problem in the textile manufacturing industry. **Ann Opl Res**, v. 57, p. 103–133, 1995. Citado na página 40.
- JAKOBS, S. On genetic algorithms for the packing of polygons. **European Journal of Operational Research**, v. 88, p. 165–181, 1996. Citado na página 40.
- JONES, D. R. A fully general, exact algorithm for nesting irregular shapes. **Journal of Global Optimization**, v. 59, p. 367–404, 2013. Citado nas páginas 13, 24, 29, 30, 36 e 44.
- KALLRATH, J. Cutting circles and polygons from area-minimizing rectangles. **Journal of Global Optimization**, v. 43, p. 299–328, 2009. Citado nas páginas 17, 25, 46, 49, 51, 52, 56, 69, 70, 71, 73, 74, 77, 87 e 88.
- KONOPASEK, M. **Mathematical Treatments of Some Apparel Marking and Cutting Problems**. U.S. Department of Commerce Report 99-26-90857-10. U.S.: [s.n.], 1981. Citado na página 32.
- LEAO, A. A.; TOLEDO, F. M.; OLIVEIRA, J. F.; CARRAVILLA, M. A. A semi-continuous mip model for the irregular strip packing problem. **International Journal of Production Research**, v. 54, p. 712–721, 2016. Citado na página 43.
- LIAO, X.; MA, J.; OU, C.; LONG, F.; LIU, X. Visual nesting system for irregular cutting-stock problem based on rubber band packing algorithm. **Advances in Mechanical Engineering**, v. 8, p. 1–15, 2016. Citado nas páginas 17, 25, 42, 62, 66, 76 e 88.
- MAHADEVAN, A. **Optimisation in computer aided pattern packing**. Tese (Doutorado) — North Carolina State University, 1984. Citado na página 32.
- MARQUES, V. M.; BISPO, C. F.; SENTIEIRO, J. J. A system for the compaction of two-dimensional irregular shapes based on simulated annealing. **IEEE Transactions on Industrial Electronics, Control and Instrumentation**, v. 3, p. 1911–1916, 1991. Citado nas páginas 24 e 37.
- MARTINEZ-SYKORA, A.; ALVAREZ-VALDES, R.; BENNELL, J.; RUIZ, R.; TAMARIT, J. Matheuristics for the irregular bin packing problem with free rotations. **European Journal of Operational Research**, v. 258, p. 440 – 455, 2017. Citado nas páginas 24 e 45.
- MILENKOVIC, V.; DANIELS, K.; LI, Z. Placement and compactation of nonconvex polygons for clothing manufacturing. In: **Wang CA (Ed.) Fourth Canadian Conference on Computational Geometry**. St Johns's, Newfoundland: [s.n.], 1992. Citado na página 42.
- MLADENOVIC, N.; PLASTRIA, F.; UROSEVIC, D. Reformulation descent applied to circle packing problems. **Computers and Operations Research**, v. 32, p. 2419–2432, 2005. Citado na página 24.
- MUNDIM, L. R.; ANDRETTA, M.; QUEIROZ, T. A biased random key genetic algorithm for open dimension nesting problems using no-fit raster. **Expert Systems with Applications**, v. 81, p. 358–371, 2017. Citado nas páginas 13, 24, 30, 35, 46, 47, 63 e 64.
- NIELSEN, B. K. An efficient solution method for relaxed variants of the nesting problem. **Proceedings of the thirteenth Australasian symposium on Theory of computing**, v. 65, p. 123–130, 2007. Citado nas páginas 24 e 37.

