Universidade Federal do Rio Grande - FURG
Programa de Pós-Graduação em Modelagem Computacional

Cássia Cris Beckel

Heurística construtiva para o empacotamento de elipses tangentes em um polígono de n lados

Rio Grande
2013

Cássia Cris Beckel

Heurística construtiva para o empacotamento de elipses tangentes em um polígono de n lados

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional da FURG, como requisito parcial para a obtenção do título de MESTRE em Modelagem Computacional.

Orientador: Leonardo Ramos Emmendorfer

Orientador - Centro de Ciências Computacionais, C3, FURG

Co-orientadora: Neuza Terezinha Oro

Co-orientadora - Instituto de Ciências Exatas e Geociências, ICEG, UPF

Rio Grande

2013

Cássia Cris Beckel

Heurística construtiva para o empacotamento de elipses tangentes em um polígono de n lados

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional da FURG, como requisito parcial para a obtenção do título de MESTRE em Modelagem Computacional.

Aprovado em 25 de Fevereiro de 2013

BANCA EXAMINADORA

Leonardo Ramos Emmendorfer

Orientador - Centro de Ciências Computacionais, C3, FURG

Neuza Terezinha Oro

Co-orientadora - Instituto de Ciências Exatas e Geociências, ICEG, UPF

Aline Brum Loreto

Examinadora Externa - Centro Desenvolvimento Tecnológico, CDTec, UFPel

---

Catia Maria dos Santos Machado

Examinadora - Instituto de Matemática, Estatística e Física, IMEF, FURG

Dedicatória

Dedico esta dissertação a minha família e a todos que me apoiaram e acreditaram no meu potencial, especialmente aos amigos, colegas do mestrado e professores que contribuíram no decorrer desta caminhada.

"A Ciência, pelo caminho da exatidão,
só tem dois olhos: a Matemática e a lógica."

De Morgan, Augustus.

Agradecimentos

Ao final desta longa caminhada, ressalto algumas pessoas importantes cujo agradecimento traduz o singelo muito obrigada.

Agradeço a minha família pela paciência e compreensão nos momentos de ausência, em especial aos meus pais Rudinei e Silvana Beckel e a minha irmã Camila. Vocês são os responsáveis pela minha educação, foram com certeza os meus primeiros mestres. Agradeço ao Otávio, que mesmo pequeno sentia cada despedida.

Aos meus avós Vilmar e Noeli Ozelame, pelo incentivo, preocupação e por acreditarem nos meus objetivos.

À minha grande amiga Martiela Vaz de Freitas, pessoa incrível que tive o prazer de conhecer e conviver durante o mestrado. Agradeço pela amizade, apoio e inúmeras divagações cotidianas.

À professora Neuza Terezinha Oro, que além de orientadora na graduação e co-orientadora do mestrado, é uma grande amiga com um raciocínio inacreditável e que sempre se fez presente. Agradeço pela sua disponibilidade, dedicação e extremo comprometimento durante todo o período do trabalho.

Ao meu orientador e professor Leonardo Ramos Emmendorfer pelas críticas construtivas, apoio e orientação. Agradeço ainda pelo constante incentivo às conquistas profissionais.

Aos amigos: Cícero Matuella Moreira, Ana Carolina Lopes, Cassia Oliveira, Emilia Taffarel e Marlon Vinícius Machado por tornarem cada instante de convivência um momento de paz, felicidade e amizade.

Aos colegas e amigos da Furg: Carlos Quadros, Fábio Aiub, Ruth Brum, Ezequiel Gibbon Gautério e Juliana Martins pelos momentos alegres e convivência acadêmica.

À Universidade Federal do Rio Grande, bem como aos professores do Programa de Pós-Graduação em Modelagem Computacional. Agradeço a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) fomentadora de bolsa para auxílio financeiro.

Resumo

Problemas de corte e empacotamento estão presentes em diversos setores da indústria, e o estudo destes problemas propicia oportunidades de colaboração entre os setores acadêmicos e industrial, com vistas a que se obtenham benefícios para ambos, contribuindo para a sociedade como um todo. Entre os setores industriais nos quais surgem problemas de corte e empacotamento estão as indústrias têxtil, automotiva, portuária, lapidária, entre outras. O presente trabalho tem como objetivo elaborar uma metodologia analítica e computacional com a qual seja possível encontrar uma solução viável para o problema de empacotamento de elipses, sendo idênticas ou não, sem sobreposição e tangentes a cada vértice e quadrante de uma elipse inicial inscrita em um polígono irregular de n lados. A metodologia analítica e computacional desenvolvida visa obter a maximização da área total das elipses empacotadas e a minimização do tempo de processamento computacional. Destaca-se a aplicabilidade das transformações em $\mathbb{R}^2$ para obter as novas equações paramétricas das elipses com centro deslocado da origem e rotacionadas em relação ao sistema de eixos cartesianos original. A heurística que realiza a verificação da inscrição de cada elipse, baseia-se em uma modificação da função *inpolygon* do *software* Matlab [34], de maneira que garante o empacotamento total das elipses no polígono. Para validar a heurística construtiva utilizaram-se 7 polígonos e com os resultados obtidos em cada simulação foi possível encontrar a função exponencial, através de um ajuste de curva, que descreve o comportamento da simulação.

Palavras-chaves: Problemas de corte e empacotamento. Heurística construtiva. Elipses tangentes.

Sumário

Lista de Figuras viii

Lista de Tabelas xiii

Lista de Símbolos xiv

Lista de Siglas xv

1 Introdução 1

1.1 Problemas de corte e empacotamento na indústria 1

1.1.1 Classificação dos problemas de corte e empacotamento 3

1.2 Tecnologia 3D Gemas 6

1.3 Introdução a programação com o *Software* MatLab 7

1.4 Justificativas para o trabalho 8

1.5 Objetivo geral do trabalho 10

1.5.1 Objetivos específicos do trabalho 10

1.6 Organização da Dissertação 11

2 Revisão bibliográfica 13

2.1 Trabalhos relacionados ao posicionamento de formas 16

2.2 Métodos para solucionar problemas de posicionamento de formas em polígonos 19

3 Formulação matemática do problema de empacotamento de elipses 24

3.1 Generalização da equação paramétrica de uma elipse 24

3.2 Coordenadas locais de ε inscrita em P 28

3.3 Generalização da otimização de elipses tangentes 29

3.4 Estratégias de otimização 31
3.4.1 Otimização da função objetivo 31
3.4.2 Etapa 1 - Restrições para otimização de ε 32
3.4.3 Etapa 2 - Restrições para otimização das elipses tangentes aos vértices V_i de ε 33
3.4.4 Etapa 3 - Restrições para otimização das elipses tangentes a cada quadrante Q_i de ε 34
3.4.5 Etapa 4 - Restrição de tangência 35
3.4.6 Etapa 5 - Condição adicional para o teste de convexidade em relação ao semi-eixo maior das elipses. 36

4 Adaptação do método de otimização de Hooke e Jeeves 40

4.1 Descrição da busca exploratória 47
4.2 Descrição da busca padrão utilizando o método de Hooke e Jeeves 50
4.3 Parâmetros obtidos com o posicionamento de ε 51

5 Metodologia analítica para o empacotamento de elipses tangentes aos vértices de ε 53

5.1 Cálculo das coordenadas dos vértices da elipse inicial 56
5.2 Posicionamento das elipses tangentes aos vértices da elipse inicial 60
5.2.1 Elipse posicionada no vértice $B1$ de ε 61
5.2.2 Elipse posicionada no vértice $C1$ de ε 65
5.2.3 Elipse posicionada no vértice $A1$ de ε 68
5.2.4 Elipse posicionada no vértice $D1$ de ε 71

6 Metodologia analítica para o empacotamento de elipses tangentes aos quadrantes de ε 75

6.1 Equações paramétricas resultantes da aplicação de simetrias em $\mathbb{R}^2$ 78
6.1.1 Elipse posicionada no 1º quadrante de ε 82

6.1.2 Elipse posicionada no 2º quadrante de ε 87

6.1.3 Elipse posicionada no 3º quadrante de ε 91

6.1.4 Elipse posicionada no 4º quadrante de ε 95

7 Verificação da inscrição das elipses no polígono *P* **99**

8 Simulações da heurística construtiva de empacotamento **105**

8.1 Experimentos numéricos 105

9 Análise dos resultados obtidos com as simulações **115**

10 Conclusão **121**

10.1 Considerações finais 121

10.2 Contribuições do trabalho 123

10.3 Propostas para trabalhos futuros 123

Referências Bibliográficas **126**

Apêndices **132**

A Projeto de lapidação virtual **133**

B Pseudo-código do Algoritmo Heuristic **135**

C Algoritmo de ajuste de curva - Exponencial **141**

D Simulação - Polígono com 23 vértices **142**

Lista de Figuras

1.1 Lapidação de gemas - um problema de corte no setor lapidário. Fonte: 3D Gemas. 2

1.2 Peças fabricadas em série. Fonte: J.A.S. Metalúrgica. 2

1.3 Concreto convencional e concreto obtido com MEC. Fonte: [20] 3

1.4 Problema de corte e empacotamento. Fonte: [1] 3

1.5 Problema unidimensional para minimizar a perda *P*. Fonte: [22] 4

1.6 Problema do corte e empacotamento bidimensional. Fonte: [17] 4

1.7 Problema de empacotamento tridimensional em *containers*. Fonte: [38] . . 5

1.8 Problema de alocação de berços em portos. Fonte: Terminal Marítimo da Ultrafertil. 5

2.1 Empacotamento de círculos em um círculo maior. Fonte: [21]. 15

2.2 Posicionamento de formas em um recipiente de dimensões fixas. Fonte: [32] 16

2.3 Posicionamento de duas elipses em um dado *container*. Fonte:[48] 17

2.4 Base do modelo coração em um dado *container*. Fonte: [48] 17

2.5 Posicionamento geral no polígono. Fonte: [25] 18

2.6 Posicionamento de círculos com PCSN. Fonte: [25] 18

2.7 Método de otimização Hooke e Jeeves. Fonte: [4] 21

3.1 Elipse posicionada na origem. Fonte: Elaborado pela autora. 25

3.2 Elipse transladada em relação a origem. Fonte: Elaborado pela autora. . . 25

3.3 Rotação de um ponto no sistema cartesiano. Fonte: Elaborado pela autora. 26

3.4 Elipse com centro transladado em relação a origem e com rotação R. Fonte: Elaborado pela autora. 27

3.5 Elipse inicial interior a um polígono dado. Fonte: Elaborado pela autora. . 29

3.6 Representação da redução gradativa do semi-eixo maior em função de α. Fonte: Elaborado pela autora. 39

4.1 Protótipos da inscrição de um modelo de lapidação oval em um gema. Fonte: 3D Gemas. 41

4.2 Contorno com seleção de pontos que descreve o maior plano da gema. Fonte: [35] 43

4.3 Distância do centro ao vértice do polígono. Fonte: Elaborado pelo autora. . 44

4.4 Rotação da elipse inscrita no polígono irregular. Fonte: Elaborado pela autora. 50

4.5 Inscrição da elipse inicial em um polígono P. Fonte: Elaborado pela autora. 52

5.1 Composição de triângulos pela Fórmula de Heron. Fonte: Elaborado pela autora. 54

5.2 (a) Rotação: vetor linha - (b) Rotação: vetor coluna. Fonte: Elaborado pela autora. 57

5.3 Representação gráfica do vértice A1 de ε. Fonte: Elaborado pela autora. . 58

5.4 Representação gráfica do vértice B1 de ε. Fonte: Elaborado pela autora. . 58

5.5 Representação gráfica do vértice C1 de ε. Fonte: Elaborado pela autora. . 59

5.6 Representação gráfica do vértice D1 de ε. Fonte: Elaborado pela autora. . 59

5.7 Representação geométrica dos vértices e do centro de ε. Fonte: Elaborado pela autora. 60

5.8 As áreas em destaque representam a sobreposição de elipses tangentes a ε. Fonte: Elaborado pela autora. 60

5.9 Elipses tangentes a ε com efeito de dimensionamento.Fonte: Elaborado pela autora. 61

5.10 Elipse tangente ao vértice B1 de ε. Fonte: Elaborado pela autora. 63

5.11 Pontos aleatórios em cada quadrante. Fonte: Elaborado pela autora. 64

5.12 Pontos aleatórios no 1º e 4º quadrante. Fonte: Elaborado pela autora. . . . 65

5.13 Elipse tangente ao vértice C1 de ε. Fonte: Elaborado pela autora. 67

5.14 Pontos aleatórios no 1º e 2º quadrante. Fonte: Elaborado pela autora. . . . 68

5.15 Elipse tangente ao vértice A1 de ε. Fonte: Elaborado pela autora. 70

5.16 Pontos aleatórios no 2º e 3º quadrantes. Fonte: Elaborado pela autora. . . 71

5.17 Elipse tangente ao vértice D1 de ε. Fonte: Elaborado pela autora. 73

5.18 Pontos aleatórios no 3º e 4º quadrante. Fonte: Elaborado pela autora. . . . 74

6.1 Blocos de Lego criados por Godtfred. Fonte: [13] 76

6.2 Posicionamento das circunferências em cada quadrante. Fonte: [13] 76

6.3 Posicionamento das circunferências com duas projeções internas. Fonte: [13] 76

6.4 Posicionamento das circunferências com cruzamento de diagonais. Fonte: [13]. 77

6.5 Pontos de tangência definidos em cada quadrante. Fonte: Elaborado pela autora. 77

6.6 Reflexão em torno do eixo das ordenadas. Fonte: [45]. 79

6.7 Reflexão em torno da origem. Fonte: [45]. 80

6.8 Reflexão em torno do eixo das abscissas. Fonte: [8]. 81

6.9 Simetrias em $\mathbb{R}^2$. Fonte: Elaborado pela autora. 82

6.10 Transformações aplicadas para obtenção de pontos de tangência. Fonte: Elaborado pela autora. 82

6.11 Ponto de tangência com direção de busca definida no 1º quadrante. Fonte: Elaborado pela autora. 83

6.12 Elipse tangente ao 1º quadrante de ε. Fonte: Elaborado pela autora. 86

6.13 Elipse 5 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora. 86

6.14 Ponto de tangência $P6$ resultado da transformação aplicada. Fonte: Elaborado pela autora. 87

6.15 Ponto de tangência com direção de busca definida no 2º quadrante. Fonte: Elaborado pela autora. 88

6.16 Elipse tangente ao 2º quadrante de ε. Fonte: Elaborado pela autora. 90

6.17 Elipse 6 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora. 90

6.18 Ponto de tangência P7 obtido com a transformação aplicada. Fonte: Elaborado pela autora. 91

6.19 Ponto de tangência com direção de busca definida no 3º quadrante. Fonte: Elaborado pela autora. 92

6.20 Elipse tangente ao 3º quadrante de ε. Fonte: Elaborado pela autora. 94

6.21 Elipse 7 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora. 94

6.22 Ponto de tangência P8 resultado da transformação aplicada. Fonte: Elaborado pela autora. 95

6.23 Ponto de tangência com direção de busca definida no 4º quadrante. Fonte: Elaborado pela autora. 95

6.24 Elipse tangente ao 4º quadrante de ε. Fonte: Elaborado pela autora. 97

6.25 Elipse 8 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora. 98

7.1 Heurística construtiva: elipses posicionadas nos vértices e quadrantes de ε sem a verificação da inscrição. Fonte: Elaborado pela autora. 99

7.2 Posição relativa do ponto em relação ao polígono P. Fonte: Adaptado pela autora [40]. 100

7.3 Função *inpolygon* aplicada ao problema de um retângulo. Fonte: [34]. . . . 102

7.4 Heurística construtiva: exemplo da verificação da inscrição em ε. Fonte: Elaborado pela autora. 103

7.5 Tempo de processamento x número de arestas. Fonte: [40]. 103

8.1 Empacotamento de elipses tangentes a uma elipse inicial em um polígono de n lados. Fonte: Heurística construtiva. 114

9.1 Simulação mostrada no Capítulo 8 com ajuste de curva . Fonte: Elaborado pela autora. 117

9.2 Simulação realizada e o ajuste de curva . Fonte: Elaborado pela autora. . . 119

9.3 Elipses empacotadas restritas ao contorno com concavidades. Fonte: Elaborado pela autora. 120

Lista de Tabelas

8.1 Legenda utilizada nas simulações na descrição da tangência 106

8.2 Simulação numérica - Posição inicial 107

8.3 Simulação numérica - Iteração 1 107

8.4 Simulação numérica - Iteração 2 108

8.5 Simulação numérica - Iteração 3 109

8.6 Simulação numérica - Iteração 4 109

8.7 Simulação numérica - Iteração 5 110

8.8 Simulação numérica - Iteração 6 110

8.9 Simulação numérica - Iteração 7 111

8.10 Simulação numérica - Iteração 8 111

8.11 Simulação numérica - Iteração 9 112

8.12 Simulação numérica - Iteração 10 113

9.1 Resultados obtidos com a heurística construtiva 116

9.2 Séries de dados utilizada no ajuste de curvas. 116

9.3 Séries de dados utilizada no ajuste de curvas. 118

Lista de Símbolos

$\boldsymbol{\varepsilon}$ - Elipse inicial

$\boldsymbol{a}$ - Semi-eixo maior da elipse (mm)

$\boldsymbol{b}$ - Semi-eixo menor da elipse (mm)

$\boldsymbol{e}$ - Excentricidade

$\boldsymbol{A}$ - Área da elipse (mm^2)

$\boldsymbol{\sigma}$ - Ângulo de rotação no sentido anti-horário (^{c})

sin - Seno de um ângulo

Lista de Siglas

PCE	-	Problema de Corte e Empacotamento
Tp	-	Tempo de processamento de cada elipse (s)
Ttp	-	Tempo total de processamento (s)
I	-	Número de iterações realizadas
Np	-	Número de pontos do polígono
At	-	Área total ocupada pelas elipses (mm^2)
Ap	-	Área do polígono P (mm^2)
Ar	-	Área restante (mm^2)
Pr	-	Percentual de área restante %

1 Introdução

Neste capítulo é apresentada a introdução à heurística desenvolvida nesta dissertação. É realizada uma contextualização aos problemas de corte e empacotamento, ressaltando a aplicabilidade na indústria lapidária e em outros ramos industriais. As principais classificações dos problemas de corte e empacotamento são expostas e por fim, apresenta-se o software Matlab e a Tecnologia 3D Gemas base desse estudo. Finalmente, são relatados os objetivos e justificativas desta pesquisa, bem como, a organização dos demais capítulos da dissertação.

1.1 Problemas de corte e empacotamento na indústria

Com os grandes avanços tecnológicos e científicos que estão disponíveis em nível mundial, cada vez mais surge a necessidade de solucionar problemas de corte e empacotamento como forma de propiciar um maior desenvolvimento industrial.

A modelagem computacional, que tem por base um problema científico e faz uso da construção de modelos e simulação de soluções viáveis, geralmente baseia-se em uma aplicação resultando em códigos computacionais que obtenham soluções precisas para os mais variados problemas industriais [5].

Nesse sentido, surgiu necessidade de encontrar tais soluções para problemas de corte e empacotamento *(PCEs)* em vários ramos industriais como: indústria têxtil, indústria automotiva, industria portuária, indústria lapidária, entre outras. Conforme [1], o reverso dos problemas de corte são os problemas de empacotamento, os quais são igualmente essenciais para o planejamento de operações logísticas da indústria, como a armazenagem, movimentação ou transporte de itens produzidos.

O problema de corte consiste basicamente em cortar unidades de peças maiores para a produção de unidades de peças menores, com quantidades e tamanhos específicos,

de maneira a otimizar uma determinada função objetivo, como a de minimizar perdas, custos e o total de objetos a serem cortados, visando desta forma maximizar os lucros de produção [12]. Na Figura 1.1, observa-se um problema de corte presente na indústria lapidária onde, a partir de uma determinada gema[1], busca-se extrair o modelo de lapidação que resulte no maior aproveitamento volumétrico da superfície.

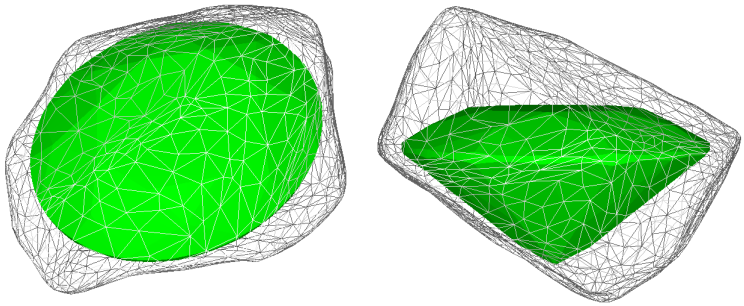


Figura 1.1: Lapidação de gemas - um problema de corte no setor lapidário. Fonte: 3D Gemas.

O problema de corte nas indústrias metalúrgicas ocorre principalmente na fabricação de peças em série. Nesses casos como mostra a Figura 1.2, por exemplo, são cortadas várias peças idênticas de uma determinada chapa de aço.

Figura 1.2: Peças fabricadas em série. Fonte: J.A.S. Metalúrgica.

No setor de aplicação estrutural de concreto, uma das áreas de concentração de estudos em engenharia civil é a de problemas de empacotamento *(Packing Problem)*. Dentro deste contexto, destaca-se a utilização do método de empacotamento compressível *(MEC)*. O *MEC* é uma ferramenta de dosagem que possibilita a seleção e otimização dos

[1]Segundo o *Gemological Institute of America* (2009) "gemas são minerais ou materiais orgânicos, utilizados como adorno e que possuem beleza, raridade e durabilidade".

constituintes do concreto através do empacotamento e visa proporcionar uma melhora na qualidade final e durabilidade do concreto. Segundo [20] a longo prazo, a durabilidade elevada do material minimizará os custos de manutenção, reparo e substituição da estrutura existente por uma nova. Na Figura 1.3 verifica-se uma mistura sem o devido tratamento e a mesma após o método de empacotamento compressível:

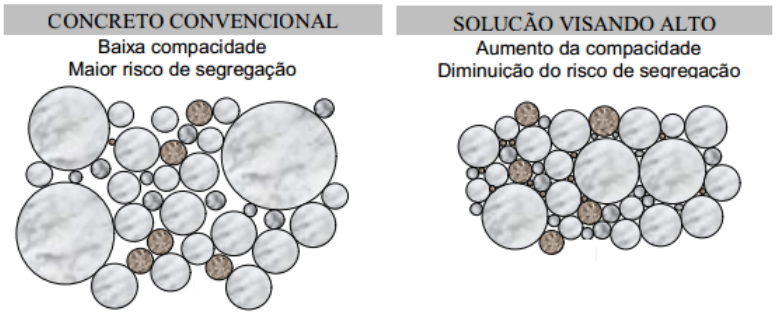


Figura 1.3: Concreto convencional e concreto obtido com MEC. Fonte: [20]

Os problemas de empacotamento consistem em alocar ou armazenar uma quantidade x de peças menores em espaços maiores, tais como *containers*, caixas, adegas, etc., de modo que se maximize o aproveitamento de espaço disponível. É possível perceber como os problemas de corte e empacotamento estão altamente correlacionados, pois a ideia de cortar peças pequenas de objetos grandes é equivalente a ideia de alocá-las em espaços maiores [1], como mostra a Figura 1.4. Dessa forma os dois problemas atualmente são estudados de maneira conjunta.

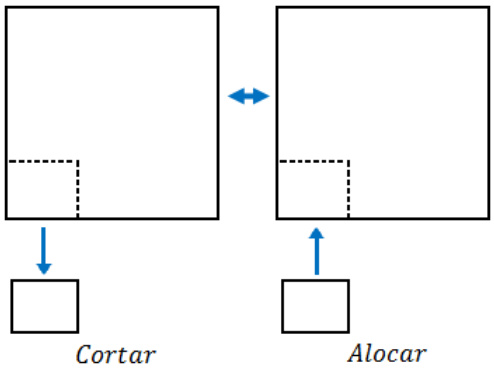


Figura 1.4: Problema de corte e empacotamento. Fonte: [1]

1.1.1 Classificação dos problemas de corte e empacotamento

Na classificação dos problemas de corte e empacotamento leva-se em consideração às dimensões que são relevantes durante o procedimento de corte ou empaco-

tamento. Segundo [17], a seguinte associação é feita: por causa do papel fundamental desempenhado pelos padrões de corte e sua natureza de combinações geométricas (combinar itens sobre objetos), pode-se dizer que os Problemas de Corte e Empacotamento são "problemas de combinações geométricas" e podem ser divididos em problemas com dimensões espaciais e não-espaciais.