- NOCEDAL, J.; WÄCHTER, A.; WALTZ, R. A. Adaptive barrier strategies for nonlinear interior methods. **SIAM Journal on Optimization**, v. 19, p. 1674–1693, 2009. Citado na página 61.
- OLIVEIRA, J. F.; GOMES, A. M.; FERREIRA, J. S. TOPOS - A new constructive algorithm for nesting problems. **Operations Research Spektrum**, v. 22, p. 263–284, 2000. Citado nas páginas 28, 37, 38 e 39.
- OLIVEIRA, J. F. C.; FERREIRA, J. A. S. Algorithms for nesting problems. In: _____. **Applied Simulated Annealing**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993. p. 255–273. ISBN 978-3-642-46787-5. Disponível em: <https://doi.org/10.1007/978-3-642-46787-5_13>. Citado nas páginas 24, 28, 37, 40 e 42.
- PREPARATA, F. P.; SHAMOS, M. I. Computational Geometry: An Introduction. **Springer-Verlag**, 1985. Citado na página 32.
- ROCHA, P.; RODRIGUES, R.; GOMES, A. M.; TOLEDO, F. M. B.; ANDRETTA, M. Two-Phase Approach to the Nesting problem with continuous rotations. **IFAC-PapersOnline**, v. 48, p. 501–506, 2015. Citado nas páginas 13, 24, 29, 30 e 44.
- RODRIGUES, M. O.; CHERRI, L. H.; MUNDIM, L. R. Mip models for the irregular strip packing problem: new symmetry breaking constraints. **ITM Web Conf.**, v. 14, p. 00005, 2017. Disponível em: <<https://doi.org/10.1051/itmconf/20171400005>>. Citado na página 44.
- SEGENREICH, S. A.; BRAGA, L. M. Optimal nesting of general plane figures: a Monte Carlo heuristical approach. **Computers and Graphics**, v. 10, p. 229–237, 1986. Citado na página 28.
- STOYAN, Y.; PANKRATOV, A.; ROMANOVA, T. Cutting and packing problems for irregular objects with continuous rotations: mathematical modelling and non-linear optimization. **Journal of the Operational Research Society**, v. 67, p. 786–800, 2016. Citado nas páginas 13, 17, 24, 25, 29, 30, 34, 45, 62, 66, 76 e 88.
- STOYAN, Y. G.; CHUGAY, A. Packing cylinders and rectangular parallelepipeds with distances between them. **European J. Oper. Res.**, v. 197, p. 446–455, 2008. Citado na página 44.
- STOYAN, Y. G.; NOVOZHILOVA, M. V.; KARTASHOV, A. V. Mathematical model and method of searching for a local extremum for the nonconvex oriented polygons allocation problem. **European Journal of Operational Research**, v. 92, p. 193–210, 1996. Citado na página 42.
- STOYAN, Y. G.; SCHEITHAUER, G.; GIL, N.; ROMANOVA, T. Phi-functions for complex 2d-objects. **4OR: Quartely Journal of the Belgian, French and Italian Operations Research Societies**, v. 2, p. 69–84, 2004. Citado nas páginas 29 e 34.
- STOYAN, Y. G.; TERNO, J.; SCHEITHAUER, G.; GIL, N.; ROMANOVA, T. Phi-functions for primary 2d-objects. **Studia Informatica Universalis**, v. 2, p. 1–32, 2001. Citado nas páginas 29 e 34.
- STOYAN, Y. G.; YASKOV, G. A mathematical model and solution method for the problem of packing various-sized circles into a strip. **European Journal of Operational Research**, v. 156, p. 590–600, 2004. Citado na página 24.
- STOYAN, Y. G.; YASKOV, G. N. Packing identical spheres into a cylinder. **International Transactions in Operational Research**, v. 17, p. 51–70, 2010. Citado na página 46.

STOYAN, Y. G.; ZLOTNIK, M. V.; CHUGAY, A. M. Solving an optimization packing problem of circles and non-convex polygons with rotations into a multiply connected region. **Journal of the Operational Research Society**, v. 63, p. 379–391, 2012. Citado nas páginas 14, 24, 25, 46, 67, 72 e 88.

SZABÓ, P. G.; MARKÓT, M. C.; CSENDES, T.; SPECHT, E.; CASADO, L. G.; GARCIA, I. **New approaches to circle packing in a square: with program codes**. [S.l.]: Optimization and its Applications, 2007. Citado na página 46.

TOLEDO, F. M. B.; 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. Citado nas páginas 24, 43 e 45.

Wächter, A.; Biegler, L. T. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. **Mathematical Programming**, v. 106, p. 25–57, 2006. Citado nas páginas 61 e 88.