Os problemas com dimensões espaciais envolvem objetos e itens que podem ter uma, duas ou três dimensões do Espaço Euclidiano de acordo com o processo de corte que se deseja aplicar. Dessa forma, classificam-se os problemas de corte de acordo com as suas características:

- **Problema Unidimensional** - O problema de corte e empacotamento unidimensional é definido como um problema de otimização combinatória, levando em consideração no processo apenas uma dimensão (comprimento) do objeto. Esses problemas são facilmente encontrados em indústrias de aço na fabricação de bobinas e na industria de papelão com bobinas de papel [22] como pode ser visto na Figura 1.5:

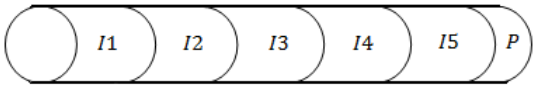


Figura 1.5: Problema unidimensional para minimizar a perda P. Fonte: [22]

- **Problema Bidimensional** - O problema de corte e empacotamento bidimensional assim como o unidimensional é um problema de otimização combinatória porém leva-se em consideração duas dimensões do Espaço Euclidiano (comprimento e largura). Problemas dessa natureza são encontrados em indústrias de móveis, setor metalúrgico e posicionamento de formas em polígonos.

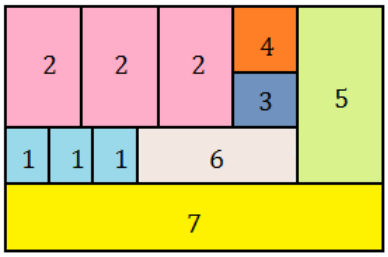


Figura 1.6: Problema do corte e empacotamento bidimensional. Fonte: [17]

- **Problema Tridimensional** - Um problema de corte ou empacotamento é considerado tridimensional quando as três dimensões do objeto (largura, comprimento e altura) são levados em consideração durante o processo, buscando alocar objetos espaciais dentro de objetos maiores de modo a minimizar os espaços ociosos.

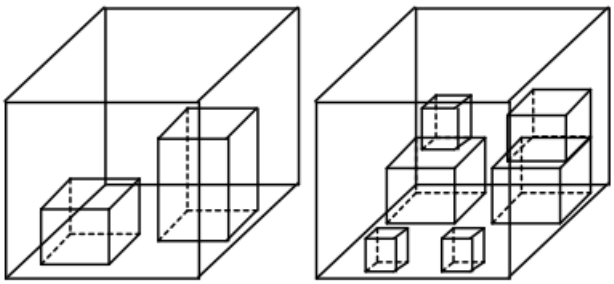


Figura 1.7: Problema de empacotamento tridimensional em *containers*. Fonte: [38]

Existem ainda os problemas multi-dimensionais ($n > 3$ *dimensões*) geralmente associados aos problemas de alocações de tarefas. Esse tipo de problema está inserido na classe dos problemas de dimensões não-espaciais na concepção de [17] e podem ser definidos como problemas abstratos, pois as dimensões levadas em consideração no decorrer da busca pela solução, não são dimensões físicas ou palpáveis.

O problema de alocação de berços em portos, conforme mostra a Figura 1.8 esta enquadrado nesse tipo de problema multi-dimensional, onde a janela de tempo passa a fazer parte do problema.

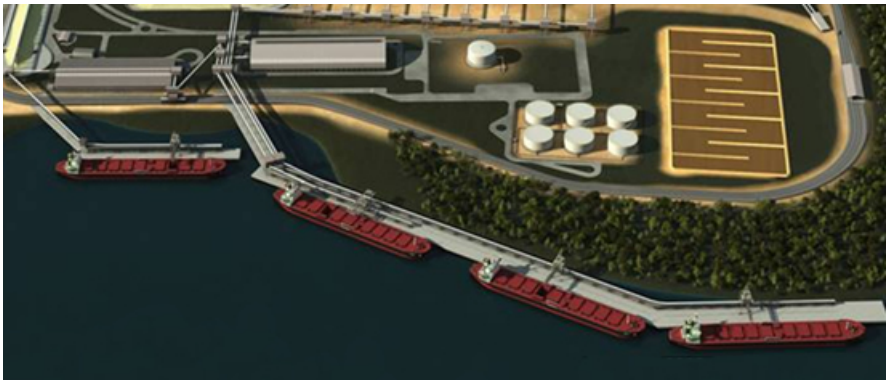


Figura 1.8: Problema de alocação de berços em portos. Fonte: Terminal Marítimo da Ultrafertil.

O empacotamento de elipses tangentes em um polígono de n lados é caracterizado como um problema bidimensional, levando em consideração os semi-eixos maior e menor de cada elipse alocada.

1.2 Tecnologia 3D Gemas

No setor industrial destaca-se o grande desenvolvimento do setor de lapidação e exportação de gemas do município de Soledade. Esse desenvolvimento propiciou a criação de várias linhas de pesquisa na área de tecnologias para a lapidação e aperfeiçoamento de técnicas visando melhorar a qualidade dos artefatos constituídos de gemas encontradas em estado bruto.

Dentre as tecnologias desenvolvidas através de pesquisas, cita-se a Tecnologia 3D Gemas como resultado do projeto de pesquisa "Digitalização 3D de gemas de pedras preciosas com *software* CAD[2] de apoio ao projeto de lapidação", que posteriormente passou a ser chamado de "Projeto 3D Gemas".

Nesse sentido, o Projeto 3D Gemas desenvolvido pelo Centro Tecnológico de Pedras, Gemas e Jóias do Rio Grande do Sul, localizado em Soledade, juntamente com a Universidade de Passo Fundo e Universidade Federal do Rio Grande do Sul propõe a Tecnologia 3D Gemas. Essa tecnologia é um conjunto de técnicas e procedimentos desenvolvidos com objetivo de auxiliar o processo de lapidação facetada de gemas coradas, que consiste em aplicar modelos de lapidação em que as gemas são totalmente limitadas por superfícies planas, visando encontrar para cada gema o projeto de lapidação que resulte no maior valor agregado, considerando volume e brilho.

Segundo [9] a referida tecnologia trabalha no desenvolvimento do *Software* Otimizador 3D Gemas, cujo propósito é desenvolver um algoritmo capaz de indicar o melhor aproveitamento volumétrico através da inserção de um modelo virtual de lapidação que não passe por uma solução trivial. Inicialmente, uma versão funcional do *software* está sendo desenvolvida, sendo que é capaz de inscrever apenas um modelo de lapidação no formato redondo ou formato oval para qualquer gema digitalizada.

A inscrição desses modelos de lapidação tem por padrão o posicionamento de elipses ou circunferências que formam a base dos modelos de lapidação. Alguns problemas de posicionamento de formas semelhantes são provenientes de um grande número de indústrias, como as de fabricação de peças automotivas, onde peças pré-definidas devem ser extraídas de chapas de metal minimizando o desperdício de material. Outro exemplo, são as indústrias de vestuário que buscam produzir o maior número de componentes a

[2]Os sistemas CAD - *Computer Adied Design* (projeto auxiliado por computador) auxiliam na criação, modificação e otimização de um projeto

partir de uma peça de tecido.

Neste trabalho, pretende-se desenvolver uma metodologia analítica e computacional capaz de encontrar a solução viável para o problema de empacotamento de múltiplas elipses tangentes entre si num polígono irregular de n lados, geradas a partir de uma elipse inicial obtida através do método de otimização irrestrita de Hooke e Jeeves, cuja modelagem matemática e computacional é descrita em [5] e [6]. Descreve ainda a utilização de simetrias em $\mathbb{R}^2$ e suas implicações na busca pelos pontos de tangência e uma heurística baseada na modificação da função *inpolygon* do MatLab que realiza a verificação de pontos interiores ou não ao polígono utilizado. As elipses devem ser posicionadas de modo que seja obtida a maximização da área total das elipses empacotadas, não havendo sobreposição de formas e sem que essas ultrapassem os limites do contorno da gema, considerando que o processo seja realizado em um tempo computacional aceitável de acordo com a necessidade do lapidário.

1.3 Introdução a programação com o *Software* MatLab

Para as implementações computacionais da heurística desenvolvida para empacotar elipses em polígonos irregulares de n lados utilizou-se o *software* MatLab (MATrix LABoratory) [34] sob licença acadêmica[3].

De acordo com [36] esse *software* é interativo e de alta performance voltado para o cálculo numérico, integrando análise numérica, cálculo com matrizes, processamento de sinais e construção de gráficos. Possui um ambiente fácil de usar, onde os problemas e soluções são expressos somente como eles são escritos matematicamente, ao contrário de outras linguagens de programação tradicional.

O *software* MatLab foi implementado em *C* (linguagem de programação) e oferece uma ampla biblioteca de funções pré-definidas, as quais são usadas para resolver determinados problemas tais como os de otimização, manipulação algébrica, redes neurais, processamento de sinais, simulação de sistemas dinâmicos, entre outros.

[3]Serial Number: 13-22935-54640-30039-45949-41945-37494-18536-34124-63074-33671-08341-01230-63402-23598.

Segundo [31] o *software* MatLab surgiu no início da década de 80, desenvolvido por Cleve Moler, no Departamento de Ciência da Computação da Universidade do Novo México, EUA. Versões anteriores a este foram desenvolvidas na firma comercial MathWorks Inc., que detêm os direitos autorais das implementações. Uma versão do *software* pode ser obtida pela internet através da página *http://www.matlab.com.br*.

Através de estudos comparativos realizados por [37] observou-se que alguns problemas matemáticos implementados em MatLab foram resolvidos muito mais rapidamente e de maneira mais eficiente que em outras linguagens como *C*, Basic, Pascal ou Fortran.

Ainda com relação a programação é possível utilizar o *Simulink*®[4] - simulação e modelagem de projetos. O *Simulink*® [34] é um ambiente desenvolvido para a simulação de vários domínios e design baseado em modelos para sistemas dinâmicos e integrados. Ele fornece um ambiente gráfico interativo e um conjunto personalizável de bibliotecas de blocos que permitem projetar, simular, implementar e testar inúmeros sistemas variantes no tempo, incluindo comunicações, controles, processamento de sinais, processamento de vídeo e processamento de imagem.

1.4 Justificativas para o trabalho

O tema escolhido para pesquisa leva em consideração um estudo anterior sobre a inscrição de secções cônicas em polígonos irregulares de n lados, encontrado no decorrer do projeto 3D Gemas e desenvolvido por [5]. Uma das motivações para o desenvolvimento de uma heurística capaz de empacotar elipses tangentes em polígonos baseia-se no fato de que grande parte dos estudos na área de empacotamento são realizados levando em consideração formas regulares, ou seja, empacotar formas regulares em objetos com a mesma característica.

A pesquisa anteriormente realizada por [5] inscreve secções cônicas - no caso a elipse e a circunferência - que formam, respectivamente, a base dos modelos de lapidação oval e redondo. Com o desenvolvimento de uma metodologia analítica desse problema e uma compreensão dos métodos de buscas das soluções viáveis, busca-se otimizar várias

[4] *MathWorks*® Simulink - Simulation and Model-Based Design. Disponível em < http://www.mathworks.com/products/simulink/ >

elipses aumentando assim o aproveitamento volumétrico de uma gema após o processo de lapidação. A inquietação inicial surge com o propósito de empacotar elipses em torno de uma elipse inicial pré-estabelecida obtendo dessa forma múltiplos modelos de lapidação oval em uma mesma gema. Essa problematização justifica-se pela sua influência no contexto científico e industrial, bem como no desenvolvimento do projeto 3D Gemas.

Para a elaboração da heurística realizou-se um estudo dos principais métodos e heurísticas indicados para resolução de problemas de corte e empacotamento bidimensional, destacando alguns deles:

- **Método *Simplex*** adaptado ao problema de corte, descrito em [22];
- **Métodos de Otimização Contínua** através de sistemas de equações não-lineares e a obtenção da solução a partir do método de Newton-Raphson, desenvolvido por [21];
- **Algoritmos de Aproximação** aplicados ao problema de corte e empacotamento, são algoritmos eficientes e que geram soluções com garantia de qualidade adaptados por [49];
- **Inferência *Fuzzy*** utilizada por [43] em problemas de corte de estoque com sobras aproveitáveis *(PCESA)*;
- **Algoritmos genéticos** utilizados por [16] como uma técnica de otimização em problemas de corte industrial bidimensional;
- **Método de otimização irrestrita de Hooke e Jeeves** aplicado ao problema de inscrição de secções cônicas em polígonos de n lados [5].

Com base no estudo realizado sobre os principais métodos, optou-se em desenvolver uma heurística construtiva sendo utilizada uma adaptação do método de otimização de Hooke e Jeeves para a inscrição da elipse inicial. Segundo [14] heurísticas construtivas ou algoritmos construtivos são métodos baseados em construir uma solução a partir de regras pré-definidas, levando em consideração as características do problema. Alguns dos métodos acima citados serão abordados no Capítulo 2.

A metodologia adotada para o desenvolvimento da heurística poderá ser utilizada para diversos problemas de corte e empacotamento, não apenas na indústria lapidá-

ria mas também na indústria têxtil, mecânica e automotiva, principalmente na usinagem de peças seriadas.

Este trabalho decorre do interesse em desenvolver uma heurística para problemas de corte e empacotamento que modifique o estado da arte nessa área, no que diz respeito as restrições impostas pelo problema principalmente tratando-se de uma região com dimensões não fixas.

1.5 Objetivo geral do trabalho

Desenvolver uma metodologia analítica e computacional, com a qual fosse possível encontrar uma solução viável para o problema de empacotamento de elipses, idênticas ou não, sem sobreposição e tangentes a cada vértice e quadrante da elipse inicial. Nesta instância do problema do empacotamento, se pretendeu obter a maximização da área total das elipses empacotadas em P e a minimização do tempo de processamento computacional.

1.5.1 Objetivos específicos do trabalho

Para o desenvolvimento da heurística construtiva e validação da metodologia faz-se necessário elaborar alguns objetivos específicos para definir as metas a serem alcançadas. Podemos destacar como prioridade:

- transformar o método de otimização irrestrita de Hooke e Jeeves em um método de otimização com restrições;
- investigar os métodos mais utilizados nos problemas de corte e empacotamento, capazes de serem aplicados a diferentes tipos de objetos;
- modelar adequadamente as soluções viáveis no espaço de busca;
- possibilitar uma modelagem mais genérica e flexível do problema, de forma a variar o número de iterações conforme a necessidade do usuário do programa;
- definir e implementar os algoritmos e sub-rotinas necessários para a validação da metodologia proposta;
- realizar adaptações na função pré-definida *inpolygon* do MatLab [34] utilizada na verificação inscrição das elipses no polígono;

- realizar as simulações computacionais com a utilização do *software* MatLab;
- elaborar um pseudo-código para disponibilizar aos demais pesquisadores na área de problemas de corte e empacotamento;
- obter a função exponencial de cada simulação, caracterizando dessa forma o comportamento da heurística;
- analisar e validar os resultados obtidos com a heurística desenvolvida.

1.6 Organização da Dissertação

Neste capítulo, foram introduzidos os problemas de corte e empacotamento, suas classificações com um enfoque no setor industrial. Situou-se o leitor sobre a importância da Tecnologia 3D Gemas no contexto desse trabalho e destacou-se a utilização do *software* MatLab para a implementação da heurística desenvolvida. Foram apresentados, também, o objetivo geral e os objetivos específicos para esse trabalho bem como a justificativa para a realização do mesmo.

O Capítulo 2 apresenta, resumidamente, trabalhos relacionados aos problemas de corte e empacotamento e algumas técnicas utilizadas por outros autores, a fim de proporcionar uma visão mais ampla das pesquisas em problemas de corte e empacotamento e também relacionadas ao posicionamento de formas. Faz-se uma abordagem mais aprofundada do método de Hooke e Jeeves proposto por [4] e indicado como o mais viável para a otimização da elipse inicial.

No Capítulo 3, é apresentada a generalização da equação paramétrica de uma elipse e a formulação matemática para a otimização da elipse inicial e para o empacotamento das elipses tangentes. Descreve-se as estratégias de otimização e a condição adicional inserida para garantir a inscrição total das elipses e formaliza-se o conceito de convexidade.

Após, o Capítulo 4 apresenta uma adaptação do método de otimização de Hooke e Jeeves e as buscas exploratória e padrão realizadas para otimizar a elipse inicial, obtendo assim os parâmetros necessários para realizar o empacotamento das elipses tangentes.

A seguir, no Capítulo 5 descreve-se a metodologia analítica para o empacotamento das elipses tangentes aos vértices da elipse inicial. Neste Capítulo é formalizado o

cálculo para a obtenção de cada vértice de uma elipse, sendo utilizado em todas as elipses empacotadas.

Já o Capítulo 6 apresenta a metodologia analítica desenvolvida para o empacotamento de elipses tangentes aos quadrantes da elipse inicial baseada no princípio da construção de peças Lego e na obtenção de equações paramétricas resultantes da aplicação de simetrias em $\mathbb{R}^2$.

No Capítulo 7 é apresentada a estratégia computacional utilizada para verificar quais elipses possuem pontos exteriores ao contorno que define o polígono. As simulações da heurística desenvolvida e os parâmetros obtidos de cada elipse bem como o tempo de processamento e aproveitamento volumétrico são descritos no Capítulo 8.

Após realizadas as simulações com os contornos escolhidos, o Capítulo 9 expõe os resultados obtidos e algumas considerações sobre restrições que possam interferir na busca pelo empacotamento ótimo, tais como polígonos não-convexos com muitos pontos e polígonos com concavidades acentuadas onde o cálculo do centro do polígono deve ser realizado considerando um centro de massa e não o centróide.

Por fim o Capítulo 10 apresenta os principais aspectos abordados, seguindo com uma discussão das principais contribuições alcançadas com a heurística desenvolvida e, posteriormente, por sugestões de trabalhos futuros.

2 Revisão bibliográfica

Este capítulo apresenta, primeiramente, os aspectos mais relevantes com relação aos problemas de corte e empacotamento, métodos distintos utilizados em pesquisas e trabalhos na área. Após, são tratadas as abordagens para os diversos problemas de posicionamento de formas em domínios distintos. Por fim, apresenta o método de Hooke e Jeeves adotado na heurística para o posicionamento da elipse inicial.

Problemas na área de corte e empacotamento *(PCEs)* são cada vez mais estudados pela comunidade acadêmica. Esses estudos buscam desenvolver métodos e heurísticas que minimizem a quantidade de matéria-prima necessária para gerar produtos no setor industrial e de fabricação e melhorar a qualidade nos serviços de logística em indústrias de grande porte.

Os problemas de corte e empacotamento são, em geral, classificados como problemas de otimização combinatória. A utilização de estratégias de otimização tem como objetivo geral minimizar ou maximizar uma função f em um domínio onde esta seja definida.

Um dos acontecimentos mais importantes da história mundial ocorrido em meados do século XVIII no Reino Unido e que acabou se espalhando pelo mundo no século XIX foi determinante para o surgimento de estudos voltados para os problemas de corte e empacotamento. Segundo [44], desde os primórdios da revolução industrial as intervenções econômicas dos produtores já apresentavam complicações ligadas as decisões de logística dos produtos.

Nesse contexto surgiram os primeiros estudos na área de *PCEs*. O trabalho pioneiro de *PCEs* foi realizado pelo matemático e economista soviético Leonid Vitaliyevich Kantorovich publicado em russo no ano de 1939 e por consequência da Guerra Fria foi traduzido para o inglês [27] e publicado somente em 1960. O estudo realizado por [27] apresenta alguns modelos de programação linear voltados para o planejamento, organização

e logística de produtos e seus métodos de solução.

Os métodos para resolver e modelar problemas de corte e empacotamento mais relevantes foram publicados por Gilmore e Gomory na década de 60. Dentre eles destaca-se a formulação feita em 1961 por [22] do Problema de Corte de Estoque Unidimensional, que propõe uma variação baseada no método *Simplex* com geração de colunas para a relaxação linear de um problema de grande porte. Em 1963, os mesmos autores estenderam a aplicabilidade dos métodos propostos em [22], adaptando-os para um problema específico de estudo de caso, acrescentando restrições que mais tarde definiram-se como fundamentais no ponto de vista prático da indústria.

No que diz respeito aos *PCEs* surgem as heurísticas construtivas como alternativas computacionalmente viáveis para resolver os problemas de otimização combinatória. Segundo [26], define-se heurística como uma regra simples e eficiente para resolver um determinado problema ou tomar uma decisão em relação ao mesmo. A limitação de se utilizar uma heurística está relacionada a busca pela solução ótima, a solução sempre é encontrada mas não necessariamente, é ótima. Com base nos estudos e métodos desenvolvidos por [22] surgiram várias heurísticas e métodos adaptados para resolver os *PCEs*.

Entre as heurísticas e métodos desenvolvidos para resolver os problemas de corte e empacotamento, destaca-se o estudo realizado por [22](1961) no qual foi proposto a formulação do Problema de Corte de Estoques utilizando Programação Linear *(PL)* em especial técnicas de Programação Inteira *(PI)*. Nesses casos faz-se uma relaxação de algumas restrições do problema, levando em consideração a aproximação da solução ótima do programa linear podendo resultar em uma solução viável para a programação linear inteira.

Dois anos mais tarde, os mesmos autores utilizaram o método *simplex* com geração de colunas com relaxamento das condições de integralidade das variáveis de decisão do problema. Segundo [22], uma série de colunas são geradas de acordo com a quantidade de parâmetros disponíveis e todos os possíveis padrões P de corte. Com as colunas determinadas é possível encontrar uma solução para o problema relaxado e posteriormente para o problema inteiro.

A limitação do método *simplex* aplicado por [22] está diretamente ligada a quantidade de padrões de corte P que em problemas de grande porte pode ultrapassar

a quantidade de milhões de padrões. Como estratégia para essa limitação gera-se um subconjunto dos melhores padrões e aplica-se o método novamente.

Outra estratégia para solucionar os problemas de corte e empacotamento utiliza resultados obtidos por métodos de Otimização Contínua na busca de soluções mais precisas. Um dos trabalhos que utiliza essa estratégia foi realizado por [21] para minimizar as dimensões de um objeto que comporte em seu interior um dado número de itens circulares idênticos, sem que haja sobreposições. Para isso, o autor propõe uma estratégia baseada na formulação de sistemas de equações não-lineares obtendo sua solução com a aplicação do método de Newton-Raphson.

A formulação do problema conforme [21] é expressa pelo modelo não-linear:

Minimizar	as dimensões do objeto
sujeito a	comportar os itens sem sobreposições

As violações remanescentes podem ser facilmente corrigidas com o afastamento de itens que se sobreponham e a posterior ampliação das dimensões do objeto. A cada iteração aumenta a quantidade de itens empacotados, conforme a Figura 2.1:

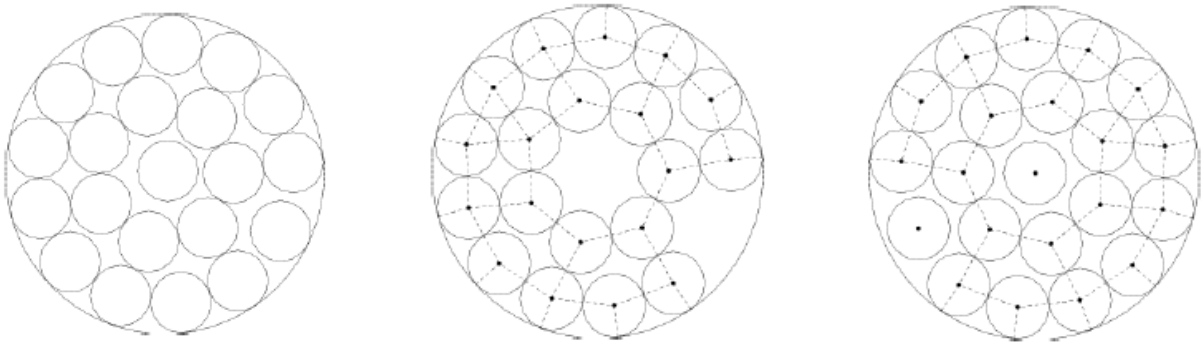


Figura 2.1: Empacotamento de círculos em um círculo maior. Fonte: [21].

No decorrer do desenvolvimento da heurística construtiva para o empacotamento de elipses tangentes em polígonos de n-lados além da revisão bibliográfica sobre os métodos adequados para resolver $PCEs$ também foram pesquisados métodos utilizados para o posicionamento de formas, mostrados na Seção 2.1.

2.1 Trabalhos relacionados ao posicionamento de formas

Com relação aos problemas de empacotamento de itens tangentes e não idênticos em polígonos irregulares de n lados observa-se a falta de referências específicas, de modo que encontram-se em maior ênfase na literatura o posicionamento de itens idênticos sem a utilização do método de otimização de Hooke e Jeeves e com relaxamento das restrições impostas pelo problema, descaracterizando parte da aplicação do problema.

Destaca-se no posicionamento de formas o estudo realizado por [32], que propôs um processo de otimização para o problema de rotação e translação de formas irregulares em recipientes de dimensões fixas baseado em heurísticas probabilísticas (polígono de obstrução e recozimento simulado) sem o uso de penalização externa.

O autor utilizou o conceito de polígono de obstrução - que mostra claramente a área obstruída por uma figura no posicionamento de outra - juntamente com o método do Recozimento Simulado *(Simulated Annealing)*, uma heurística de otimização de exploração local que analisa a cada iteração apenas uma solução do problema. A Figura 2.2 representa uma das soluções obtida por [32] para o problema do posicionamento de formas em recipientes de dimensões fixas:

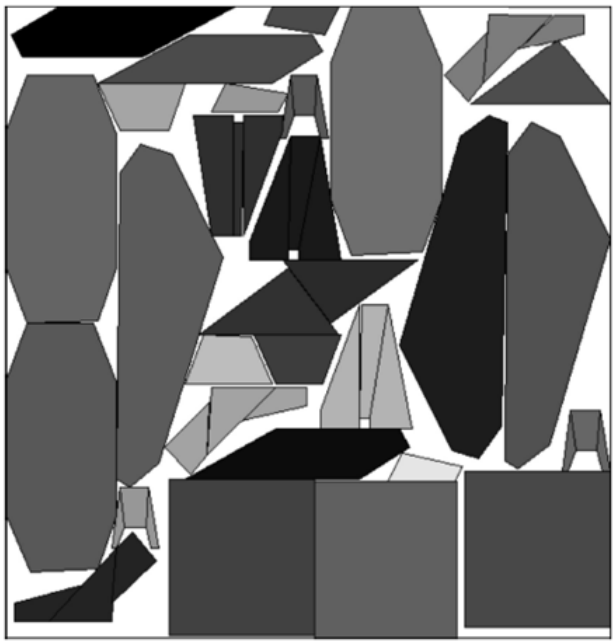


Figura 2.2: Posicionamento de formas em um recipiente de dimensões fixas. Fonte: [32]

O posicionamento de formas é muito utilizado no corte de diamantes e de outras gemas onde a precisão máxima deve ser obtida. Para isso, [48] uniu geometria e cálculo multivariável com técnicas de otimização, sendo capaz de criar algoritmos que automaticamente geram planos precisos de corte que maximizam brilho e rendimento (maximização do volume).

O posicionamento de múltiplas elipses, mostrado na Figura 2.3 é inicialmente abordado por [48] com o propósito de desenvolver outros modelos de lapidação além dos modelos redondo e oval.

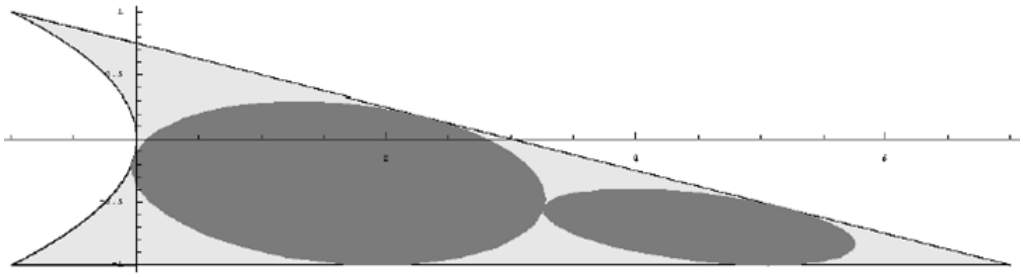


Figura 2.3: Posicionamento de duas elipses em um dado *container*. Fonte:[48]

Segundo [48], a partir da elipse pode-se obter os demais modelos de lapidação, para isso fez o uso de métodos numéricos praticáveis para problemas de otimização semi-infinita com restrições não-côncavas. Na Figura 2.4 a seguir, o autor utiliza o mesmo *container* para inscrever a base de dois modelos de lapidação coração:

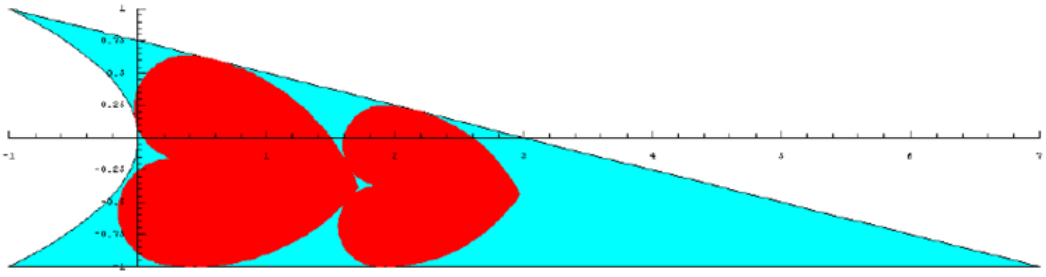


Figura 2.4: Base do modelo coração em um dado *container*. Fonte: [48]

Em contrapartida aos métodos numéricos, polígono de obstrução e ao recozimento simulado, [24] e [25] propõe a utilização de algoritmos evolutivos baseados na seleção de espécies para a evolução e iterações entre as espécies para determinar uma aptidão.

O autor realiza uma modificação no método evolutivo denominado *Fitness*

Sharing, onde todos os indivíduos tem a mesma aptidão objetiva, mas eles agem para reduzir a aptidão compartilhada através da competição de recursos. A modificação no *Fitness Sharing* resultou no método *Pure Co-evolutionary Shape Nesting (PCSN)* que utiliza proporções de espécies para representar a população, assim simulando uma população infinitamente grande. Com nenhuma descoberta de operadores, como a mutação ou recombinação, a evolução consiste em uma única seleção com todas as espécies presentes na população inicial.

Os primeiros testes de [24] aplicaram o *Fitness Sharing* em problemas de posicionamento unidimensionais e bidimensionais de formas, com o posicionamento de quadrados alinhados aos eixos. Já em [25] os experimentos foram realizados com polígonos de forma arbitrária animando as formas e o polígono onde serão posicionados, conforme pode ser visto na Figura 2.5:

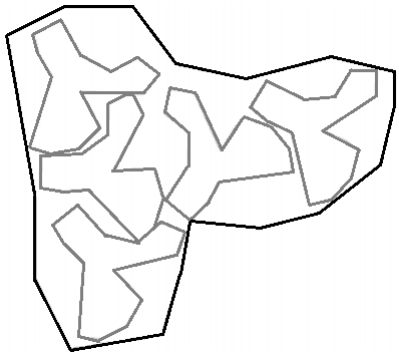


Figura 2.5: Posicionamento geral no polígono. Fonte: [25]

Em [25] também é possível observar a utilização do *PCSN* para posicionar os círculos no mesmo polígono utilizado para aninhar os polígonos menores. O autor posicionou círculos com 72 unidades de diâmetro, obtendo mais círculos posicionados com o *PCSN*, visto na Figura 2.6, do que utilizando três pacotes de *softwares* comerciais.

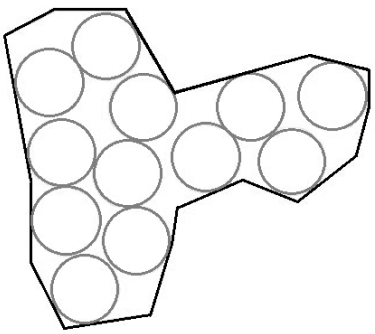


Figura 2.6: Posicionamento de círculos com PCSN. Fonte: [25]

Dessa forma, é possível concluir que a heurística adotada por [24] e [25] é mais eficiente computacionalmente e otimiza mais formas em um dado polígono que os *softwares* de otimização disponíveis no mercado.

2.2 Métodos para solucionar problemas de posicionamento de formas em polígonos

No que diz respeito ao posicionamento de formas em polígonos, tanto regulares quanto irregulares, geralmente são utilizados vários métodos e algoritmos para desenvolver uma heurística própria da solução. Um dos métodos utilizados para o assentamento de formas num dado polígono é o Recozimento Simulado *(Simulated Annealing)*. De acordo com [32], o recozimento simulado trata-se de um método de otimização baseado em busca por exploração local do gradiente ascendente de modo que é possível escapar de mínimos locais permitindo que seja explorado o espaço em direções que resultem num acréscimo da função objetivo.

Em determinados problemas de otimização classificados como difíceis com um número de restrições consideravelmente grande, aplica-se uma relaxação para produzir limites no algoritmo de busca. Dentre as relaxações mais utilizadas, destacam-se as de programação linear e as relaxações Lagrangeanas.

A otimização segundo [10], é uma área da Matemática Aplicada que trata de como calcular e computar valores ótimos que induzem um desempenho ótimo. Esses valores devem satisfazer as restrições que aparecem em determinados casos, impostas pelo problema inicial.

Outro método utilizado para resolver problemas de posicionamento de formas é o método de otimização de Hooke e Jeeves [4], que combinado com movimentos de translação e rotação, resulta no posicionamento ideal das formas que se deseja fixar. No decorrer do processo de otimização desenvolvido para inscrever a maior elipse possível em um polígono irregular de n lados são utilizadas várias etapas de tentativa e erro e um método iterativo de otimização.

O método de otimização de Hooke e Jeeves não faz parte dos métodos clássicos utilizados para minimizar ou maximizar uma função que não contenha restrições, o que

podemos chamar de otimização irrestrita. Segundo [4], "a otimização irrestrita trata do problema de minimizar ou maximizar uma função na ausência de qualquer restrição". Dessa forma, não existe a necessidade da solução estar contida no mesmo espaço das possibilidades de soluções existentes.

Conforme [23], quando se deseja encontrar o valor máximo ou mínimo de uma função complexa, às vezes os métodos falham ou não são viáveis pela extensão dos cálculos. A dificuldade em resolver os problemas com funções complexas originou o que hoje é conhecida como "busca direta". A vantagem de utilizar esse tipo de método é que basta definir uma estratégia de busca simples e o computador executará com tempo de execução inferior a métodos que não utilizam a idéia de busca em direções.

Com relação ao método de Hooke e Jeeves [15] afirmam que, os métodos diretos baseiam-se na comparação dos valores da função objetivo e são particularmente atrativos em situações onde as derivadas da função objetivo e das funções restrições não são viáveis, considerando o espaço de busca.

Geralmente, a solução de um problema de uma variável não auxilia na resolução de um problema correspondente com diversas variáveis. Esse tipo de método de otimização, como por exemplo o método de Hooke e Jeeves, utiliza o que se chama de "força bruta" e caracteriza-se pela falta de elegância matemática.

O desenvolvimento desse método por Robert Hooke e T. A. Jeeves decorreu de um problema envolvendo uma função complexa em que a utilização de métodos convencionais de otimização não era a melhor escolha. Eles realizaram experimentos com funções objetivo distintas e descobriram que os principais desligamentos ocorrem em curvas de contorno longas e finas. Dessa forma, elaboraram uma estratégia chamada "busca padrão" (*Pattern search*) para que fosse possível manipulá-las.

Uma das grandes vantagens da utilização do método de Hooke e Jeeves é que esse método pode ser facilmente modificado e ajustado ao problema que se deseja resolver. Essa modificação é observada em problemas que levam em consideração as restrições do contorno ou espaço de soluções possíveis. Transforma-se um método de otimização irrestrita em um método de otimização com restrições para garantir que a solução ótima esteja contida na região poligonal formada pelos vértices que descrevem o contorno do maior plano onde será inserida a elipse inicial.

No problema de inscrição da elipse inicial, tem-se que a função objetivo a ser

maximizada deve conter a restrição em relação ao ponto inicial que será assumido como parâmetro do método de busca de Hooke e Jeeves.

De acordo com [33], esse método utiliza um algoritmo de busca direta, ou seja, o espaço de soluções forma-se pela discretização dos parâmetros de ajuste, onde cada parâmetro corresponde a um determinado eixo desse espaço. Feito isso, é atribuído para cada parâmetro um valor mínimo e um valor máximo, bem como o número de intervalos. O algoritmo é baseado em sucessivas buscas exploratórias e lineares. As buscas lineares são definidas como buscas padrão.

O método proposto por Hooke e Jeeves é ilustrado por [4] na Figura 2.7:

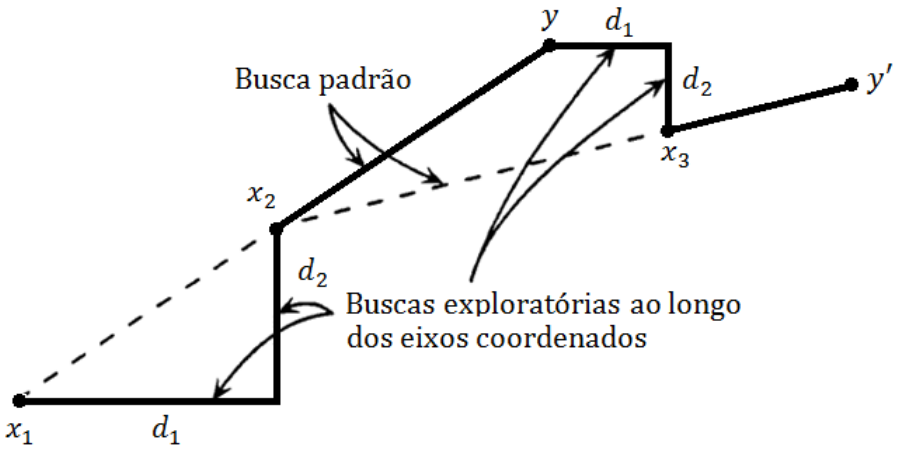


Figura 2.7: Método de otimização Hooke e Jeeves. Fonte: [4]

O método propõe a seguinte busca: conhecendo um ponto inicial x_1 e realizada uma busca exploratória nas direções da coordenada, irá dar origem ao ponto x_2. Já a busca padrão realizada na direção definida por $x_2 - x_1$ conduz ao ponto y e assim sucessivamente.

Esse método proposto pode ser baseado na utilização de buscas lineares ou na discretização de etapas. O algoritmo que descreve o método de otimização de Hooke e Jeeves utilizando buscas lineares é proposto por [4], no qual é necessário estabelecer um critério de parada δ e fixar um ponto inicial que servirá de base para as buscas lineares. Esse método realiza buscas pelas direções da coordenada:

Passo de inicialização:

Escolha um escalar $\delta > 0$ a ser usado na terminação do algoritmo. Escolha o ponto inicial x_1, defina $y_1 = x_1$, defina $k = j = 1$ e vá para o **Passo principal.**

Passo principal

1. Seja λ_1 uma solução ideal para o problema da minimização da $f(y_j + \lambda d_j)$ sujeita a $\lambda \in \mathbb{R}$, e seja $y_{j+1} = y_j + \lambda_j d_j$. Se $j < n$, substitua j por $j + 1$, e repita o **Passo 1**. Caso contrário, se $j = n$, defina $x_{k+1} = y_{j+1}$. Se $\|x_{k+1} - x_k\| < \delta$, pare; caso contrário, vá para o **Passo 2**.

2. Defina $d = x_{k+1} - x_k$, e seja $\widehat{\lambda}$ uma solução otimizada para o problema da minimização de $f(x_{k+1} + \lambda d)$ sujeita a $\lambda \in \mathbb{R}$. Defina $y_1 = x_{k+1} + \widehat{\lambda} d$, defina $j = 1$, substitua k por $k + 1$, e vá para o **Passo 1**.

O método de Hooke e Jeeves[4], que utiliza buscas lineares, converge mais rapidamente para a solução ótima e resulta em um esforço computacional menor (diminuição do tempo de processamento) do que se utilizarmos a discretização de etapas. No método com a utilização da discretização de etapas é definida a direção das coordenadas de busca, isso facilmente é percebido no passo de inicialização do algoritmo proposto por [4].

Passo de inicialização

Seja $d_1, \ldots, d_n$ a direção das coordenadas. Escolha um escalar $\delta > 0$ para ser usado na terminação do algoritmo. Além disso, escolha o tamanho do passo inicial, $\Delta \geq \delta$, e o fator de aceleração, $\alpha > 0$. Escolha o ponto inicial x_1, deixe $y_1 = x_1$, deixe $k = j = 1$, e vá para o **Passo Principal**.

Passo principal

1. Se $f(y_j + \Delta d_j) < f(y_j)$, o teste termina com sucesso; deixe $y_{j+1} = y_j + \Delta d_j$, e vá para o **Passo 2**. Se, de qualquer modo, $f(y_j + \Delta d_j) \geq f(y_j)$ o teste é considerado falho. No caso, $f(y_j - \Delta d_j) < f(y_j)$, deixe $y_{j+1} = y_j - \Delta d_j$, e vá para o **Passo 2**; se $f(y_j - \Delta d_j) \geq f(y_j)$, deixe $y_{j+1} = y_j$, e vá para o **Passo 2**.

2. Se $j < n$, substitua j por $j + 1$, e repita o **Passo 1**. Senão, vá para o **Passo 3** se $f(y_{n+1}) < f(x_k)$, e vá para o **Passo 4** se $f(y_{n+1}) \geq f(x_k)$.

3. Deixe $x_{k+1} = y_{n+1}$, e deixe $y_1 = x_{k+1} + \alpha(x_{k+1} - x_k)$. Substitua k por $k+1$, deixe $j = 1$, e vá para o **Passo 1**.

4. Se $\Delta \leq \delta$, pare; x_k é a solução prescrita. Caso contrário, substitua Δ por $\frac{\Delta}{2}$. Se $y_1 = x_k$, e $x_{k+1} = x_k$, substitua k por $k+1$, deixe $j = 1$, e repita o **Passo 1**.

Em relação à utilização do fator ou passo de aceleração, descrito no passo de inicialização, na segunda fase do método de Hooke e Jeeves utilizando discretização de etapas [47] afirma que:

> "[...] os componentes do vetor base que conduziram a uma otimização da função objetivo, são submetidos a um processo de *aceleração*. Assim, o passo de aceleração tem o propósito de acelerar a convergência da função objetivo para o ponto de mínimo (minimização) ou de máximo (maximização). Após o passo de aceleração, novos componentes para o vetor base são obtidos." (2007, p.81).

No passo principal, as etapas 1 e 2 descrevem uma busca exploratória ao longo das direções de cada coordenada que compõe o vetor base, já na etapa 3 temos o passo de aceleração na direção de $x_{k+1} - x_k$.

3 Formulação matemática do problema de empacotamento de elipses

Este capítulo trata da formulação matemática do problema de empacotamento de elipses tangentes em um polígono de n lados. Primeiramente, é realizada a generalização da equação paramétrica que será utilizada no procedimento para encontrar as elipses tangentes. Apresenta-se, então, as coordenadas locais da elipse inicial e a generalização da otimização das elipses tangentes. Finalizando, define-se as estratégias de otimização.

3.1 Generalização da equação paramétrica de uma elipse

Para determinar os pontos de tangência de cada elipse em relação a ε (elipse inicial) será utilizada a equação paramétrica da elipse com centro deslocado da origem $C(0,0)$ do sistema de coordenadas cartesiano e com um ângulo de rotação σ. Considerando um sistema de coordenadas fixo $S(0,i,j)$ cada ponto P tem coordenadas (x,y) conforme a Figura 3.1, quando aplicada uma rotação R obtêm-se um novo sistema de coordenadas $S'(0,i',j')$. Esse novo sistema S', rotacionado σ^c [41, 19], tem a mesma origem que o sistema S. Dessa forma, o movimento de rotação realizado não gera nenhum movimento de translação de pontos. O ângulo θ que é utilizado para determinar as coordenas do ponto $P(x,y)$ não deve ser confundido com o ângulo de rotação σ dos semi-eixos das elipses rotacionadas em relação ao sistema de coordenadas cartesianas originais.

Primeiramente, consideramos uma elipse com centro na origem do sistema de coordenadas cartesiano, sem que seja aplicada qualquer rotação R, conforme mostra a Figura 3.1 e as coordenadas do ponto $P(x,y)$ são definidas pela Equação 3.1:

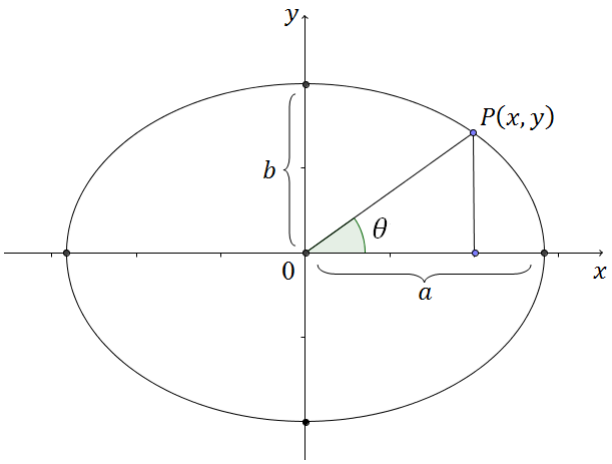


Figura 3.1: Elipse posicionada na origem. Fonte: Elaborado pela autora.

$$\begin{cases} x = a \cdot \cos\theta \\ y = b \cdot \sin\theta \end{cases} \tag{3.1}$$

Como o centro da elipse $C(x_c, y_c)$ está posicionado na origem as coordenadas serão nulas, ou seja, $x_c = y_c = 0$. Na Figura 3.2, a elipse foi transladada em relação a origem [8, 41] e as novas coordenadas do ponto $P(x, y)$ descritas pela Equação 3.2 são obtidas com base na Equação 3.1:

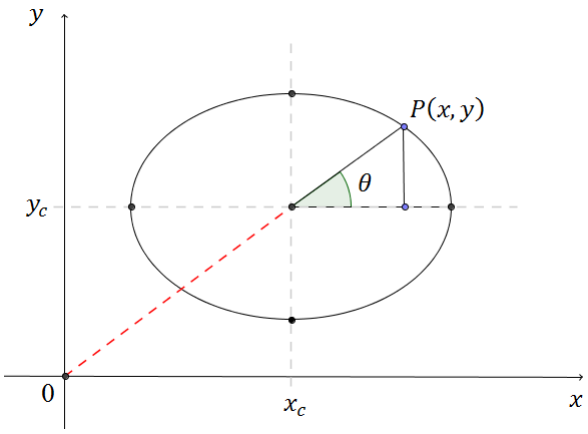


Figura 3.2: Elipse transladada em relação a origem. Fonte: Elaborado pela autora.

Segundo [19], as novas coordenadas são obtidas através da translação considerando um deslocamento paralelo ao eixo XX' e paralelo ao eixo YY', conforme a Equação 3.2:

$$\begin{cases} x = x_c + a \cdot \cos\theta \\ y = y_c + b \cdot \sin\theta \end{cases} \tag{3.2}$$

Considerando que as elipses empacotadas possuem um ângulo de giro σ^c em relação aos eixos cartesianos, utiliza-se o conceito de rotação de um ponto [8, 19, 29, 41] mostrado na Figura 3.3 e faz-se a generalização da equação paramétrica:

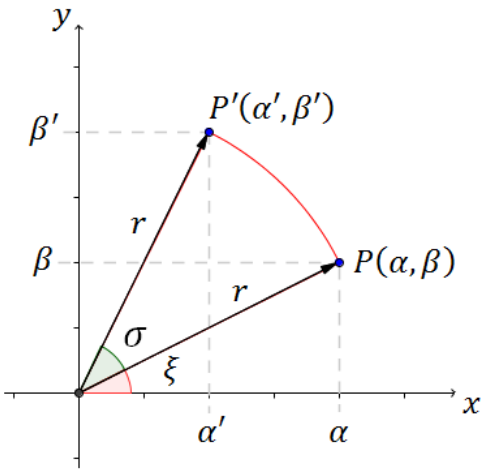


Figura 3.3: Rotação de um ponto no sistema cartesiano. Fonte: Elaborado pela autora.