_____. **COIN OR Project**. 2015. [Http://projects.coin-or.org/Ipopt](http://projects.coin-or.org/Ipopt). Citado na página 61.

WANG, H.; HUANG, W.; ZHANG, Q.; XU, D. An improved algorithm for the packing of unequal circles within a larger containing circle. **European Journal of Operational Research**, v. 141, p. 440–453, 2002. Citado na página 24.

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. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S037722170600292X>>. Citado na página 23.

ZOUTENDIJK, G. **Methods of Feasible Directions, A Study in Linear and Non-linear Programming**. [S.l.]: Elsevier, 1960. Citado nas páginas 44 e 46.

_____. Nonlinear programming, computational methods. **Integer and Nonlinear Programming**, v. 143, p. 37–86, 1970. Citado nas páginas 44 e 46.

PROCEDIMENTO PARA DETERMINAR OS PARÂMETROS DE UMA PARÁBOLA

Uma parábola é o lugar geométrico dos pontos de um plano equidistantes de uma determinada linha, chamada diretriz, e de um ponto fora dele, chamado foco (veja a Figura 59).

Figura 59 – Elementos de uma parábola.

Dados dois pontos fixos, há apenas uma parábola que os tem por vértice e foco, uma vez que a diretriz é fixada automaticamente como perpendicular à reta que passa por esses dois pontos e que está à mesma distância do vértice, do que está o vértice do foco. Assim, podemos calcular os parâmetros A, B, C, D, E e F da equação geral da parábola, $Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0$, a partir do vértice e do foco, (v_x, v_y) e (f_x, f_y) , respectivamente. Para isto, realizamos o seguinte procedimento.

1. Calculamos a distância focal.

A distância focal é a distância do vértice ao foco, que por sua vez é a distância do vértice

à diretriz, esta é dada por

$$\sqrt{(f_x - v_x)^2 + (f_y - v_y)^2}.$$

2. Encontramos a equação do eixo de simetria.

O eixo de simetria é a reta que passa pelo vértice e pelo foco. Sua equação é dada por:

$$y = \frac{(f_y - v_y)}{(f_x - v_x)}(x - v_x) + v_y. \quad (\text{A.1})$$

Se $f_x = v_x$, a equação do eixo de simetria é $x = f_x$.

3. Calculamos o ponto de interseção do eixo de simetria com a reta diretriz.

Seja (d_x, d_y) o ponto de interseção do eixo de simetria com a reta diretriz. A distância do vértice a este ponto (quer dizer à reta diretriz) é a mesma que do vértice ao foco, a distância focal, portanto:

$$(d_x - v_x)^2 + (d_y - v_y)^2 = (f_x - v_x)^2 + (f_y - v_y)^2. \quad (\text{A.2})$$

Por outro lado, como (d_x, d_y) é um ponto do eixo de simetria que satisfaz a equação da reta (A.1), temos

$$d_y = \frac{(f_y - v_y)}{(f_x - v_x)}(d_x - v_x) + v_y. \quad (\text{A.3})$$

Ao substituir d_y , da equação (A.3), na equação (A.2) e desenvolvendo o lado esquerdo desta equação obtemos

$$d_x^2 - 2v_x d_x + v_x^2 - (f_x - v_x)^2 = 0.$$

Desta forma, podemos calcular o valor de d_x , aplicando a fórmula geral para obter raízes:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a},$$

com $a = 1$, $b = -2v_x$ e $c = v_x^2 - (f_x - v_x)^2$. Assim,

$$d_x = \frac{2v_x + 2(f_x - v_x)}{2} = f_x.$$

ou

$$d_x = \frac{2v_x - 2(f_x - v_x)}{2} = 2v_x - f_x.$$

Concluimos que $(d_x, d_y) = (f_x, 2v_y - f_y)$, no caso que a equação do eixo de simetria seja $x = f_x$, ou $(d_x, d_y) = (2v_x - f_x, 2v_y - f_y)$, caso contrário.

4. Encontramos a equação da reta diretriz

A reta diretriz é a reta perpendicular ao eixo de simetria que passa pelo ponto (d_x, d_y) . Se a equação do eixo de simetria é $x = f_x$, então a equação da reta diretriz é $y = d_y$; caso contrário, para determinar a equação da reta diretriz usamos a equação fundamental da reta (equação da reta dado um ponto e o coeficiente angular).