Sendo r o comprimento do vetor da origem até o ponto P, $\alpha = r \cdot \cos(\xi)$ e $\beta = r \cdot \sin(\xi)$, a relação entre o ponto P e P' é dada pela Equação 3.3:

$$\begin{cases} \alpha' = r \cdot \cos(\xi + \sigma) \\ \beta' = r \cdot \sin(\xi + \sigma) \end{cases} \tag{3.3}$$

utilizando as fórmulas para seno e cosseno da soma de dois ângulos [19] , descritas nas Equações 3.4 e 3.5:

$$\sin(\xi + \sigma) = \sin(\xi) \cdot \cos(\sigma) + \sin(\sigma) \cdot \cos(\xi) \tag{3.4}$$

e

$$\cos(\xi + \sigma) = \cos(\xi) \cdot \cos(\sigma) - \sin(\xi) \cdot \sin(\sigma) \tag{3.5}$$

podemos reescrever a Equação 3.3:

$$\begin{cases} \alpha' = r \cdot \cos(\xi) \cdot \cos(\sigma) - r \cdot \sin(\xi) \cdot \sin(\sigma) \\ \beta' = r \cdot \sin(\xi) \cdot \cos(\sigma) + r \cdot \cos(\xi) \cdot \sin(\sigma) \end{cases} \tag{3.6}$$

ou seja,

$$\begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) \\ \sin(\sigma) & \cos(\sigma) \end{pmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{3.7}$$

e acrescentando a translação a partir da origem [19], temos a Equação 3.8:

$$\begin{bmatrix} \alpha' \\ \beta' \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) \\ \sin(\sigma) & \cos(\sigma) \end{pmatrix} \cdot \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{3.8}$$

Denominamos σ o ângulo da rotação realizada no sentido anti-horário (mesmo sentido que será utilizado para definir os vértices das elipses) e substituímos novamente α e β na Equação 3.8, resultando na equação paramétrica de uma elipse com centro deslocado da origem do sistema de coordenadas cartesiano, com uma rotação σ e semi-eixos a e b:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) \\ \sin(\sigma) & \cos(\sigma) \end{pmatrix} \cdot \begin{bmatrix} a \cdot \cos(\xi) \\ b \cdot \sin(\xi) \end{bmatrix} \tag{3.9}$$

onde $\sigma \in [0, 2\pi]$ e a matriz R de rotação, definida por [19]:

$$R = \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) \\ \sin(\sigma) & \cos(\sigma) \end{pmatrix} \tag{3.10}$$

Com a formulação da equação paramétrica de uma elipse rotacionada em relação ao sistema de coordenadas cartesianas originais, utilizamos ξ e $\sigma \in [0, 2\pi]$. Dessa forma, uma elipse deslocada da origem e com uma rotação R definida com $\sigma \in [0, 2\pi]$ mostrada na Figura 3.4 terá sua equação descrita por 3.11:

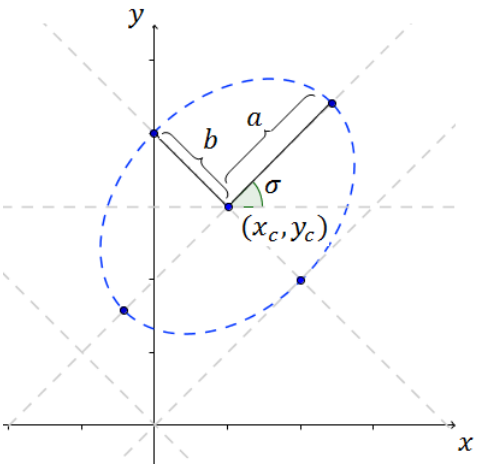


Figura 3.4: Elipse com centro transladado em relação a origem e com rotação R. Fonte: Elaborado pela autora.

$$\begin{cases} x = x_c + a\cos(\sigma)\cos(\xi) - b\sin(\sigma)\sin(\xi) \\ y = y_c + a\cos(\sigma)\sin(\xi) + b\sin(\sigma)\cos(\xi) \end{cases} \tag{3.11}$$

Para cada elipse tangente a ε será utilizada uma variação da Equação 3.11. Essa variação é resultado do produto da Equação 3.11 pela matriz que define a simetria em $\mathbb{R}^2$ utilizada.

3.2 Coordenadas locais de ε inscrita em P

Para um dado conjunto de pontos $\{(x_1, y_1), (x_2, y_2), \cdots, (x_{n_p}, y_{n_p})\}$, que determina os vértices de um polígono P, a elipse inicial ε, contida na área definida pelo polígono, é determinada pelos parâmetros calculados:

- as coordenadas do centro da elipse: $C(x_c, y_c)$;
- ângulo de giro do semi-eixo maior da elipse: σ (^{c});
- semi-eixo maior da elipse: a;
- semi-eixo menor da elipse: b;
- excentricidade pré-definida: e;
- vértices: $A1, B1, C1, D1$.

Quanto a relação entre os elementos básicos de uma elipse, sabemos que:

$$b = a\sqrt{1 - e^2} \tag{3.12}$$

Essa relação determina o semi-eixo menor b de todas as elipses tangentes a serem empacotadas em P.

Na Figura 3.5, os parâmetros de ε foram calculados e utilizados para o empacotamento das demais elipses. Os parâmetros das coordenadas do centro de ε são obtidos juntamente com a informação das coordenadas dos vértices que formam P através do

software 3DReshaper[5], mesmo assim calcula-se novamente as coordenadas do centro para verificar a equivalência dos resultados obtidos.

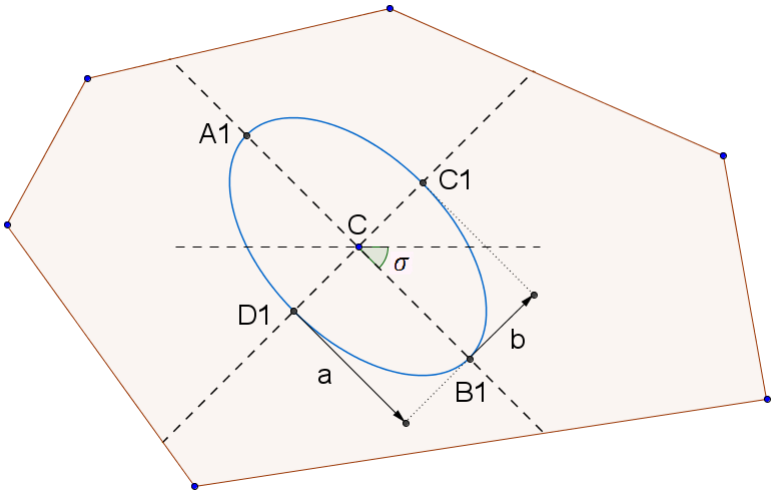


Figura 3.5: Elipse inicial interior a um polígono dado. Fonte: Elaborado pela autora.

A excentricidade de ε é pré-definida e fixa em 0,593946 sendo que para as elipses tangentes fixa-se o mesmo valor. Ao final da inscrição essa excentricidade é verificada novamente como padrão para o modelo oval de lapidação. Utiliza-se a notação a e b para os semi-eixos na primeira iteração, na inscrição final aw e bw apenas para diferenciar que houve uma mudança no valor inicial.

3.3 Generalização da otimização de elipses tangentes

Considerando ε com parâmetros $(x_c, y_c, \sigma, aw, bw)$, a heurística construtiva estabelece que para cada vértice $V_i(Vi_x, Vi_y)$ de ε tem-se uma elipse tangente inscrita no polígono P, descrita com base nos parâmetros:

- coordenadas do centro: $C(C_x, C_y)$
- ponto de tangência $P_i(Pi_x, Pi_y)$ coincidindo com um dos vértices $V_i(Vi_x, Vi_y)$
- semi-eixo maior aw;
- semi-eixo menor bw;

[5] *3DReshaper* é uma ferramenta utilizada para o processamento de nuvens de pontos 3D. Ele cobre as necessidades integrais em termos de pontos em nuvem, malha 3D e reconstrução de superfícies.

- ângulo de rotação σ.

A heurística desenvolvida define também que em cada quadrante Q_i de $\varepsilon(x_c, y_c, \sigma, aw, bw)$ será posicionada uma elipse tangente, inscrita no polígono P, com parâmetros:

- coordenadas do centro: $C(C_x, C_y)$
- ponto de tangência $P_i(Pi_x, Pi_y)$ definido pela intersecção da bissetriz do quadrante Q_i com curva;
- semi-eixo maior aw;
- semi-eixo menor bw;
- ângulo de rotação σ.

Como parte do processo de otimização e empacotamento, a heurística construtiva visa encontrar o conjunto de parâmetros $(x_c^*, y_c^*, \sigma^*, aw^*, bw^*)$ de ε que seja a base para inscrever as elipses $aw\,(C_x, C_y, \sigma)$ sendo que ε esta sujeito a:

$$S = \left\{ (x_c, y_c, \sigma, aw) : \left(x_c + aw\cos(\sigma)\,, y_c + aw\sqrt{1-e^2}\sin(\sigma) \right) \subset P, \sigma \in [0, 2\pi] \right\}, \tag{3.13}$$

que essencialmente traduz a necessidade de ε e das elipses tangentes serem interiores ao polígono P.

Dessa forma, temos estabelecido um *problema de otimização com restrições*. Métodos para solução computacional desse problema existem na literatura [4]. Em particular, podemos empregar métodos que não necessitem da informação do vetor gradiente $\nabla aw(x_c, y_c, \sigma)$, uma vez que não podemos obtê-lo analiticamente, e também queremos evitar o caminho alternativo de determinar esses gradientes computacionalmente.

Existem vários métodos adequados aos problemas dessa natureza, sendo que os mais utilizados são:

- Pesquisa operacional → tomada de decisão [43];
- Algoritmos genéticos [16];
- Algoritmos de aproximação para problemas de empacotamento [49].

3.4 Estratégias de otimização

3.4.1 Otimização da função objetivo

A área total do polígono é resultante da soma das áreas de cada triângulo, expressa pela Equação 3.14:

$$A_P = \sum_{i=1}^{nt} A_i \tag{3.14}$$

onde A_i é a área de cada triângulo i e nt o número de triângulos gerados.

No Capítulo 5, demonstra-se a Fórmula de Heron para o cálculo da área de cada triângulo. Sendo utilizada a área total do polígono para o cálculo do percentual de área restante, de modo que a área total das elipses inscritas tem que ser necessariamente menor ou igual a área total do polígono.

Satisfazendo essa condição, podemos definir o problema de otimização com as restrições estabelecidas em quatro etapas e uma condição adicional:

1 - Restrições para otimização de ε;

2 - Restrições para otimização das elipses tangentes aos vértices V_i de ε;

3 - Restrições para otimização das elipses tangentes aos quadrantes Q_i de ε;

4 - Restrição de tangência;

5 - Condição adicional para o teste de convexidade em relação ao semi-eixo maior das elipses.

Na heurística desenvolvida considera-se apenas o empacotamento de 9 elipses, sendo elas:

- 1 Elipse: ε;

- 4 Elipses: 1 elipse tangente em cada quadrante Q_i;

- 4 Elipses: 1 elipse tangente em cada vértice V_i.

3.4.2 Etapa 1 - Restrições para otimização de ε

Essa etapa realiza a otimização de ε através da restrição S e posteriormente, utilizando uma otimização em função da área poligonal de P e áreas das elipses.

Maximizar $a(x_c, y_c, \sigma)$

sujeito a:

$$S = \left\{ (x_c, y_c, \sigma, a) : \left(x_c + a\cos(\sigma), y_c + a\sqrt{1-e^2}\sin(\sigma) \right) \subset P, \sigma \in [0, 2\pi] \right\}$$

A otimização em função da área poligonal determina a função f a ser maximizada de modo que a área total ocupada pelas elipses seja menor ou igual a área do polígono P. Para isso, leva-se em consideração a área de cada elipse determinada e a área total com a restrição da área poligonal mostrada na Equação 3.18:

$$f(aw, bw) = \sum_{i=1}^{n} aw \cdot bw \cdot \pi \tag{3.15}$$

como $bw = aw \cdot \sqrt{1-e^2}$ e $n = 9$, temos:

$$f(aw) = \sum_{i=1}^{9} aw_i \left(aw_i \cdot \sqrt{1-e^2} \right) \cdot \pi \tag{3.16}$$

$$f(aw) = \sum_{i=1}^{9} aw_i^2 \cdot \sqrt{1-e^2} \cdot \pi \tag{3.17}$$

$$A_T = f(aw) = \sum_{i=1}^{9} aw_i^2 \cdot \sqrt{1-e^2} \cdot \pi \leq A_P \tag{3.18}$$

3.4.3 Etapa 2 - Restrições para otimização das elipses tangentes aos vértices V_i de ε

$$\boxed{\text{E1} \Rightarrow aw1 = 0,5 \cdot \lambda_1 \in \mathbb{R}^+ \Rightarrow B1(B1x, B1y)}$$

Maximizar $aw1(C1_x, C1_y, \sigma)$

sujeito a:

$$S_1 = \left\{(C1_x, C1_y, \sigma, aw1) : \left(C1_x + aw1\cos(\sigma), C1_y + aw1\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$

$$\sigma \in [0, 2\pi]$$

$$(B1x, B1y) \in \varepsilon$$

$$t = 0$$

$$\boxed{\text{E2} \Rightarrow aw2 = 0,6 \cdot \lambda_2 \in \mathbb{R}^+ \Rightarrow C1(C1x, C1y)}$$

Maximizar $aw2(C2_x, C2_y, \sigma)$

sujeito a:

$$S_2 = \left\{(C2_x, C2_y, \sigma, aw2) : \left(C2_x + aw2\cos(\sigma), C2_y + aw2\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$

$$\sigma \in [0, 2\pi]$$

$$(C1x, C1y) \in \varepsilon$$

$$t = \frac{\pi}{2}$$

$$\boxed{\text{E3} \Rightarrow aw3 = 0,5 \cdot \lambda_3 \in \mathbb{R}^+ \Rightarrow A1(A1x, A1y)}$$

Maximizar $aw3(C3_x, C3_y, \sigma)$

sujeito a:

$$S_3 = \left\{(C3_x, C3_y, \sigma, aw3) : \left(C3_x + aw3\cos(\sigma), C3_y + aw3\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$

$$\sigma \in [0, 2\pi]$$

$$(A1x, A1y) \in \varepsilon$$

$$t = \pi$$

$$\boxed{\text{E4} \Rightarrow aw4 = 0,6 \cdot \lambda_4 \in \mathbb{R}^+ \Rightarrow D1(D1x, D1y)}$$

Maximizar $aw4(C4_x, C4_y, \sigma)$

sujeito a:

$$S_4 = \left\{(C4_x, C4_y, \sigma, aw4) : \left(C4_x + aw4\cos(\sigma), C4_y + aw4\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$
$$\sigma \in [0, 2\pi]$$
$$(D1x, D1y) \in \varepsilon$$
$$t = \frac{3\pi}{2}$$

3.4.4 Etapa 3 - Restrições para otimização das elipses tangentes a cada quadrante Q_i de ε

Nessa etapa, o ponto de tangência é determinado pela intersecção da bissetriz do quadrante Q_i com a curva que descreve ε. Levando em consideração que a bissetriz do quadrante ocorre em $t = \frac{\pi}{4}$, assumimos t com o mesmo valor para todos os quadrantes e variando apenas a equação paramétrica geral.

$$\boxed{\text{E5} \Rightarrow aw5 = 0,8 \cdot \lambda_5 \in \mathbb{R}^+ \Rightarrow P5(P5_x, P5_y) \in 1^{\text{o}} \text{ quadrante de } \varepsilon}$$

Maximizar $aw5(C5_x, C5_y, \sigma)$

sujeito a:

$$S_5 = \left\{(C5_x, C5_y, \sigma, aw5) : \left(C5_x + aw5\cos(\sigma), C5_y + aw5\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$
$$\sigma \in [0, 2\pi]$$
$$(P5_x, P5_y) \in \varepsilon$$
$$t = \frac{\pi}{4}$$

$$\boxed{\text{E6} \Rightarrow aw6 = 0,8 \cdot \lambda_6 \in \mathbb{R}^+ \Rightarrow P6(P6_x, P6_y) \in 2^{\text{o}} \text{ quadrante de } \varepsilon}$$

Maximizar $aw6(C6_x, C6_y, \sigma)$

sujeito a:

$$S_6 = \left\{(C6_x, C6_y, \sigma, aw6) : \left(C6_x + aw6\cos(\sigma), C6_y + aw6\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$
$$\sigma \in [0, 2\pi]$$
$$(P6_x, P6_y) \in \varepsilon$$
$$t = \frac{\pi}{4}$$

$$\boxed{\text{E7} \Rightarrow aw7 = 0,8 \cdot \lambda_7 \in \mathbb{R}^+ \Rightarrow P7(P7_x, P7_y) \in 3^{\text{o}} \text{ quadrante de } \varepsilon}$$

Maximizar $aw7(C7_x, C7_y, \sigma)$

sujeito a:

$$S_7 = \left\{(C7_x, C7_y, \sigma, aw7) : \left(C7_x + aw7\cos(\sigma), C7_y + aw7\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$
$$\sigma \in [0, 2\pi]$$
$$(P7_x, P7_y) \in \varepsilon$$
$$t = \frac{\pi}{4}$$

$$\boxed{\text{E8} \Rightarrow aw8 = 0,8 \cdot \lambda_8 \in \mathbb{R}^+ \Rightarrow P8(P8_x, P8_y) \in 4^{\text{o}} \text{ quadrante de } \varepsilon}$$

Maximizar $aw8(C8_x, C8_y, \sigma)$

sujeito a:

$$S_8 = \left\{(C8_x, C8_y, \sigma, aw8) : \left(C8_x + aw8\cos(\sigma), C8_y + aw8\sqrt{1-e^2}\sin(\sigma)\right) \subset P\right\}$$
$$\sigma \in [0, 2\pi]$$
$$(P8_x, P8_y) \in \varepsilon$$
$$t = \frac{\pi}{4}$$

3.4.5 Etapa 4 - Restrição de tangência

A restrição de tangência é estabelecida para que as elipses sejam tangentes entre si, de modo que não são permitidas elipses secantes. Essa restrição assegura também que não ocorra a sobreposição dos itens empacotados em P, o que acarretaria em modelos de lapidação defeituosos.

Para o empacotamento das elipses tangentes aos quadrantes Q_i não é necessária a restrição, visto que o ponto de tangência é determinado pela intersecção da curva com a bissetriz do quadrante. No empacotamento das elipses tangentes aos vértices V_i de ε, o ponto de tangência deve coincidir obrigatoriamente com um vértice de ε, essa restrição é dada por:

$$V_\varepsilon = Ve_i \qquad i = 1, \ldots, 4. \tag{3.19}$$

onde Ve indica um vértice qualquer da elipse a ser empacotada e V_ε um vértice de ε. Em cada vértice de ε é empacotada uma única elipse, temos então o número de elipses variando $i = 1, \ldots, 4$.

3.4.6 Etapa 5 - Condição adicional para o teste de convexidade em relação ao semi-eixo maior das elipses.

A condição adicional para o teste de convexidade é utilizada somente nas elipses que possuem pontos exteriores ao polígono P. Essa condição realiza uma redução gradativa no tamanho do semi-eixo maior aw da elipse que estiver parcialmente fora do contorno do polígono, de modo que, ao reduzir o tamanho do semi-eixo maior, reduzimos também o tamanho do semi-eixo menor que é expresso em função de aw, considerando uma excentricidade e fixa.

Além de reduzir o tamanho do semi-eixo, essa condição realiza um teste de convexidade em relação a direção de busca assumida para o posicionamento do semi-eixo maior considerando as arestas do polígono P. O passo de redução é definido pela Equação 3.20:

$$aw_i' = aw_i - \alpha \cdot aw_i \tag{3.20}$$

onde aw_i é o semi-eixo maior da elipse i que contém pontos externos não pertencentes ao polígono P e α é o coeficiente de redução estimado em 0.1, pelo teste de convexidade $\alpha \in [0, 1]$.

Com base na Equação 3.20, temos que:

$$aw_i' = aw_i(1-\alpha) \tag{3.21}$$

Levando em consideração os pontos que formam o polígono não-necessariamente convexo é possível formular o conceito de convexidade expresso por [2] através da Definição 3.4.1:

Definição 3.4.1. Um conjunto $C \subset \mathbb{R}^n$ é um conjunto convexo se o segmento de linha que junta quaisquer dois pontos de C estiver também em C.

Equivalentemente, o conjunto $C \subset \mathbb{R}^n$ é convexo se

$$\left.\begin{array}{l} x_1, x_2 \in C \\ \alpha \in \mathbb{R}, 0 \leqslant \alpha \leqslant 1 \end{array}\right\} \Rightarrow (1-\alpha)\, x_1 + \alpha x_2 \in C \tag{3.22}$$

Considere C um conjunto contendo os pontos x_1 e x_2 e $\alpha \in [0,1]$. Diz-se que o ponto $b = \alpha x_1 + (1-\alpha)x_2$ é o resultado de uma combinação convexa de x_1 e x_2. Além disso, calculando $\alpha + (1-\alpha)$ terá como resultado 1. Pode-se estender esta definição para um conjunto de pontos:

Tem-se que o ponto

$$\begin{cases} b = \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n \\ \alpha_i \geq 0 \quad i = 1, 2, ..., n \end{cases} \tag{3.23}$$

é uma combinação convexa dos pontos $x_1, x_2, ..., x_n$. Então,

$$\alpha_1 + \alpha_2 + ... + \alpha_n = \sum_{i=1}^{n} \alpha_i = 1 \tag{3.24}$$

A Definição 3.4.1 é equivalente ao Teorema 3.4.1 proposto por [2]:

Teorema 3.4.1. *Sejam os pontos u_1 e u_2, pertencentes ao $\mathbb{R}^n$, e u_3 um ponto qualquer entre u_1 e u_2. Para algum $0 \leq \alpha \leq 1$ tem-se:*

$$\alpha(u_1 - u_2) = u_3 - u_2 \tag{3.25}$$

$$u_3 = \alpha u_1 + (1-\alpha)u_2 \tag{3.26}$$

como cada coeficiente da Equação 3.26 é maior ou igual a 0, a soma deles é igual a 1, então u_3 é uma combinação convexa de u_1 e u_2.

Um subconjunto $C \in \mathbb{R}^n$ é chamado *convexo*, se e somente se, para quaisquer pontos u_1 e u_2 pertencentes a C qualquer combinação convexa

$$b = \alpha_1 u_1 + \alpha_2 u_2 \tag{3.27}$$

também pertencer a C. Em outras palavras, se C é convexo, então

$$\begin{cases} u_1 \in C \\ u_2 \in C \end{cases} \Rightarrow \alpha u_1 + (1-\alpha) u_2 \in C \qquad 0 \leq \alpha \leq 1 \tag{3.28}$$

Portanto, um conjunto C é considerado convexo se todos os pontos do segmento de uma reta que une dois pontos de C também pertencem a C [2].

O valor de α utilizado na heurística construtiva desenvolvida é definido com base no cálculo de incerteza considerando a propagação do erro nos semi-eixos maior e menor. Segundo [18], a maneira de obter-se o resultado dessa propagação é através da aplicação do método de Kleine e McClintock.

Nesse método, o resultado do cálculo do erro é uma função das variáveis independentes aw e bw. Através de experimentos onde fez-se a variação do comprimento dos semi-eixos e uma análise da área obtida, estimou-se $\alpha = 0,1$. O método de Kleine e McClintock [50] aplicado na heurística construtiva é descrito por:

$$\Delta Z = \sqrt{\left(\frac{\partial A}{\partial aw} \cdot \Delta aw\right)^2 + \left(\frac{\partial A}{\partial bw} \cdot \Delta bw\right)^2} \tag{3.29}$$

onde ΔZ representa o erro no resultado de área e Δaw e Δbw o erro das variáveis aw e bw. Assumindo $\Delta aw = \Delta bw = 0,1$.

Como:

$$A = aw \cdot bw \cdot \pi \tag{3.30}$$

temos as derivadas parciais:

$$\frac{\partial A}{\partial aw} = bw \cdot \pi \quad e \quad \frac{\partial A}{\partial bw} = aw \cdot \pi \tag{3.31}$$

Na Figura 3.6 é possível observar redução no comprimento do semi-eixo maior em função do valor escolhido para α. Quanto maior o valor de α maior será a redução do semi-eixo.