Como a reta diretriz é perpendicular ao eixo de simetria seu coeficiente angular é dado por

$$\frac{v_x - f_x}{f_y - v_y},$$

assim a equação da reta diretriz é dada por

$$y = \frac{v_x - f_x}{f_y - v_y}(x - 2v_x + f_x) + (2v_y - f_y). \quad (\text{A.4})$$

5. Propomos a definição de uma parábola como o conjunto de pontos que estão equidistantes do foco e da diretriz.

Seja (q_x, q_y) o ponto de interseção da reta diretriz e uma reta L paralela ao eixo de simetria, que passa pelo ponto (p_x, p_y) . A equação da reta L é dada por:

$$y = \frac{(f_y - v_y)}{(f_x - v_x)}(x - p_x) + p_y. \quad (\text{A.5})$$

Suponha que (p_x, p_y) é um ponto da parábola, isso quer dizer que ele está à mesma distância de (f_x, f_y) e de (q_x, q_y) , portanto

$$(p_x - f_x)^2 + (p_y - f_y)^2 = (p_x - q_x)^2 + (p_y - q_y)^2. \quad (\text{A.6})$$

Como (q_x, q_y) é um ponto da reta L , ele satisfaz a equação (A.5) e, portanto,

$$q_y = \frac{(f_y - v_y)}{(f_x - v_x)}(q_x - p_x) + p_y. \quad (\text{A.7})$$

Ao substituir q_y na equação (A.6), obtemos

$$(p_x - f_x)^2 + (p_y - f_y)^2 - (p_x - q_x)^2 \left(1 + \frac{(f_y - v_y)^2}{(f_x - v_x)^2} \right) = 0. \quad (\text{A.8})$$

Ao desenvolver os termos quadráticos obtemos:

$$\left(-\frac{(f_y - v_y)^2}{(f_x - v_x)^2} \right) p_x^2 + p_y^2 - 2p_x f_x - 2p_y f_y + f_x^2 + f_y^2 + (2p_x q_x - q_x^2) \left(1 + \frac{(f_y - v_y)^2}{(f_x - v_x)^2} \right) = 0. \quad (\text{A.9})$$

Resta agora determinar o valor de q_x . Se o eixo de simetria é paralelo ao eixo da ordenada, $q_x = p_x$; caso contrário, calculamos as coordenadas do ponto (q_x, q_y) igualando a equação da reta diretriz (A.4) à equação da reta L (A.5), assim

$$\frac{v_x - f_x}{f_y - v_y}(q_x - 2v_x + f_x) + (2v_y - f_y) = \frac{(f_y - v_y)}{(f_x - v_x)}(q_x - p_x) + p_y.$$

Assim,

$$q_x = \frac{(f_x - v_x)^2(2v_x - f_x) + (2v_y - f_y)(f_x - v_x)(f_y - v_y)}{(f_x - v_x)^2 + (f_y - v_y)^2} + \frac{(f_y - v_y)^2}{(f_x - v_x)^2 + (f_y - v_y)^2}p_x - \frac{(f_x - v_x)(f_y - v_y)}{(f_x - v_x)^2 + (f_y - v_y)^2}p_y.$$

Substituindo q_x na equação (A.9), deduzimos que a equação da parábola é dada por

$$Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0,$$

com

$$A = -(f_y - v_y)^2,$$

$$B = 2(f_x - v_x)(f_y - v_y),$$

$$C = -(f_x - v_x)^2,$$

$$D = -2((f_x - v_x)(f_y - v_y)(2v_y - f_y) - 2(f_x - v_x)^3 - f_x(f_y - v_y)^2),$$

$$E = -2((f_x - v_x)(f_y - v_y)(2v_x - f_x) - 2(f_y - v_y)^3 - f_y(f_x - v_x)^2),$$

e

$$F = ((f_x - v_x)(2v_x - f_x) + (f_y - v_y)(2v_y - f_y))^2 - (f_x^2 + f_y^2)((f_x - v_x)^2 + (f_y - v_y)^2).$$