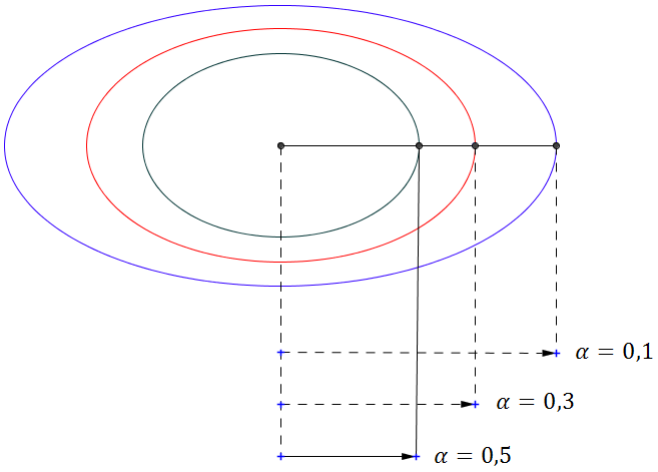


Figura 3.6: Representação da redução gradativa do semi-eixo maior em função de α. Fonte: Elaborado pela autora.

Com $\alpha = 0,1$, temos uma redução de 10% no comprimento do semi-eixo maior e na variação de área calculada através do método de Kleine e McClintock [50]. Na determinação do parâmetro α observamos que se realizada uma redução muito grande a cada passo, talvez não seja considerado que existe uma elipse maior interior ao polígono e localizada junto a aresta.

4 Adaptação do método de otimização de Hooke e Jeeves

Este capítulo apresenta a adaptação realizada no método de otimização de Hooke e Jeeves. Descreve primeiramente, os aspectos iniciais da Tecnologia 3D Gemas no que diz respeito a inscrição da elipse inicial. Posteriormente, são descritas as buscas exploratória e padrão utilizando o método. Por fim, são apresentados os parâmetros obtidos com o posicionamento da elipse inicial.

Com a Tecnologia 3D Gemas estão sendo desenvolvidos recursos computacionais que permitem obter resultados mais satisfatórios no decorrer do processo de lapidação virtual, visando um maior aproveitamento volumétrico no processo de lapidação de gemas [7]. O processo de lapidação virtual de gemas envolve 4 etapas, descritas a seguir: digitalização da gema com o scanner 3D a laser, geração do modelo virtual (tridimensional) da gema, processamento e geração de projetos virtuais de lapidação disponível no Apêndice A e, por fim, análise e escolha dos projetos virtuais de maior aproveitamento volumétrico da gema digitalizada [9].

Com essas etapas definidas é possível desenvolver um protótipo que encontra o plano com o contorno de maior área da gema martelada, servindo de base para o cálculo do tamanho do modelo que pode ser inserido nessa gema, mostrado na Figura 4.1. Dessa forma, se torna possível encontrar o posicionamento e o tamanho final do modelo de lapidação em uma gema, porém, ainda alguns ajustes são necessários devido à dificuldade de trabalhar com poliedros que não são convexos, o que acontece com grande parte das gemas.

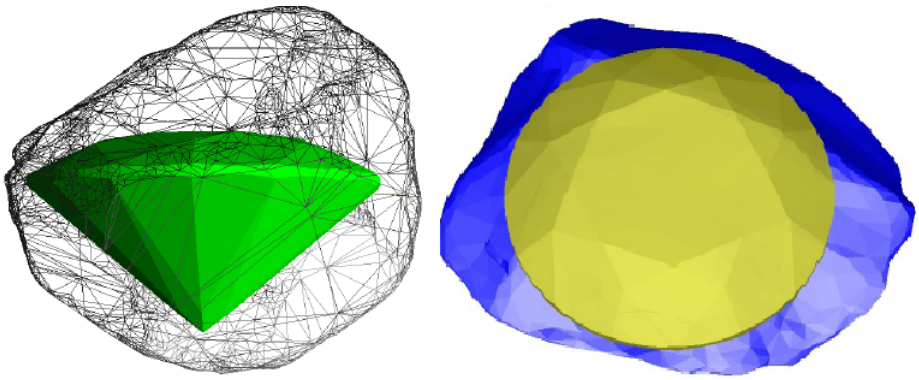


Figura 4.1: Protótipos da inscrição de um modelo de lapidação oval em um gema. Fonte: 3D Gemas.

A partir da utilização da ferramenta desenvolvida no Projeto 3D-Gemas, os lapidários da região poderão minimizar as perdas do processo de lapidação. As gemas em estado bruto estão desvalorizadas, por isso a necessidade das empresas buscarem alternativas no beneficiamento das gemas, no entanto elas não possuem artefatos inovadores nem lapidários qualificados para comercialização no mercado interno nem para exportar com valor agregado. Segundo [3], o valor agregado pelo beneficiamento dessas matérias-primas oscila entre 50 e 100 vezes o seu valor em estado bruto, por isso a necessidade do desenvolvimento de tecnologias que melhorem a qualidade final do produto.

Para o desenvolvimento do *Software* Otimizador 3D Gemas, considera-se que as gemas são compostas por seus contornos paralelos entre si e por planos perpendiculares aos eixos referenciais da gema [9], retirando desta forma a complexidade de trabalhar simultaneamente com três eixos.

Inicialmente, entendeu-se que encontrando a maior área da superfície gerada pelo contorno e sabendo a quantidade de contornos existentes acima e abaixo deste, pode-se calcular as dimensões do modelo que poderá ser inscrito na gema naquela posição.

Porém, para calcular ε inscrita em um contorno da gema deve-se saber qual é o centro de geométrico deste contorno, para isto, divide-se a superfície em triângulos. Após calcula-se a área superficial de cada triângulo, pode-se também encontrar qual o centro geométrico de cada triângulo, e por conseqüência o centro da superfície gerada pelo contorno. O cálculo do centro geométrico e da área do polígono são descritos detalhadamente no Capítulo 5.

Além disso, considera-se que a maior elipse possível inscrita no contorno deve

ter o semi-eixo maior igual à menor distância entre o centro geométrico e os pontos do contorno. Sabendo o comprimento do semi-eixo da elipse inserida no contorno, pode-se encontrar o tamanho que terá o modelo de lapidação, pois todas as dimensões do modelo são baseadas no eixo maior da elipse. Pelas deformidades que caracterizam uma gema, o polígono que descreve o maior plano, obtido através da seleção dos pontos críticos (inclusões, rachaduras e picos da gema), é geralmente não-convexo.

Levando em consideração a aplicabilidade da metodologia elaborada no ramo industrial é necessário estabelecer algumas restrições como: não é permitido ocorrer a sobreposição de formas nem formas parcialmente fora dos limites do polígono de n lados, isso acarretaria em peças defeituosas e sem utilidade na indústria.

A proposta de realizar o melhor posicionamento de várias elipses tangentes a uma elipse obtida utilizando métodos de busca exploratória e padrão, como exposto anteriormente, parte de um processo inicial que inscreve uma elipse maximizada em um polígono irregular de n lados e descrito em [5, 6]. Visando a melhor compreensão do problema de pesquisa escolhido, será realizada a revisão do método utilizado para resolver matematicamente o problema de encontrar e inscrever a maior elipse, com excentricidade pré-definida, dado um polígono não-necessariamente convexo, descrito em [5].

Cabe ressaltar que em testes a Tecnologia 3D Gemas atualmente faz apenas a otimização de uma elipse maximizada. Através do estudo desenvolvido, pode-se maximizar nove elipses aumentando o aproveitamento volumétrico da gema e obtendo mais modelos de lapidação da mesma gema.

Inicialmente, obtêm-se um vetor contendo as coordenadas dos vértices que descrevem o contorno do maior plano da gema. O vetor de pontos é obtido pela redução de uma malha triangular de pontos descrita em [7] e representado por 4.1:

$$vx = \begin{bmatrix} xv_1 & xv_2 & \cdots & xv_{np} \\ yv_1 & yv_2 & \cdots & yv_{np} \end{bmatrix} \tag{4.1}$$

O processo de inscrição de uma elipse utiliza uma transformação $3D \rightarrow 2D$ realizada pelo *software* MeshLab [35], gerando dessa forma o plano que descreve o maior contorno da gema, representado pela Figura 4.2:

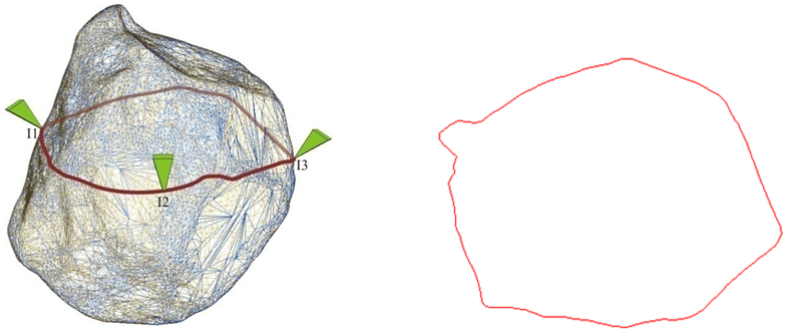


Figura 4.2: Contorno com seleção de pontos que descreve o maior plano da gema. Fonte: [35]

A transformação $3D$ é obtida a partir da generalização das transformações $2D$ incluindo considerações sobre o eixo $ZZ^{'}$ e realizada com base nos pontos de referência da gema. Juntamente com as coordenadas dos vértices são obtidas as coordenadas (x_c, y_c) do centro do polígono que será utilizado nos procedimentos de rotação, translação e buscas exploratórias. A inscrição da elipse leva em consideração uma excentricidade fixa em 0,593946. O vetor de pontos que descreve o maior plano pode ser escrito de acordo com 4.2:

$$\overrightarrow{V_i} = xv_i\,\overrightarrow{i} + yv_i\,\overrightarrow{j}, \quad i = 1, 2, \ldots, np \tag{4.2}$$

e o centro $C(x_c, y_c)$ do polígono, para tornar o processo mais simples denotaremos por $C(h, k)$.

Em seguida, são determinados os eixos da elipse em função da excentricidade. Sejam, $h = xp$ e $k = yp$ as coordenadas do centro de geométrico do polígono gerado pela transformação $3D \rightarrow 2D$ a partir do maior plano obtido na superfície da gema, calcula-se a distância do centro a cada vértice que forma o polígono para que seja posicionado o semi-eixo maior de ε.

A distância d entre dois pontos, tais como $A(x_1, y_1)$ e $B(x_2, y_2)$, de acordo com [30, 46] pode ser definida pela Equação 4.3:

$$\begin{aligned} d &= |\overrightarrow{AB}| \\ d &= |B - A| \end{aligned} \tag{4.3}$$

Generalizando, temos que a distância (d_1, d_2) é calculada pela Equação 4.4

$$\begin{aligned} d_1 &= |xv_i - h| \\ d_2 &= |yv_i - k|, \quad i = 1, 2, \ldots, n \end{aligned} \tag{4.4}$$

onde (xv_i, yv_i) são as coordenadas do vértice, conforme a Figura 4.3

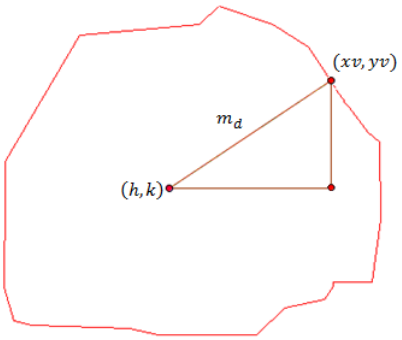


Figura 4.3: Distância do centro ao vértice do polígono. Fonte: Elaborado pelo autora.

A menor distância que pode ser assumida, comparando o centro e cada ponto que forma vx pode ser calculada pela norma dada pela Equação 4.5:

$$d \cdot d = d_1 \cdot d_1 + d_2 \cdot d_2 \tag{4.5}$$

ou seja, a menor distância é obtida com a Equação 4.6:

$$\begin{aligned} m_d &= \sqrt{d^2} \\ m_d &= \sqrt{d_1^2 + d_2^2} \end{aligned} \tag{4.6}$$

Conhecendo a menor distância, é realizada a determinação do semi-eixo maior a da elipse dado pela Equação 4.7 e utilizando apenas 90% da menor distância para garantir que esteja totalmente inscrito no polígono:

$$a = 0.9 \cdot m_d \tag{4.7}$$

Com o semi-eixo maior conhecido e a excentricidade pré-definida, determina-se o semi-eixo menor b. O semi-eixo menor é dado em função do semi-eixo maior e da excentricidade, expresso pela Equação 4.8:

$$b = a \cdot \sqrt{1 - e^2} \tag{4.8}$$

São assumidos somente valores positivos para b , pois este representa a medida do semi-eixo menor da elipse. Com os semi-eixos e a excentricidade definidos, parte-se para o posicionamento de ε utilizando o método de busca linear de Hooke e Jeeves [4].

O método de otimização irrestrita foi adaptado para uma região limitada definida pelo polígono de n-lados onde serão posicionadas as múltiplas elipses tangentes entre si. No método original de Hooke e Jeeves [4], o ponto de partida x_0 pode ser escolhido aleatoriamente e assume-se $x_0 = y_0$, já pelo método modificado tem-se que o ponto inicial de busca deve estar contido na região descrita pelo polígono P ou coincida com um dos vértices do polígono. Para satisfazer essa condição, o ponto inicial é calculado a partir da média aritmética das coordenadas dos vértices do polígono, descritas por 4.9:

$$xk_0 = \frac{\sum_{i=1}^{np} xv_i}{np} \quad e \quad yk_0 = \frac{\sum_{i=1}^{np} yv_i}{np} \tag{4.9}$$

onde np é o número de pontos que descrevem o polígono P.

Dessa forma, tem-se as coordenadas iniciais de busca xk_0 e yk_0. Define-se, também, para o algoritmo um critério de parada $\delta > 0$. Na inscrição da elipse o critério de parada assumido foi $\delta = 2,2 \cdot 10^{-16}$. Para o processo de otimização utilizaremos o método de Hooke e Jeeves que consiste em duas etapas de busca: exploratória e padrão.

Primeiramente é realizada uma busca unidimensional na direção do eixo x partindo-se do ponto inicial $Xk_0(xk_0, yk_0)$ considerando o tamanho do passo definido como a norma desse vetor que denotaremos por Δ e assumindo $j = 1$. Para assegurar que o próximo ponto obtido esteja contido no interior do polígono divide-se a norma por 10. Essa busca unidimensional é considerada uma busca exploratória, pois a partir de um determinado ponto inicial realiza-se uma procura do melhor posicionamento que a elipse pode assumir dentro do polígono desejado. Procura-se aumentar o semi-eixo maior da elipse na direção dada, caso isso não seja possível, faz-se uma nova expansão 10 vezes menor e verifica-se novamente a inscrição da elipse.

Definido o ponto inicial Xk_0 e o tamanho do passo pré-definido, a próxima etapa consiste em testar a função objetivo a ser maximizada realizando perturbações sucessivas no ponto inicial, com direções de busca $d_1, \ldots, d_n$. O movimento exploratório inicial faz uma análise da função objetivo no ponto Xk_0, considerando duas hipóteses:

- na direção de d_j assumindo $Xk_{j+1} = Xk_j + \Delta \cdot d_j$;

- na direção contrária a d_j assumindo $Xk_{j+1} = Xk_j - \Delta \cdot d_j$.

Esses dois pontos são afastados do Xk_1 por $\triangle$. Com o movimento exploratório realizado podem surgir duas situações [4]:

- Se algum dos dois pontos resultar em um crescimento do valor da função objetivo, ou seja, se $f(Xk_j + \triangle \cdot d_j) \geq f(Xk_j)$, para o caso da maximização, então pode se considerar que a busca exploratória ocorreu de forma satisfatória. Nesse caso, o ponto que produziu sucesso na busca é armazenado como ponto temporário que chamamos de Xk_2 para a continuação da busca e substituímos Xk_1 por Xk_2;
- A segunda hipótese é que pode ocorrer que nenhum dos dois pontos tenha produzido um crescimento no valor da função objetivo, ou seja, ambos os pontos resultaram em $f(Xk_j + \triangle \cdot d_j) \leq f(Xk_j)$. Nesse caso, o tamanho do passo pré-definido como $\triangle$ é reduzido pela metade, sendo assim o passo será $\frac{\triangle}{2}$ e a busca exploratória com Xk_1 é realizada novamente.

Cabe ressaltar que, primeiramente, é realizada a avaliação da função objetivo de $Xk_{j+1} = Xk_j + \Delta \cdot d_j$ e só é realizado o teste com o ponto $Xk_{j+1} = Xk_j - \Delta \cdot d_j$ se o primeiro resultar em uma função objetivo com valor menor que a função objetivo no ponto anterior, ou seja, $f(Xk_j + \Delta \cdot d_j) \leq f(Xk_j)$.

Se a fase de busca exploratória obtiver sucesso irá produzir um novo ponto base, obtendo uma direção de busca que pode ser definida como a distância entre o ponto base e o ponto inicial, descrita pela Equação 4.10:

$$d = |Xk_2 - Xk_1| \tag{4.10}$$

Tem-se, então, como novo ponto base Xk_2 e a próxima busca exploratória é realizada, considerando a Equação 4.11:

$$\begin{aligned} Xk_3 &= |Xk_1 + 2 \cdot d| \\ Xk_3 &= |Xk_2 + d| \end{aligned} \tag{4.11}$$

Se a segunda exploração não for bem sucedida e não obtiver melhoras na maximização da $f(Xk_2)$, então uma busca exploratória com centro no ponto Xk_2 é realizada. Caso o processo falhe, novamente o tamanho do passo Δ é reduzido e repete-se a busca.

O procedimento pode ser modificado de modo que passos de tamanhos diferentes sejam utilizados ao longo de distintas direções, essa adaptação algumas vezes é realizada para efeito de dimensionamento.

Define-se $\Delta = \frac{\sqrt{x_k \cdot x_k}}{10}$ e calcula-se $f(x_{k+i-1} + \Delta \cdot d_{k+i-1})$ enquanto a condição $x_{k+i-1} + \Delta \cdot d_{k+i-1} \in S$, sendo S a superfície do polígono.

4.1 Descrição da busca exploratória

A busca exploratória é realizada utilizando a parametrização de segmentos, rotação e translação pelo centro [8, 29, 41]. Inicialmente, considera-se a matriz descrita por 4.12:

$$vx = \begin{bmatrix} xv_1 & xv_2 & \cdots & xv_{np} \\ yv_1 & yv_2 & \cdots & yv_{np} \end{bmatrix} \tag{4.12}$$

como a matriz que contém os vértices vx_i tendo $i = 0, \ldots, np-1$ sendo np o número total de vértices e $\overrightarrow{V_0} = \overrightarrow{V_{np}}$.

Seja n um parâmetro auxiliar, inserido via teclado pelo usuário do programa, que determinará a formação da matriz de pontos dentro do contorno pelo qual o segmento será inserido e testado. Para que seja realizada a translação da elipse, a partir do centro $C(x_c, y_c)$, é necessário que seja feita a parametrização do segmento de reta que une dois vértices, o que será descrito no Teorema 4.1.1 proposto por [11] e adaptado para o polígono.

Teorema 4.1.1. *Seja $(0, \overrightarrow{e_1}, \overrightarrow{e_2})$ um sistema de coordenadas no plano. Sejam $X = (x, y)$, $A = (x_0, y_0)$ e $\overrightarrow{v} = (a, b)$, respectivamente um ponto genérico de r, um ponto dado de r e um vetor $\overrightarrow{v}$ não nulo de direção paralela à reta r.*

Da equação vetorial da reta r:

$$X = A + t\overrightarrow{v} \qquad (t \in \mathbb{R})$$

vem:

$$(x, y) = (x_0, y_0) + t(a, b)$$

Logo:

$$\begin{cases} x = x_0 + ta \\ y = y_0 + tb \end{cases} \quad (t \in \mathbb{R}) \tag{4.13}$$

em que a e b não são todos nulos, pois o vetor $\overrightarrow{v} \neq \overrightarrow{0}$.

As Equações 4.13 dizem-se ***equações paramétricas da reta r****, em relação ao sistema de coordenadas fixado.*

Reciprocamente, dado um sistema de equações lineares 4.14, com a e b não todos nulos, ele representa uma reta r do plano, pois a reta que passa pelo ponto de coordenadas (x_0, y_0) *e tem a direção do vetor não nulo de coordenadas* (a, b) *terá como equações paramétricas as 4.14. No caso de r ser definida por dois pontos* $A = (x_1, y_1)$ *e* $B = (x_2, y_2)$ *as equações paramétricas de r serão:*

$$\begin{cases} x = x_1 + t(x_2 - x_1) \\ y = y_1 + t(y_2 - y_1) \end{cases} \quad (t \in \mathbb{R}) \tag{4.14}$$

Conhecidos os vértices que descrevem o contorno do maior plano e, com o parâmetro n já determinado, o próximo passo é determinar $dt = \frac{1}{n}$ que será utilizado na parametrização, levando em consideração a divisão do semi-eixo menor em n segmentos.

Com relação aos vértices que formam o contorno, tem-se que (xv, yv) são as coordenadas de cada vértice do polígono e (xva, yva) são as coordenadas do vértice anterior obtidas no sentido anti-horário. Considerando o vetor do vértice anterior $vxa_i(xva_i, yva_i)$ como fixo, percorrem-se os demais vértices para realizar a parametrização dos segmentos que unem dois vértices. Para cada vértice que pertence a matriz vx, é feita a seguinte parametrização descrita na Equação 4.15:

$$\begin{cases} xt = xv_i + t \cdot (xva_i - xv_i) \\ yt = yv_i + t \cdot (yva_i - yv_i) \end{cases} \tag{4.15}$$

onde $t = k \cdot dt$ com $k = 0, 1, 2, \ldots, n$. Então, o valor de t varia no intervalo $[0, 1]$ pelo critério de parametrização adotado.

Quando se atribui valores intermediários, obtemos (xt, yt) percorrendo um segmento entre dois vértices. Quanto maior o valor atribuído ao parâmetro n, maior será

o número de pontos gerados ao longo do segmento, por essa razão defini-se n como o número de refinamentos na aresta do polígono.

A seguir é realizada a translação [8, 19] a partir do centro, percorrendo os vértices no sentido anti-horário, definida pela Equação 4.16:

$$\begin{cases} z_t = x_v - x_c \\ w_t = y_v - y_c \end{cases} \tag{4.16}$$

Após a translação, o próximo passo é realizar a rotação considerando o ângulo de rotação σ [8, 19, 41]. Neste Capítulo será abordada a rotação aplicada ao problema de posicionamento de ε, utilizando os conceitos abordados no Capítulo 3.

Supondo que giramos todos os pontos do $\mathbb{R}^2$ no sentido anti-horário de um ângulo σ, em torno da origem de um sistema de coordenadas cartesianas, ou seja, a rotação em relação ao centro transladado (z_t, w_t). Então, se o centro C tem coordenadas (z_t, w_t), depois da rotação obtemos um novo centro C' com coordenadas (r_t, s_t). Para obter a relação entre as coordenadas de C' e C, consideramos v_0 vetor da origem até $C(z_t, w_t)$, representado por r o comprimento desse vetor v, temos a relação descrita pela Equação 4.17:

$$\begin{cases} z_t = r \cdot \cos\theta \\ w_t = r \cdot \sin\theta \end{cases} \tag{4.17}$$

e

$$\begin{cases} r_t = r \cdot \cos(\theta + \sigma) \\ s_t = r \cdot \sin(\theta + \sigma) \end{cases} \tag{4.18}$$

Se um sistema de coordenadas zw faz um ângulo σ com um sistema rs e ambos têm a mesma origem, valem as relações estabelecidas em 4.19:

$$\begin{cases} r_t = z_t \cdot \cos\sigma + w_t \cdot \sin\sigma \\ s_t = -z_t \cdot \sin\sigma + w_t \cdot \cos\sigma \end{cases} \tag{4.19}$$

Os sistemas de coordenadas e a rotação de σ é exposto na Figura 4.4. Se o vetor (r_t, s_t) não estiver contido na região descrita pelo contorno, assume-se que (xva, yva) será o vértice (xv, yv).

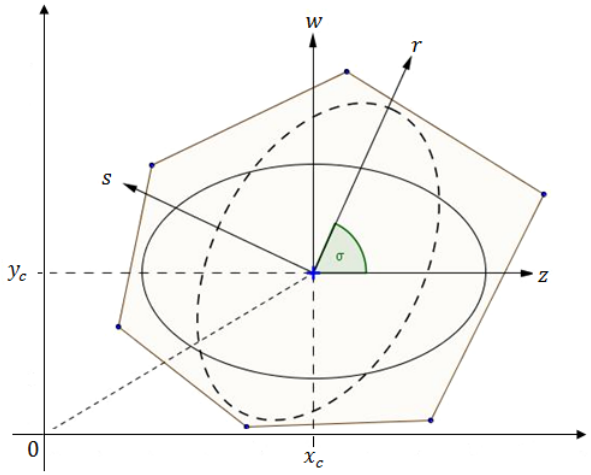


Figura 4.4: Rotação da elipse inscrita no polígono irregular. Fonte: Elaborado pela autora.

4.2 Descrição da busca padrão utilizando o método de Hooke e Jeeves

O método de Hooke e Jeeves [4], descrito anteriormente, realiza dois tipos de busca: a busca exploratória e a busca padrão. A busca padrão tem por objetivo calcular a nova direção de busca para alocar da melhor maneira possível o semi-eixo maior da elipse, obtendo assim a elipse inicial com a excentricidade pré-definida. O objetivo da busca unidimensional exploratória é definir um novo ponto que será o centro da elipse e, posteriormente, tenta-se aumentar o semi-eixo maior. Caso isso não seja possível, o tamanho do passo Δ é reduzido e a expansão é menor [47]. A busca exploratória ocorre ao longo das direções canônicas dada pela Equação 4.20:

$$d_{k_i+N} = \frac{x_{k_i+N} - x_{k_i}}{|x_{k_i+N} - x_{k_i}|_2} \tag{4.20}$$

Na busca exploratória, designam-se as direções canônicas por N. Note que na Equação 4.20 considera-se um vetor unitário, pois o que interessa é a direção que está sendo indicada e não o comprimento do vetor.

A primeira direção canônica tenta realizar uma mudança na coordenada x_c e pode ser escrita como $d_{k_i} = (1, 0, 0)$, a segunda direção canônica assumida tenta alterar a coordenada y_c e podemos descrevê-la como $d_{k_i} = (0, 1, 0)$ e, por fim, a terceira direção canônica realiza mudanças no ângulo de rotação σ, e descrita por $d_{k_i} = (0, 0, 1)$.

Realizada a busca exploratória ao longo das direções canônicas, parte-se então para a busca padrão ao longo de uma direção que é determinada por x_{k_i} e x_{k_i+N} [47]. Dessa forma, dada a função a ser maximizada $f(x_{k_i+N}+\Delta\cdot d_{k_i+N})$, com $\Delta_{k_i+N}\in\mathbb{R}$ de modo que $x_{k_i+N}+\Delta\cdot d_{k_i+N}\in S$, e definindo-se como nova aproximação para o ponto máximo de f restrita a superfície do polígono, $x_{k_i+N+1}=x_{k_i+N}+\Delta_{k_i+N}\cdot d_{k_i+N}$. Conforme o algoritmo de maximização utilizando o método de Hooke e Jeeves [4], quando a tolerância for alcançada retornamos para x_{k_i}, caso contrário, assumimos k_i como k_i+N e definimos novamente $\Delta=\frac{\sqrt{x_k\cdot x_k}}{10}$ e calcula-se $f(x_{k+i-1}+\Delta\cdot d_{k+i-1})$ enquanto a condição $x_{k+i-1}+\Delta\cdot d_{k+i-1}\in S$, sendo S a superfície do polígono P.

4.3 Parâmetros obtidos com o posicionamento de ε

Para a obtenção dos parâmetros que serão utilizados no posicionamento das demais elipses faz-se uma simulação aplicando o método de otimização de Hooke e Jeeves no algoritmo de inscrição de secções cônicas em polígonos não-convexos de n-lados. A partir das simulações realizadas com o auxílio do *software* MatLab é possível analisar o processo de otimização de ε inscrita num polígono. Foram atribuídos valores iniciais como:

- a excentricidade fixa em 0,593946 para o modelo de lapidação oval;
- o valor de n: número de pontos em cada aresta do polígono (discretização);
- a tolerância tol;
- o número máximo de iterações.

Após cada iteração do processo de inscrição de ε, obtemos as coordenadas do novo centro $C(x_c, y_c)$, um ângulo σ de rotação no sentido anti-horário e o comprimento do semi-eixo maior aw da elipse inscrita. A Figura 4.5 mostra a inscrição de ε:

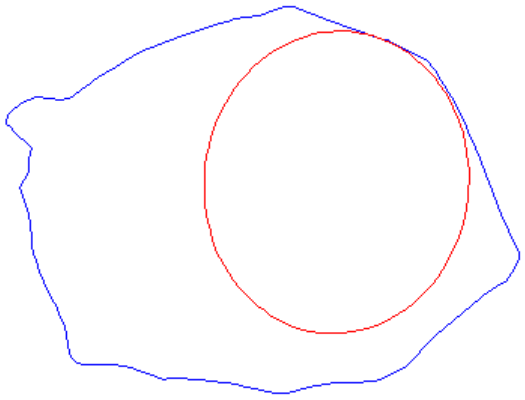


Figura 4.5: Inscrição da elipse inicial em um polígono P. Fonte: Elaborado pela autora.

A elipse inscrita será utilizada para o empacotamento das demais elipses tangentes em P com a metodologia analítica desenvolvida.

5 Metodologia analítica para o empacotamento de elipses tangentes aos vértices de ε

Neste capítulo, descreve-se detalhadamente a metodologia analítica para o empacotamento de elipses tangentes aos vértices da elipse inicial bem como o cálculo das coordenadas dos vértices da elipse inicial. Apresenta ainda, o cálculo dos semi-eixos, do centro e dos pontos aleatórios de cada elipse tangente.

Inicialmente, define-se um critério de parada para ε. Além da tolerância estimada, o critério 5.1 adotado seta o número máximo de iterações que o usuário deseja realizar levando em consideração que ε não ocupe toda a superfície do polígono:

$$i_{max} = 10 \tag{5.1}$$

Com os dados obtidos através da inscrição de ε é calculado o comprimento do semi-eixo menor da elipse que será utilizado na estimativa da área da elipse e na projeção dos vértices. Para diferenciar o semi-eixo maior de ε na iteração inicial utilizamos a e na inscrição final aw. A área de ε é definida em 5.2 como:

$$A = \pi \cdot aw \cdot bw \tag{5.2}$$

A área do polígono P é calculada com os vértices e o centro do polígono utilizando um processo de triangulação. O centro geométrico $C(x_{cg}, y_{cg})$ do polígono definido por vx é calculado utilizando o mesmo método aplicado para encontrar o ponto inicial de busca do método de Hooke e Jeeves, descrito por 5.3:

$$x_{cg} = \frac{\sum_{i=1}^{np} xv_i}{np} \quad e \quad y_{cg} = \frac{\sum_{i=1}^{np} yv_i}{np} \tag{5.3}$$

onde xv_i e yv_i são as coordenadas dos vértices e np indica o número de vértices que formam o polígono.

Definido o centro, divide-se o polígono em n-triângulos de acordo com o número de vértices. A área de cada triângulo é obtida com a Fórmula de Heron, visando uma estabilidade numérica e evitando a sobreposição dos triângulos formados pela união de dois vértices e o centro conforme mostrado na Figura 5.1:

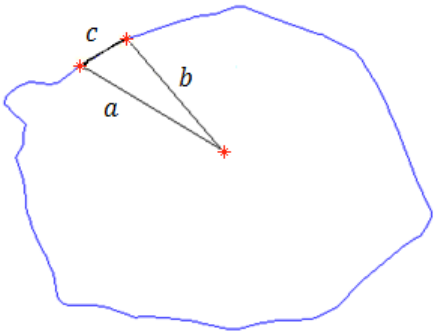


Figura 5.1: Composição de triângulos pela Fórmula de Heron. Fonte: Elaborado pela autora.

De acordo com [28], quando as medidas a,b e c dos lados dos triângulos obedecem o critério imposto de que $a \geq b \geq c$. Se $c - (a - b) < 0$ temos que os lados do triângulo não obedecem o critério acima, caso contrário, calcula-se a área pela Fórmula de Heron descrita na Equação 5.4:

$$A = \sqrt{p \cdot (p-a) \cdot (p-b) \cdot (p-c)} \tag{5.4}$$

sendo p o semi perímetro do triângulo de lados a, b e c e descrito na Equação 5.5:

$$p = \frac{a+b+c}{2} \tag{5.5}$$

de modo que é possível obter a Equação 5.6 equivalente a Equação 5.4:

$$A = \sqrt{\left(\frac{a+b+c-2a}{2}\right) \cdot \left(\frac{a+b+c-2b}{2}\right) \cdot \left(\frac{a+b+c-2c}{2}\right) \cdot \left(\frac{a+b+c}{2}\right)} \tag{5.6}$$

realizando a simplificação da Equação 5.6, temos:

$$A = \frac{1}{4} \cdot \sqrt{(-a+b+c) \cdot (a-b+c) \cdot (a+b-c) \cdot (a+b+c)} \tag{5.7}$$

para verificar o critério estabelecido por [28], definimos:

$$A = \frac{1}{4}\sqrt{(a+(b+c)).(c-(a-b)).(c+(a-b)).(a+(b-c))} \tag{5.8}$$

Como forma de garantir a eficiência do cálculo e obedecendo aos critérios da radiciação, devem-se associar os parênteses de modo que não resulte em uma área negativa. A área de cada triângulo é utilizada para determinar a área total do polígono P através do somatório descrito na Equação 5.9:

$$A_P = \sum_{i=1}^{nt} A_i \tag{5.9}$$

onde A_i é a área de cada triângulo i e nt o número de triângulos gerados pela união do centro e dois vértices.

A área restante para o posicionamento das elipses tangentes é obtida pela diferença entre a área do polígono e a área de ε. Com a minimização da área restante podem resultar duas situações:

1 - Maximização da área ocupada pelas elipses $\longrightarrow$ maior aproveitamento volumétrico da gema;

2 - Elipses com pontos $P_{(x,y)}$ exteriores ao contorno $\longrightarrow$ define-se a matriz 5.10 que contém os np-pontos de cada elipse (vértices, ponto de tangência, centro e pontos aleatórios):

$$E = \begin{bmatrix} Ex_i & Ex_{i+1} & \cdots & Ex_{np} \\ Ey_i & Ey_{i+1} & \cdots & Ey_{np} \end{bmatrix} \tag{5.10}$$

A matriz 5.10 é utilizada no teste para a verificação de cada ponto em relação ao polígono, onde cada ponto pode estar contido ou não no polígono ou ainda coincidir com um vértice:

- $P_{(x,y)}(in) \mapsto P_{(x,y)} \in P$
- $P_{(x,y)}(\sim in) \mapsto P_{(x,y)} \notin P$

Os pontos $P_{(x,y)}(in)$ que pertencem ao polígono são armazenados em uma matriz E_{in} bem como os pontos $P_{(x,y)}(\sim in)$ são alocados em uma matriz $E_{\sim in}$. Se não existir nenhum ponto $P_{(x,y)}$ exterior ao polígono então a matriz $E_{\sim in}$ será vazia. Caso contrário, aplica-se o processo de refinamento do semi-eixo maior da elipse que esta sendo inserida até obter a matriz $E_{\sim in}$ sem nenhum elemento.

Quanto menor a área restante, maior a probabilidade da matriz $E_{\sim in}$ conter algum ponto $P_{(x,y)}$, isso faz com que a busca da elipse inscrita em P torne-se mais demorada. Com a área de ε e do polígono definidas, a área restante é dada por:

$$A_R = -\left[(\pi \cdot aw \cdot bw) - \sum_{i=1}^{nt} A_i\right] \tag{5.11}$$

onde (aw, bw) são os semi-eixos utilizados no cálculo da área de ε.

Com o espaço disponível para posicionar as demais elipses tangentes a ε, o próximo passo é definir os vértices da elipse obtida e determinar graficamente os semi-eixos, ou seja, representar o vetor que une o centro ao vértice definido. Os vértices de ε são definidos em função do comprimento dos semi-eixos, basta adicionar ou subtrair o comprimento dos semi-eixos (aw, bw) às coordenadas do centro (x_c, y_c) levando em consideração o movimento de rotação realizado por essa elipse em torno do sistema de coordenadas original.

5.1 Cálculo das coordenadas dos vértices da elipse inicial

Os vértices são definidos levando em consideração o ângulo de rotação σ e armazenados em vetores coluna. Leva-se em consideração que vetores linha representam o sistema de coordenadas da mão esquerda no qual as rotações são realizadas no sentido horário. No vetor coluna, as rotações ocorrem no sentido anti-horário e representam a regra da mão direita, padrão em muitos sistemas e em computação gráfica (OpenGL) [19] mostrado na Figura 5.2.

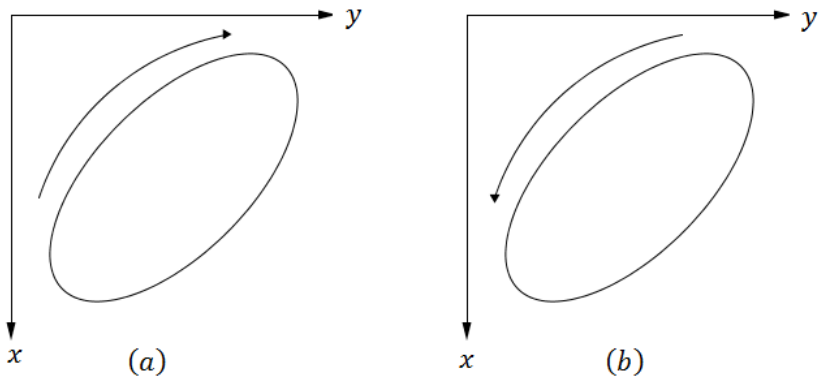


Figura 5.2: (a) Rotação: vetor linha - (b) Rotação: vetor coluna. Fonte: Elaborado pela autora.

Utilizamos o ângulo $-\sigma$ para definir o posicionamento dos vértices com relação a origem, sendo que o vértice $A1$ está posicionado dessa forma a esquerda da origem como mostra a Figura 5.3 e o vértice $B1$ a direita, mostrado na Figura 5.4.

A descrição dos vértices ocorre da mesma forma para todas as elipses. Dessa forma para simplificar o cálculo dos vértices são definidos os parâmetros abaixo:

- $x = x_c$;
- $y = y_c$;
- $a = aw$;
- $b = bw$;
- $rot = -\sigma$;
- $s = \sin(rot)$;
- $c = \cos(rot)$.

Vértice A1:

$$\begin{cases} \text{A1}x = x - a \cdot c \\ \text{A1}y = y + a \cdot s \end{cases}$$

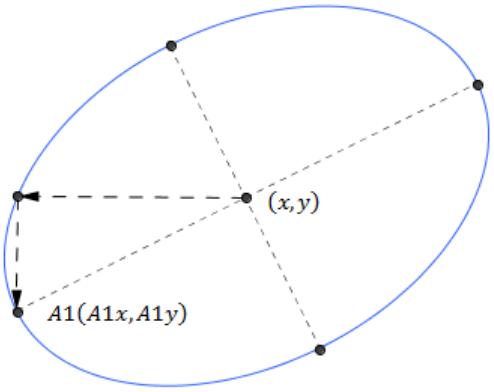


Figura 5.3: Representação gráfica do vértice A1 de ε. Fonte: Elaborado pela autora.

Vértice B1

$$\begin{cases} \text{B1}x = x + a \cdot c \\ \text{B1}y = y - a \cdot s \end{cases}$$

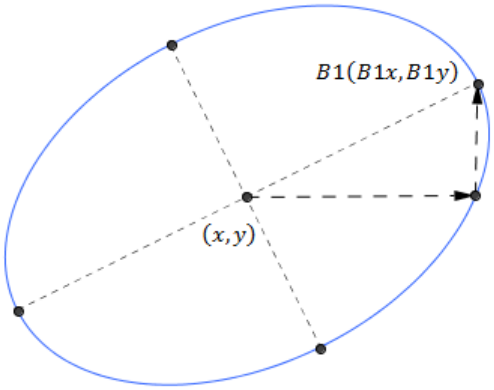


Figura 5.4: Representação gráfica do vértice B1 de ε. Fonte: Elaborado pela autora.

O semi-eixo maior é definido por $\overrightarrow{A1C}$ onde C é o centro da elipse. Dessa forma, estabelecemos que o eixo maior da elipse é definido por $\overrightarrow{A1B1}$.

Vértice C1

$$\begin{cases} \text{C1}x = x + b \cdot s \\ \text{C1}y = y + b \cdot c \end{cases}$$

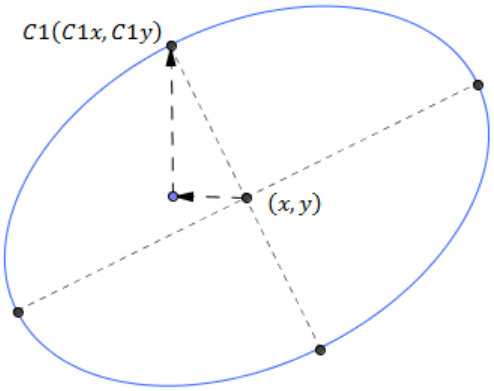


Figura 5.5: Representação gráfica do vértice C1 de ε. Fonte: Elaborado pela autora.

Vértice D1

$$\begin{cases} \mathrm{D1}x = x - b \cdot s \\ \mathrm{D1}y = y - b \cdot c \end{cases}$$

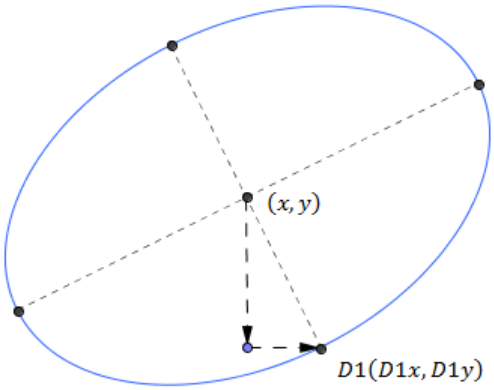


Figura 5.6: Representação gráfica do vértice D1 de ε. Fonte: Elaborado pela autora.

O semi-eixo menor é definido por $\overrightarrow{C1C}$ e o eixo menor da elipse é dado por $\overrightarrow{C1D1}$. Com os vértices e o centro de ε faz-se a representação mostrada na Figura 5.7:

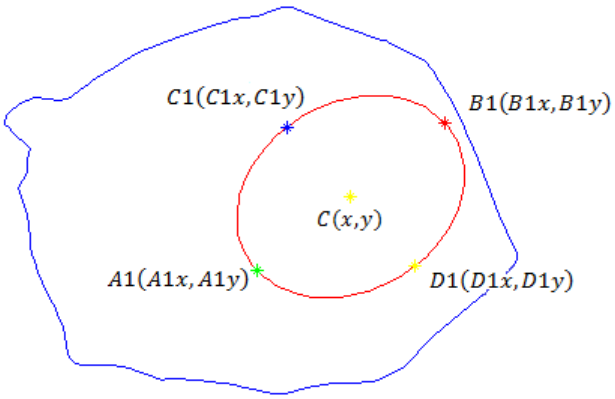


Figura 5.7: Representação geométrica dos vértices e do centro de ε. Fonte: Elaborado pela autora.

5.2 Posicionamento das elipses tangentes aos vértices da elipse inicial

O comprimento do semi-eixo maior da elipse tangente é calculado em função da distância do centro de ε até o vértice escolhido. Por efeito de dimensionamento, nos vértices $A1$ e $B1$ - que determinam o eixo maior de ε - o comprimento do semi-eixo maior da elipse tangente é definido pela equação 5.12:

$$aw = 0.5 \cdot \lambda, \quad \lambda \in \mathbb{R}^{+} \tag{5.12}$$

Entende-se por efeito de dimensionamento, a condição que deve ser obedecida para que não haja uma ou mais elipses sobrepostas. Se essa condição não for levada em consideração, ocorre a sobreposição de elipses na iteração inicial conforme a Figura 5.8:

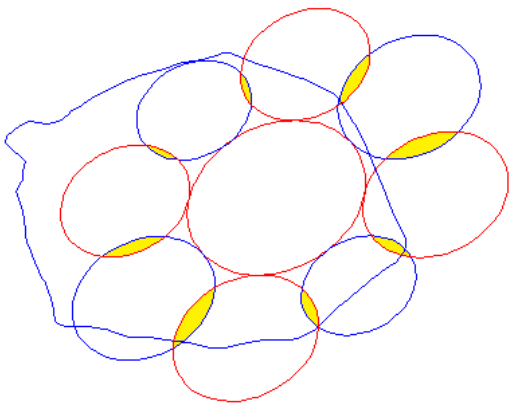


Figura 5.8: As áreas em destaque representam a sobreposição de elipses tangentes a ε. Fonte: Elaborado pela autora.

O efeito de dimensionamento baseia-se no comprimento dos semi-eixos de ε. As elipses que serão posicionadas nos vértices que compõe o eixo maior - vértices $A1$ e $B1$ - utilizam 50% da distância do vértice ao centro de ε. Já as elipses posicionadas nos vértices que descrevem o eixo menor - vértices $C1$ e $D1$ - utilizam 60% da distância entre o vértice e o centro de ε.

Estipula-se 50% e 60% da distância do centro aos vértices de ε pois dessa forma as elipses posicionadas possuem todas a mesma área e não são secantes às elipses posicionadas no ponto aleatório calculado em cada quadrante na iteração inicial, como pode ser visto na Figura 5.9:

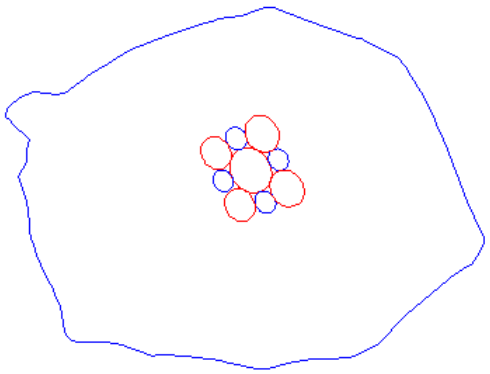


Figura 5.9: Elipses tangentes a ε com efeito de dimensionamento.Fonte: Elaborado pela autora.

Os parâmetros do efeito de dimensionamento foram estabelecidos com base nos testes de intersecção entre duas elipses. Nesse teste, verifica-se a posição dos pontos de uma elipse em relação a elipse vizinha com a função *inpolygon* disponível no MatLab [34].

5.2.1 Elipse posicionada no vértice $B1$ de ε

Especificamente, para a elipse tangente posicionada no vértice $B1$ de ε, temos o comprimento do semi-eixo maior dado pela Equação 5.13:

$$aw1 = 0.5 \cdot \lambda_1, \quad \lambda_1 \in \mathbb{R}^+ \tag{5.13}$$

O valor de λ_1 para a elipse tangente ao vértice $B1$, ou seja, quando $t = 0^o$ em ε é definido por 5.14:

$$\begin{cases} d1_x = |B1x - x_c| \\ d1_y = |B1y - y_c| \end{cases} \tag{5.14}$$

dessa forma, temos λ_1 em função da distância entre o centro (x_c, y_c) e o vértice $B1$ obtida por 5.15:

$$\lambda_1 = \sqrt{(d1_x)^2 + (d1_y)^2} \tag{5.15}$$

Considera-se a excentricidade fixa em $0,593946$ para todas as elipses, sendo assim o semi-eixo menor pode ser descrito por 5.16:

$$bw1 = aw1 \cdot \sqrt{1 - e^2} \tag{5.16}$$

O centro $C1(C1_x, C1_y)$ da elipse tangente posicionada no vértice $B1$ é obtido através da equação paramétrica, assumindo $t = 0^o$ e o ângulo de rotação σ, descrito em 5.17:

$$\begin{cases} C1_x = B1x + aw1 \cdot (\cos(0) \cdot \cos(\sigma)) - bw1 \cdot (\sin(0) \cdot \sin(\sigma)) \\ C1_y = B1y + aw1 \cdot (\cos(0) \cdot \sin(\sigma)) + bw1 \cdot (\sin(0) \cdot \cos(\sigma)) \end{cases} \tag{5.17}$$

Com o centro definido, parte-se para o cálculo dos vértices. Denominamos os vértices da primeira elipse tangente a ε fazendo referência a elipse e o respectivo vértice calculado.

Vértice E1A1:

$$\begin{cases} \text{e1A1}x = C1_x - aw1 \cdot c \\ \text{e1A1}y = C1_y + aw1 \cdot s \end{cases}$$

Vértice E1B1:

$$\begin{cases} \text{e1B1}x = C1_x + aw1 \cdot c \\ \text{e1B1}y = C1_y - aw1 \cdot s \end{cases}$$

Vértice E1C1:

$$\begin{cases} \text{e1C1}x = C1_x + bw1 \cdot s \\ \text{e1C1}y = C1_y + bw1 \cdot c \end{cases}$$

Vértice E1D1:

$$\begin{cases} \text{e1D1}x = C1_x - bw1 \cdot s \\ \text{e1D1}y = C1_y - bw1 \cdot c \end{cases}$$

O eixo menor dessa elipse é definido por $\overrightarrow{E1C1E1D1}$ e o eixo maior é definido por $\overrightarrow{E1A1E1B1}$. Com o centro e os vértices definidos, utilizamos a equação paramétrica 5.18 com centro deslocado da origem e rotacionada [8, 19] com relação ao sistema de eixos cartesianos para plotar a elipse mostrada na Figura 5.10:

$$\begin{cases} x1(t) = C1_x + aw1 \cdot (\cos(t) \cdot \cos(\sigma)) - bw1 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y1(t) = C1_y + aw1 \cdot (\cos(t) \cdot \sin(\sigma)) + bw1 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{5.18}$$

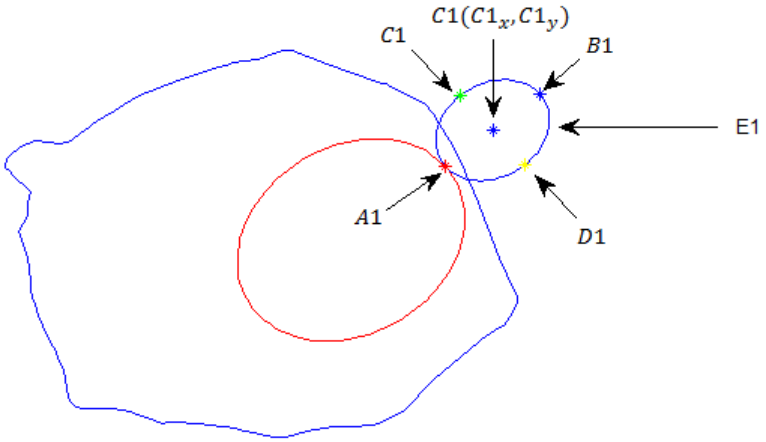


Figura 5.10: Elipse tangente ao vértice B1 de ε. Fonte: Elaborado pela autora.

Pela Figura 5.10 é possível observar que o vértice $B1$ de ε coincide com o vértice $A1$, condição estabelecida pela restrição de tangência. Para a verificação da inscrição da elipse no polígono é necessário definir um ponto em cada quadrante que mantém contato direto com o polígono. Esse ponto é a intersecção da bissetriz do quadrante com a curva

que descreve a elipse. Como o método de otimização de Hooke e Jeeves [4] realiza buscas sempre na direção do contorno, na elipse 1 os quadrantes 1 e 4 são os que possuem maior contato com o polígono. A busca realizada pelo método de Hooke e Jeeves posiciona ε próxima ao contorno do polígono na direção assumida logo as elipses tangentes possuem alguns pontos exteriores ao contorno.

Os pontos aleatórios em cada quadrante são calculados através da intersecção da bissetriz com a curva. Para cada quadrante existe o valor do ângulo definido conforme mostrado na Figura 5.11:

- **1º Quadrante:** $t = \frac{\pi}{4}$;
- **2º Quadrante:** $t = \frac{3\pi}{4}$;
- **3º Quadrante:** $t = \frac{5\pi}{4}$;
- **4º Quadrante:** $t = \frac{7\pi}{4}$.

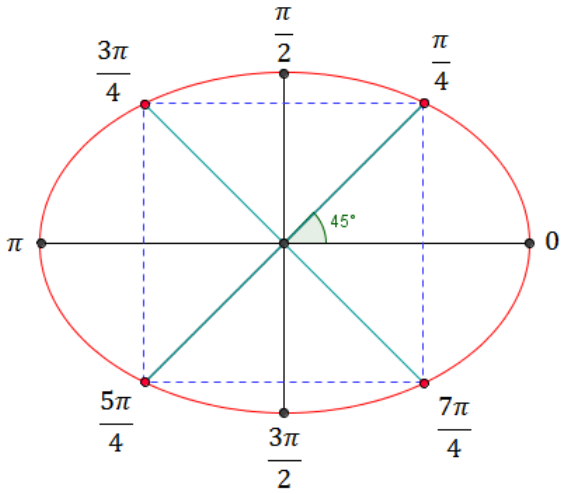


Figura 5.11: Pontos aleatórios em cada quadrante. Fonte: Elaborado pela autora.

Com a generalização dos pontos aleatórios em cada quadrante realizada na Figura 5.11, calculamos os pontos aleatórios no 1º e 4º quadrantes da elipse 1 tangente ao vértice $B1(B1x, B1y)$ de ε pelas Equação 5.19 e 5.20:

- **1º quadrante:** $t = \frac{\pi}{4}$

$$\begin{cases} PA1_x = C1_x + aw1 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw1 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ PA1_y = C1_y + aw1 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw1 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.19}$$

- **4º quadrante:** $t = \frac{7\pi}{4}$

$$\begin{cases} PA2_x = C1_x + aw1 \cdot (\cos(\frac{7\pi}{4}) \cdot \cos(\sigma)) - bw1 \cdot (\sin(\frac{7\pi}{4}) \cdot \sin(\sigma)) \\ PA2_y = C1_y + aw1 \cdot (\cos(\frac{7\pi}{4}) \cdot \sin(\sigma)) + bw1 \cdot (\sin(\frac{7\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.20}$$

Os pontos aleatórios calculados no 1º e 4º quadrantes da elipse tangente ao vértice $B1(B1x, B1y)$ são mostrados na Figura 5.12:

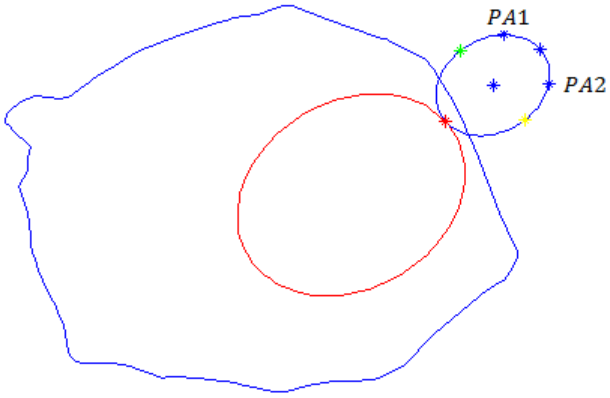


Figura 5.12: Pontos aleatórios no 1º e 4º quadrante. Fonte: Elaborado pela autora.

A matriz $E1$ que armazena os pontos utilizados para realizar a verificação da inscrição da elipse com a função *inpolygon* do Matlab é formada pelos vértices, centro e pontos aleatórios dos quadrantes estabelecidos:

$$E1 = \begin{bmatrix} e1A1x & e1B1x & e1C1x & e1D1x & C1_x & PA1_x & PA2_x \\ e1A1y & e1B1y & e1C1y & e1D1y & C1_y & PA1_y & PA2_y \end{bmatrix} \tag{5.21}$$

5.2.2 Elipse posicionada no vértice *C1* de ε

O semi-eixo maior da elipse posicionada no vértice *C1* de ε é calculado pela Equação 5.22:

$$aw2 = 0.6 \cdot \lambda_2, \quad \lambda_2 \in \mathbb{R}^+ \tag{5.22}$$

sendo definido quando $t = \frac{\pi}{2}$ em ε e o valor de λ_2 dado por:

$$\begin{cases} d2_x = |C1x - x_c| \\ d2_y = |C1y - y_c| \end{cases} \tag{5.23}$$

dessa forma λ_2 é calculado em função da distância do centro $C(x_c, y_c)$ ao vértice $C1$, obtida pela Equação 5.24:

$$\lambda_2 = \sqrt{(d2_x)^2 + (d2_y)^2} \tag{5.24}$$

O semi-eixo menor dessa elipse é obtido com a Equação 5.25:

$$bw2 = aw2 \cdot \sqrt{1 - e^2} \tag{5.25}$$

O centro $C2(C2_x, C2_y)$ da elipse tangente ao vértice $C1$, ou seja, em $t = \frac{\pi}{2}$ em ε é obtido pela Equação 5.26:

$$\begin{cases} C2_x = C1x + aw2 \cdot (\cos(\frac{\pi}{2}) \cdot \cos(\sigma)) - bw2 \cdot (\sin(\frac{\pi}{2}) \cdot \sin(\sigma)) \\ C2_y = C1y + aw2 \cdot (\cos(\frac{\pi}{2}) \cdot \sin(\sigma)) + bw2 \cdot (\sin(\frac{\pi}{2}) \cdot \cos(\sigma)) \end{cases} \tag{5.26}$$

Os vértices da elipse 2 posicionada no vértice $C1$ de ε com centro $C2(C2_x, C2_y)$ são calculados da seguinte forma:

Vértice E2A1:

$$\begin{cases} \text{e2A1}x = C2_x - aw2 \cdot c \\ \text{e2A1}y = C2_y + aw2 \cdot s \end{cases}$$

Vértice E2B1:

$$\begin{cases} \text{e2B1}x = C2_x + aw2 \cdot c \\ \text{e2B1}y = C2_y - aw2 \cdot s \end{cases}$$

Vértice E2C1:

$$\begin{cases} \text{e2C1}x = C2_x + bw2 \cdot s \\ \text{e2C1}y = C2_y + bw2 \cdot c \end{cases}$$

Vértice E2D1:

$$\begin{cases} \text{e2D1}x = C2_x - bw2 \cdot s \\ \text{e2D1}y = C2_y - bw2 \cdot c \end{cases}$$

O eixo menor é definido por $\overrightarrow{E2C1E2D1}$ e o eixo maior por $\overrightarrow{E2A1E2B1}$. A equação paramétrica 5.27 representa a elipse 2 mostrada na Figura 5.13 :

$$\begin{cases} x2(t) = C2_x + aw2 \cdot (\cos(t) \cdot \cos(\sigma)) - bw2 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y2(t) = C2_y + aw2 \cdot (\cos(t) \cdot \sin(\sigma)) + bw2 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{5.27}$$

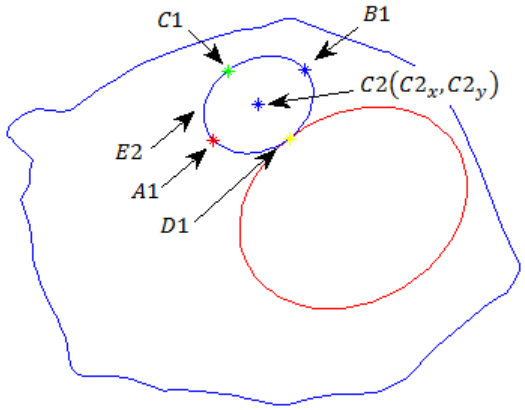


Figura 5.13: Elipse tangente ao vértice C1 de ε. Fonte: Elaborado pela autora.

Para a elipse 2, o 1º e 2º quadrantes são os que possuem a maior probabilidade de possuírem alguns pontos exteriores ao polígono. Nesses quadrantes são definidos os pontos aleatórios em função da intersecção da bissetriz do quadrante com a curva que descreve a elipse:

- **1º quadrante:** $t = \frac{\pi}{4}$

$$\begin{cases} PA3_x = C2_x + aw2 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw2 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ PA3_y = C2_y + aw2 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw2 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.28}$$

- **2º quadrante:** $t = \frac{3\pi}{4}$

$$\begin{cases} PA4_x = C2_x + aw2 \cdot (\cos(\frac{3\pi}{4}) \cdot \cos(\sigma)) - bw2 \cdot (\sin(\frac{3\pi}{4}) \cdot \sin(\sigma)) \\ PA4_y = C2_y + aw2 \cdot (\cos(\frac{3\pi}{4}) \cdot \sin(\sigma)) + bw2 \cdot (\sin(\frac{3\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.29}$$

Os pontos aleatórios $PA3$ e $PA4$ calculados nos quadrantes que mantém maior contato com o contorno são mostrados na Figura 5.14:

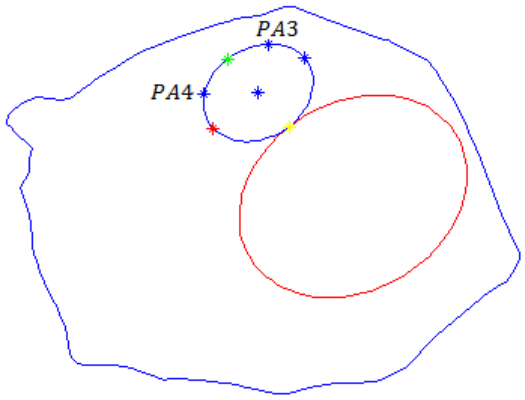


Figura 5.14: Pontos aleatórios no 1º e 2º quadrante. Fonte: Elaborado pela autora.

A matriz que armazena os pontos para a verificação da inscrição da elipse no polígono é definida por:

$$E2 = \begin{bmatrix} e2A1x & e2B1x & e2C1x & e2D1x & C2_x & PA3_x & PA4_x \\ e2A1y & e2B1y & e2C1y & e2D1y & C2_y & PA3_y & PA4_y \end{bmatrix} \tag{5.30}$$

5.2.3 Elipse posicionada no vértice $A1$ de ε

A elipse posicionada no vértice $A1$ de ε tem o semi-eixo maior definido pela Equação 5.31:

$$aw3 = 0.5 \cdot \lambda_3, \;\; \lambda_3 \in \mathbb{R}^+ \tag{5.31}$$

de modo que λ_3 é obtido calculando-se a distância do centro (x_c, y_c) ao vértice $A1$ de ε, ou seja, quando $t = \pi$:

$$\begin{cases} d3_x = |A1x - x_c| \\ d3_y = |A1y - y_c| \end{cases} \tag{5.32}$$

onde λ_3 é definido pela Equação 5.33:

$$\lambda_3 = \sqrt{(d3_x)^2 + (d3_y)^2} \tag{5.33}$$

e o semi-eixo menor da elipse é dado por:

$$bw3 = aw3 \cdot \sqrt{1 - e^2} \tag{5.34}$$

O centro $C3(C3_x, C3_y)$ da elipse tangente ao vértice $A1$ é obtido pela equação paramétrica 5.35, definindo $t = \pi$:

$$\begin{cases} C3_x = A1x + aw3 \cdot (\cos(\pi) \cdot \cos(\sigma)) - bw3 \cdot (\sin(\pi) \cdot \sin(\sigma)) \\ C3_y = A1y + aw3 \cdot (\cos(\pi) \cdot \sin(\sigma)) + bw3 \cdot (\sin(\pi) \cdot \cos(\sigma)) \end{cases} \tag{5.35}$$

O cálculo dos vértices da elipse 3 baseia-se no ângulo de rotação, coordenadas do centro e comprimento dos semi-eixos.

Vértice E3A1:

$$\begin{cases} \text{e3A1}x = C3_x - aw3 \cdot c \\ \text{e3A1}y = C3_y + aw3 \cdot s \end{cases}$$

Vértice E3B1:

$$\begin{cases} \text{e3B1}x = C3_x + aw3 \cdot c \\ \text{e3B1}y = C3_y - aw3 \cdot s \end{cases}$$

Vértice E3C1:

$$\begin{cases} \text{e3C1}x = C3_x + bw3 \cdot s \\ \text{e3C1}y = C3_y + bw3 \cdot c \end{cases}$$

Vértice E3D1:

$$\begin{cases} \text{e3D1}x = C3_x - bw3 \cdot s \\ \text{e3D1}y = C3_y - bw3 \cdot c \end{cases}$$

O eixo maior é definido por $\overrightarrow{E3A1E3B1}$ bem como o eixo menor é definido por $\overrightarrow{E3C1E3D1}$. A equação paramétrica 5.36 é utilizada para plotar a elipse 3 mostrada na Figura 5.15:

$$\begin{cases} x3(t) = C3_x + aw3 \cdot (\cos(t) \cdot \cos(\sigma)) - bw3 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y3(t) = C3_y + aw3 \cdot (\cos(t) \cdot \sin(\sigma)) + bw3 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{5.36}$$

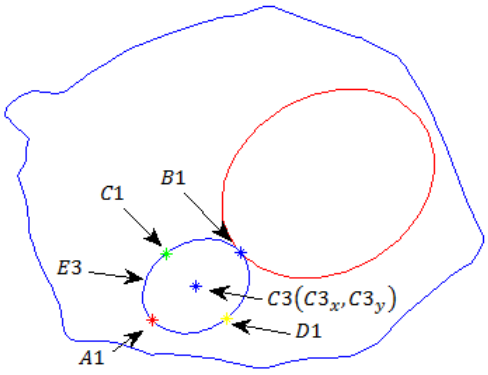


Figura 5.15: Elipse tangente ao vértice A1 de ε. Fonte: Elaborado pela autora.

O 2º e 3º quadrantes são os que possuem a maior probabilidade de conter alguns pontos exteriores ao polígono, por isso será definido um ponto aleatório com a Equação 5.37 nesses quadrantes através da intersecção da bissetriz do quadrante com a curva.

- **2º quadrante:** $t = \frac{3\pi}{4}$

$$\begin{cases} PA5_x = C3_x + aw3 \cdot (\cos(\frac{3\pi}{4}) \cdot \cos(\sigma)) - bw3 \cdot (\sin(\frac{3\pi}{4}) \cdot \sin(\sigma)) \\ PA5_y = C3_y + aw3 \cdot (\cos(\frac{3\pi}{4}) \cdot \sin(\sigma)) + bw3 \cdot (\sin(\frac{3\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.37}$$

- **3º quadrante:** $t = \frac{5\pi}{4}$

$$\begin{cases} PA6_x = C3_x + aw3 \cdot (\cos(\frac{5\pi}{4}) \cdot \cos(\sigma)) - bw3 \cdot (\sin(\frac{5\pi}{4}) \cdot \sin(\sigma)) \\ PA6_y = C3_y + aw3 \cdot (\cos(\frac{5\pi}{4}) \cdot \sin(\sigma)) + bw3 \cdot (\sin(\frac{5\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.38}$$

Na Figura 5.16 é possível observar os pontos aleatórios localizados no 2º e 3º quadrantes da elipse tangente ao vértice $A1$ de ε.

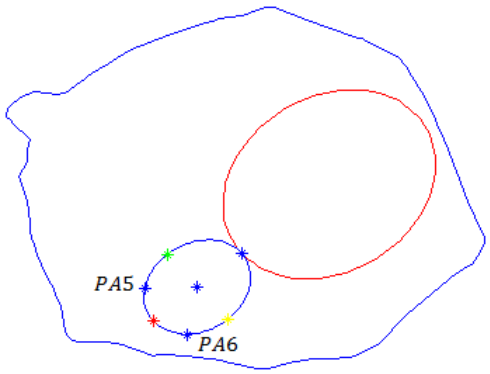


Figura 5.16: Pontos aleatórios no 2º e 3º quadrantes. Fonte: Elaborado pela autora.

Na matriz $E3$ descrita em 5.39 são armazenados os vértices, os pontos aleatórios e o centro da elipse 3 utilizados na verificação da inscrição da elipse no polígono:

$$E3 = \begin{bmatrix} e3A1x & e3B1x & e3C1x & e3D1x & C3_x & PA5_x & PA6_x \\ e3A1y & e3B1y & e3C1y & e3D1y & C3_y & PA5_y & PA6_y \end{bmatrix} \tag{5.39}$$

5.2.4 Elipse posicionada no vértice $D1$ de ε

Para empacotar a elipse tangente ao vértice $D1$ de ε definimos o comprimento do semi-eixo maior pela Equação 5.40:

$$aw4 = 0.6 \cdot \lambda_4, \;\; \lambda_4 \in \mathbb{R}^+ \tag{5.40}$$

onde λ_4 estimado para a elipse tangente ao vértice $D1$, ou seja, quando $t = \frac{3\pi}{2}$, é dado por:

$$\begin{cases} d4_x = |D1x - x_c| \\ d4_y = |D1y - y_c| \end{cases} \tag{5.41}$$

desse modo λ_4 é calculado em função da distância do centro da elipse inicial ao vértice $D1$, obtida pela Equação 5.42:

$$\lambda_4 = \sqrt{(d4_x)^2 + (d4_y)^2} \tag{5.42}$$

estabelecendo que o semi-eixo menor é descrito por:

$$bw4 = aw4 \cdot \sqrt{1 - e^2} \tag{5.43}$$

O centro $C4(C4_x, C4_y)$ da elipse tangente ao vértice $D1$ é obtido pela equação paramétrica 5.44, com $t = \frac{3\pi}{4}$:

$$\begin{cases} C4_x = D1x + aw4 \cdot (\cos(\frac{3\pi}{2}) \cdot \cos(\sigma)) - bw4 \cdot (\sin(\frac{3\pi}{2}) \cdot \sin(\sigma)) \\ C4_y = D1y + aw4 \cdot (\cos(\frac{3\pi}{2}) \cdot \sin(\sigma)) + bw4 \cdot (\sin(\frac{3\pi}{2}) \cdot \cos(\sigma)) \end{cases} \tag{5.44}$$

Os vértices da elipse 4 tangente ao vértice $D1$ de ε e com centro $C4(C4_x, C4_y)$ são definidos por:

Vértice E4A1:

$$\begin{cases} \text{e4A1}x = C4_x - aw4 \cdot c \\ \text{e4A1}y = C4_y + aw4 \cdot s \end{cases}$$

Vértice E4B1:

$$\begin{cases} \text{e4B1}x = C4_x + aw4 \cdot c \\ \text{e4B1}y = C4_y - aw4 \cdot s \end{cases}$$

Vértice E4C1:

$$\begin{cases} \text{e4C1}x = C4_x + bw4 \cdot s \\ \text{e4C1}y = C4_y + bw4 \cdot c \end{cases}$$

Vértice E4D1:

$$\begin{cases} \text{e4D1}x = C4_x - bw4 \cdot s \\ \text{e4D1}y = C4_y - bw4 \cdot c \end{cases}$$

O eixo maior da elipse 4 é definido por $\overrightarrow{E4A1E4B1}$ e o eixo menor é definido por $\overrightarrow{E4C1E4D1}$. Com os vértices, centro e eixos definidos podemos representar graficamente a elipse 4 através da equação paramétrica 5.45:

$$\begin{cases} x4(t) = C4_x + aw4 \cdot (\cos(t) \cdot \cos(\sigma)) - bw4 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y4(t) = C4_y + aw4 \cdot (\cos(t) \cdot \sin(\sigma)) + bw4 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{5.45}$$

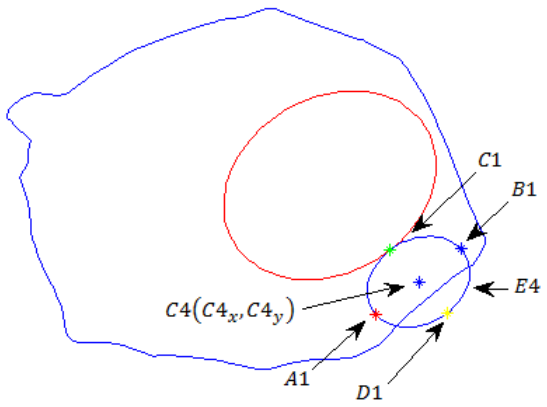


Figura 5.17: Elipse tangente ao vértice D1 de ε. Fonte: Elaborado pela autora.

A Figura 5.18 mostra a inscrição da elipse tangente ao vértice $D1$ de ε, sendo que o 3º e 4º quadrantes são os que possuem pontos exteriores ao contorno. Nesses quadrantes são definidos os pontos aleatórios pelas Equações 5.46 e 5.47:

- **3º quadrante:** $t = \frac{5\pi}{4}$

$$\begin{cases} PA7_x = C4_x + aw4 \cdot (\cos(\frac{5\pi}{4}) \cdot \cos(\sigma)) - bw4 \cdot (\sin(\frac{5\pi}{4}) \cdot \sin(\sigma)) \\ PA7_y = C4_y + aw4 \cdot (\cos(\frac{5\pi}{4}) \cdot \sin(\sigma)) + bw4 \cdot (\sin(\frac{5\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.46}$$

- **4º quadrante:** $t = \frac{7\pi}{4}$

$$\begin{cases} PA8_x = C4_x + aw4 \cdot (\cos(\frac{7\pi}{4}) \cdot \cos(\sigma)) - bw4 \cdot (\sin(\frac{7\pi}{4}) \cdot \sin(\sigma)) \\ PA8_y = C4_y + aw4 \cdot (\cos(\frac{7\pi}{4}) \cdot \sin(\sigma)) + bw4 \cdot (\sin(\frac{7\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{5.47}$$

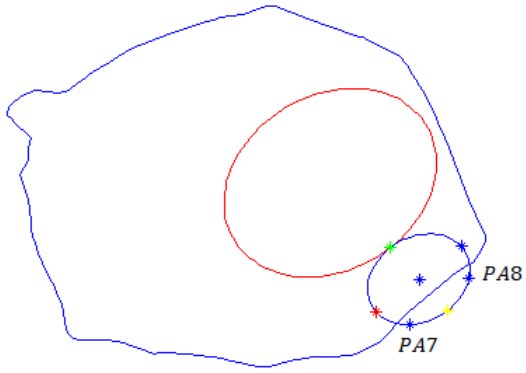


Figura 5.18: Pontos aleatórios no 3º e 4º quadrante. Fonte: Elaborado pela autora.

A matriz $E4$ descrita a seguir armazena os pontos aleatórios $PA7$ e $PA8$ mostrados na Figura 5.18, o centro $C4(C4_x, C4_y)$ e os vértices da elipse 4:

$$E4 = \begin{bmatrix} e4A1x & e4B1x & e4C1x & e4D1x & C4_x & PA7_x & PA8_x \\ e4A1y & e4B1y & e4C1y & e4D1y & C4_y & PA7_y & PA8_y \end{bmatrix} \tag{5.48}$$

Realizado o empacotamento das elipses tangentes aos vértices de ε, as próximas etapas consistem em empacotar as elipses tangentes aos quadrantes Q_i e realizar a verificação da inscrição de todas as elipses.

6 Metodologia analítica para o empacotamento de elipses tangentes aos quadrantes de ε

Este capítulo descreve a metodologia analítica desenvolvida para o empacotamento de elipses tangentes em cada quadrante da elipse inicial com base na ideia adotada em peças Lego. A seguir, são descritas as equações paramétricas resultantes da aplicação de simetrias em $\mathbb{R}^2$ e as respectivas elipses obtidas em cada quadrante da elipse inicial.

Para determinar os pontos de tangência em cada quadrante de ε utilizaremos a equação paramétrica geral. A equação paramétrica 6.1 descreve ε com centro deslocado da origem e rotacionada, em relação ao sistema original de coordenadas cartesianas, e será utilizada para obter as equações paramétricas das demais elipses tangentes.

$$\begin{cases} x(t) = x_c + aw \cdot (\cos(t) \cdot \cos(\sigma)) - bw \cdot (\sin(t) \cdot \sin(\sigma)) \\ y(t) = y_c + aw \cdot (\cos(t) \cdot \sin(\sigma)) + bw \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \qquad t \in [0, 2\pi] \qquad (6.1)$$

O posicionamento das elipses tangentes a ε baseia-se no princípio adotado por [13] em meados de 1958, na construção de peças Lego, e na utilização de simetrias em $\mathbb{R}^2$ [8, 11, 19, 29, 41]. A idéia do Lego é projetar elementos de construção de brinquedos, mais particularmente edifícios formados por tijolos de brinquedo ou blocos adaptados para serem ligados entre si por meio de projeções que se estendem a partir de faces dos elementos e dispostas de modo a envolver porções salientes de um elemento adjacente [13], quando dois elementos são montados conforme a Figura 6.1:

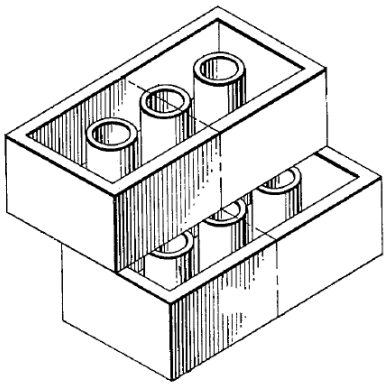


Figura 6.1: Blocos de Lego criados por Godtfred. Fonte: [13]

A idéia utilizada por [13] é que cada circunferência seja inserida de modo que a área máxima ao redor da circunferência central seja ocupada, fixando melhor o bloco a ser encaixado, mantendo um ponto de tangência (contato) em cada quadrante da circunferência central.

O ponto de tangência é resultante da intersecção da bissetriz de cada quadrante com a circunferência, dessa forma tem-se o posicionamento perfeito para as circunferências tangentes, como pode ser visto na Figura 6.2:

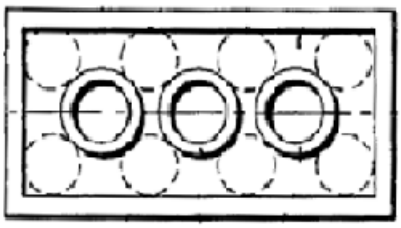


Figura 6.2: Posicionamento das circunferências em cada quadrante. Fonte: [13]

Na Figura 6.3 é possível observar a vista em um plano na direção da face aberta de uma peça qualquer de Lego, sob a forma de um paralelepípedo retangular de construção oca com uma abertura na face e quatro paredes laterais:

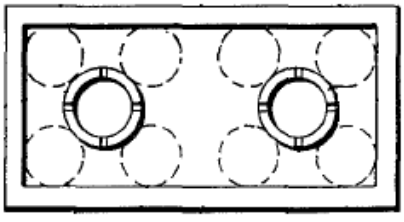


Figura 6.3: Posicionamento das circunferências com duas projeções internas. Fonte: [13]

O princípio do ponto de tangência mostrado na Figura 6.4 será adotado para o posicionamento das elipses tangentes. Notamos que o ponto de tangência no primeiro quadrante está localizado em $\frac{\pi}{4}$:

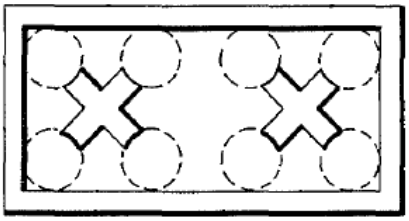


Figura 6.4: Posicionamento das circunferências com cruzamento de diagonais. Fonte: [13].

Adaptamos a Figura 6.3 para definir os pontos de tangência em cada quadrante da circunferência, mostrados na Figura 6.5 e utilizados no empacotamento das elipses tangentes aos quadrantes de ε:

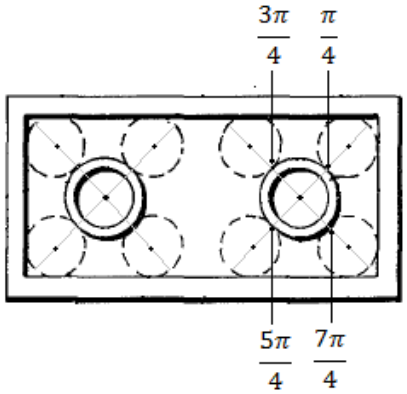


Figura 6.5: Pontos de tangência definidos em cada quadrante. Fonte: Elaborado pela autora.

Com o primeiro ponto de tangência definido, parte-se para a busca dos demais pontos de tangência. Essa busca pode ser realizada de duas maneiras:

1. Utilizando $t \in [0, 2\pi]$ na equação paramétrica 6.1 e assumindo os valores de t em cada quadrante:

   (a) 1º quadrante: $t = \frac{\pi}{4}$

   (b) 2º quadrante: $t = \frac{3\pi}{4}$

   (c) 3º quadrante: $t = \frac{5\pi}{4}$

   (d) 4º quadrante: $t = \frac{7\pi}{4}$

2. Utilizando as simetrias em $\mathbb{R}^2$:

   (a) Reflexão em torno do eixo das ordenadas;

   (b) Reflexão em torno da origem;

   (c) Reflexão em torno do eixo das abscissas.

O primeiro tipo de busca utiliza apenas a equação paramétrica 6.1 com $t \in [0, 2\pi]$ nos casos já determinados para cada quadrante. No desenvolvimento da heurística construtiva para o empacotamento de elipses tangentes aos quadrantes de ε, os pontos de tangência serão determinados utilizando as simetrias em $\mathbb{R}^2$.

6.1 Equações paramétricas resultantes da aplicação de simetrias em $\mathbb{R}^2$

Na heurística construtiva desenvolvida, as simetrias em $\mathbb{R}^2$ são definidas como transformações no plano e utilizadas no cálculo do ponto de tangência, sendo esses criados no sentido anti-horário de ε. As transformações [8, 11, 19, 29, 41] são descritas de acordo com o ponto calculado.

Considerando:

$$A = \begin{bmatrix} aw \cdot \cos(t) \cdot \cos(\sigma) & -bw \cdot \sin(t) \cdot \sin(\sigma) \\ aw \cdot \cos(t) \cdot \sin(\sigma) & bw \cdot \sin(t) \cdot \cos(\sigma) \end{bmatrix} \; t \in [0, 2\pi] \tag{6.2}$$

obtida com base na equação paramétrica 6.1, temos a matriz A definida por 6.2, que será utilizada juntamente com as coordenadas do centro no cálculo dos demais pontos de tangência. As transformações em $\mathbb{R}^2$ [8, 11] são descritas na ordem que aparecem considerando o sentido anti-horário.

- **Reflexão em torno do eixo das ordenadas:**

  A reflexão em torno do eixo das ordenadas é uma transformação linear [41, 45] utilizada para obter o ponto de tangência no 2º quadrante de ε, demonstrada a seguir pela Equação 6.3:

$$T1\,(x, y) = (-x, y) \tag{6.3}$$

onde qualquer ponto com coordenadas (x, y) é transformado através de uma operação de simetria [8, 41] representada pela matriz canônica de transformação 6.4:

$$[T1] = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.4}$$

Para aplicar a transformação $T1(x, y)$ na heurística de empacotamento devemos considerar as coordenadas do centro de ε e a matriz A:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + A \cdot \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.5}$$

resultando na equação paramétrica 6.6:

$$\begin{cases} x(t) = x_c - aw \cdot (\cos(t) \cdot \cos(\sigma)) - bw \cdot (\sin(t) \cdot \sin(\sigma)) & t \in [0, 2\pi] \\ y(t) = y_c - aw \cdot (\cos(t) \cdot \sin(\sigma)) + bw \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.6}$$

onde aw e bw são os semi-eixos de ε. A reflexão em torno do eixo das ordenadas no sistema cartesiano é mostrada na Figura 6.6:

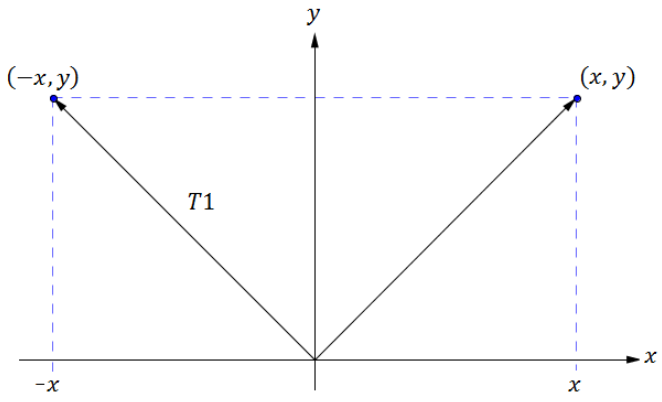


Figura 6.6: Reflexão em torno do eixo das ordenadas. Fonte: [45].

- **Reflexão em torno da origem:**

A reflexão em torno da origem [8, 45] dará origem ao ponto de tangência posicionado no 3º quadrante de ε, descrita pela Equação 6.7:

$$T2\,(x, y) = (-x, -y) \tag{6.7}$$

definida pela matriz canônica 6.8:

$$[T2] = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.8}$$

aplicando a transformação linear $T2(x, y)$ na equação paramétrica geral obtemos a Equação 6.10:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + A \cdot \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.9}$$

$$\begin{cases} x(t) = x_c - aw \cdot (\cos(t) \cdot \cos(\sigma)) + bw \cdot (\sin(t) \cdot \sin(\sigma)) \quad t \in [0, 2\pi] \\ y(t) = y_c - aw \cdot (\cos(t) \cdot \sin(\sigma)) - bw \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.10}$$

Na Figura 6.7 podemos observar a reflexão em torno da origem no sistema cartesiano original:

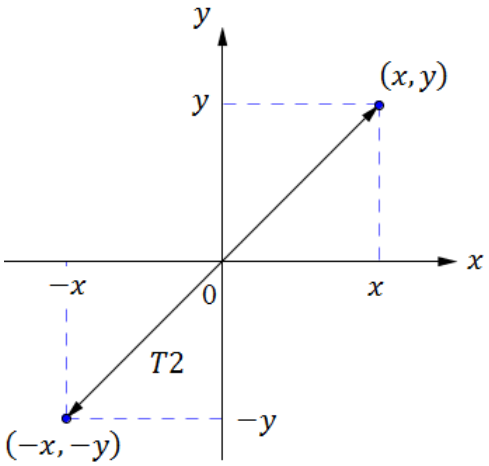


Figura 6.7: Reflexão em torno da origem. Fonte: [45].

- **Reflexão em torno do eixo das abscissas:**

  Essa transformação linear leva cada ponto (x, y) para sua imagem $(x, -y)$ simétrica em relação ao eixo das abscissas [45] e será utilizada para determinar o ponto de tangência no 4º quadrante de ε:

$$T3\,(x, y) = (x, -y) \tag{6.11}$$

sendo representada pela matriz canônica 6.12:

$$[T3] = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.12}$$

com a matriz canônica definida, utilizaremos as coordenadas do centro de ε e a matriz A para obter a equação paramétrica:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \end{bmatrix} + A \cdot \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.13}$$

resultando na equação paramétrica 6.14:

$$\begin{cases} x(t) = x_c + aw \cdot (\cos(t) \cdot \cos(\sigma)) + bw \cdot (\sin(t) \cdot \sin(\sigma)) & t \in [0, 2\pi] \\ y(t) = y_c + aw \cdot (\cos(t) \cdot \sin(\sigma)) - bw \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.14}$$

A transformação linear realizada é mostrada na Figura 6.8:

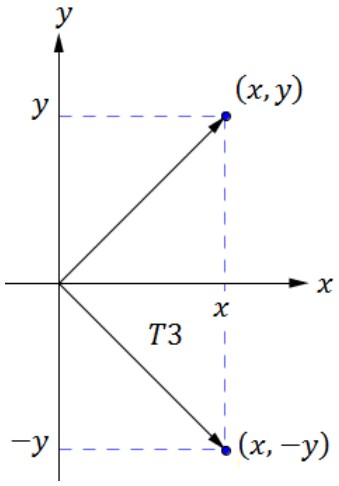


Figura 6.8: Reflexão em torno do eixo das abscissas. Fonte: [8].

Na Figura 6.9 são representadas as simetrias em $\mathbb{R}^2$ que serão utilizadas nessa heurística construtiva:

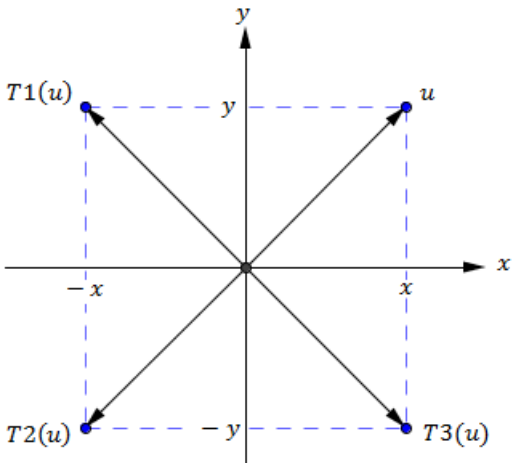


Figura 6.9: Simetrias em $\mathbb{R}^2$. Fonte: Elaborado pela autora.

Essas transformações geram os pontos de tangência em cada quadrante em função do ponto calculado no 1º quadrante. Na Figura 6.10 temos o ponto $P5(P5_x, P5_y)$, que é o primeiro a ser calculado, e através da equação paramétrica obtemos os demais pontos $P6, P7$ e $P8$.

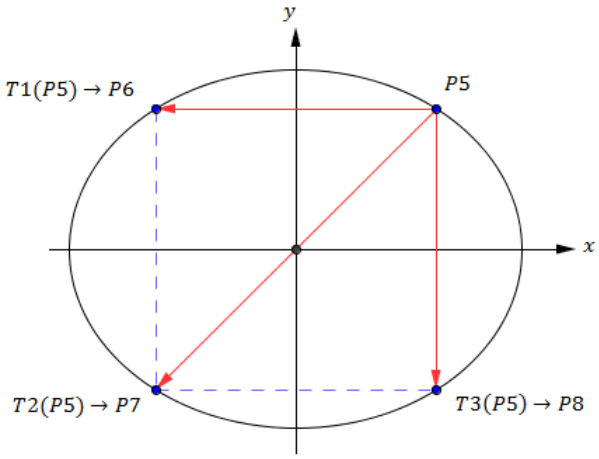


Figura 6.10: Transformações aplicadas para obtenção de pontos de tangência. Fonte: Elaborado pela autora.

6.1.1 Elipse posicionada no 1º quadrante de ε

O primeiro ponto de tangência calculado localiza-se no 1º quadrante, assumindo que será posicionada apenas uma elipse em cada quadrante de ε. Considera-se que o ponto de tangência deve coincidir com a intersecção da bissetriz do quadrante ($t = \frac{\pi}{4}$) com a curva, estabelecendo a mesma rotação de ε. Assumindo que a rotação das elipses

é a igual a rotação de ε, verifica-se que estas possuem os eixos maior e menor paralelos aos eixos de ε e das elipses tangentes aos vértices.

Como o ponto é dado em função da bissetriz do quadrante, ou seja, quando $t = \frac{\pi}{4}$ com um ângulo de rotação σ e coordenadas do centro $C(x_c, y_c)$ de ε, utilizaremos a equação paramétrica 6.15 para determinar o ponto de tangência da 5ª elipse a ser empacotada pela heurística:

$$\begin{cases} P5_x = x_c + aw \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ P5_y = y_c + aw \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.15}$$

onde aw e bw são respectivamente, o semi-eixo maior e menor de ε e σ é obtido como parâmetro após a inscrição de ε.

As coordenadas desse ponto são armazenadas e utilizadas no cálculo da elipse tangente. Seguindo a heurística criada para o posicionamento do centro das elipses tangentes aos vértices de ε, estima-se o comprimento do semi-eixo maior da elipse a ser empacotada. O centro será calculado levando em consideração a mesma direção de busca do ponto de tangência, conforme a Figura 6.11 :

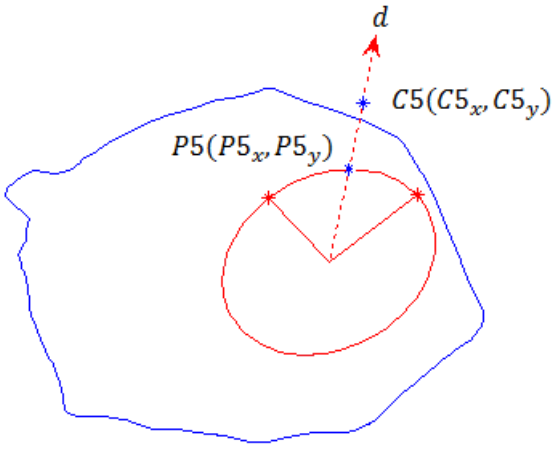


Figura 6.11: Ponto de tangência com direção de busca definida no 1º quadrante. Fonte: Elaborado pela autora.

O comprimento do semi-eixo maior de cada elipse tangente aos quadrantes de ε é dado pela Equação 6.16:

$$aw = 0.8 \cdot \lambda, \quad \lambda \in \mathbb{R}^+ \tag{6.16}$$

Assume-se apenas 80% da distância do centro de ε ao ponto de tangência, evitando dessa forma, que as elipses sejam secantes entre si e haja a sobreposição. O comprimento do semi-eixo maior da elipse posicionada no 1º quadrante de ε é calculado utilizando um escalar λ conforme a Equação 6.17:

$$aw5 = 0.8 \cdot \lambda_5, \quad \lambda_5 \in \mathbb{R}^+ \tag{6.17}$$

O valor desse escalar λ_5 para a elipse tangente ao ponto $P5(P5_x, P5_y)$ é definido por:

$$\begin{cases} d5_x = |P5_x - x_c| \\ d5_y = |P5_y - y_c| \end{cases} \tag{6.18}$$

onde λ_5 determina a distância do centro de ε ao ponto $P5(P5_x, P5_y)$, conforme a Equação 6.19:

$$\lambda_5 = \sqrt{(d5_x)^2 + (d5_y)^2} \tag{6.19}$$

e o semi-eixo menor da elipse 5 é calculado por:

$$bw5 = aw5 \cdot \sqrt{1 - e^2} \tag{6.20}$$

O centro $C5(C5_x, C5_y)$ da elipse tangente ao ponto $P5(P5_x, P5_y)$ é obtido pela equação paramétrica 6.21, assumindo $\frac{\pi}{4}$ e ângulo de rotação σ:

$$\begin{cases} C5_x = P5_x + aw5 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw5 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ C5_y = P5_y + aw5 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw5 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.21}$$

Com as coordenadas do centro definidas, determinam-se os vértices da elipse tangente ao 1º quadrante de ε, com $t = \frac{\pi}{4}$:

Vértice E5A1:

$$\begin{cases} \text{e5A1}x = C5_x - aw5 \cdot c \\ \text{e5A1}y = C5_y + aw5 \cdot s \end{cases}$$

Vértice E5B1:

$$\begin{cases} \text{e5B1}x = C5_x + aw5 \cdot c \\ \text{e5B1}y = C5_y - aw5 \cdot s \end{cases}$$

Vértice E5C1:

$$\begin{cases} \text{e5C1}x = C5_x + bw5 \cdot s \\ \text{e5C1}y = C5_y + bw5 \cdot c \end{cases}$$

Vértice E5D1:

$$\begin{cases} \text{e5D1}x = C5_x - bw5 \cdot s \\ \text{e5D1}y = C5_y - bw5 \cdot c \end{cases}$$

O comprimento do eixo maior da elipse é definido por $\overrightarrow{E5A1E5B1}$ e o do eixo menor dessa elipse tangente ao 1º quadrante de ε é definido por $\overrightarrow{E5C1E5D1}$ e podemos representar a elipse graficamente utilizando a equação paramétrica 6.22:

$$\begin{cases} x5(t) = C5_x + aw5 \cdot (\cos(t) \cdot \cos(\sigma)) - bw5 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y5(t) = C5_y + aw5 \cdot (\cos(t) \cdot \sin(\sigma)) + bw5 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.22}$$

Para garantir a inscrição dessa elipse no polígono, calcula-se um ponto aleatório no quadrante que mantém maior contato com o polígono e que pela heurística construtiva é indicado pela direção de busca, nesse caso o 1º quadrante. Definimos o ponto aleatório como a bissetriz do quadrante, $BA5(BA5_x, BA5_y)$. A designação de $BA5$ faz referência a bissetriz aleatória calculada na elipse 5 através da equação paramétrica 6.23:

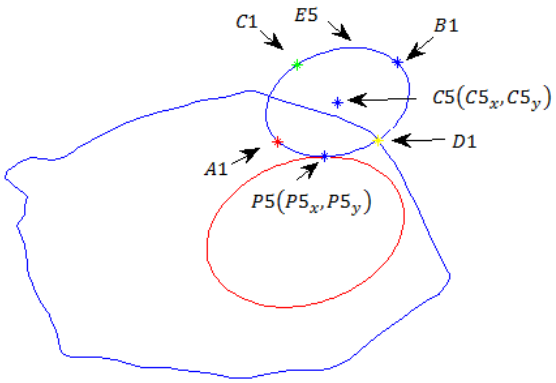


Figura 6.12: Elipse tangente ao 1º quadrante de ε. Fonte: Elaborado pela autora.

$$\begin{cases} BA5_x = P5_x + 2 \cdot aw5 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - 2 \cdot bw5 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ BA5_y = P5_y + 2 \cdot aw5 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + 2 \cdot bw5 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.23}$$

O posicionamento do ponto $BA5(BA5_x, BA5_y)$ leva em consideração o ponto de tangência e o centro da elipse tangente, ou seja, ambos estão posicionados na mesma direção de busca d da bissetriz do 1º quadrante e são equidistantes, conforme mostra a Figura 6.13:

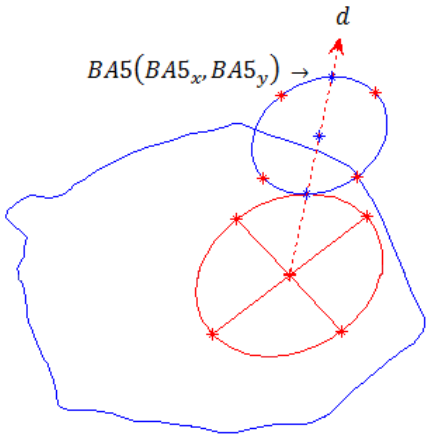


Figura 6.13: Elipse 5 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora.

Os principais pontos da elipse tangente ao 1º quadrante de ε são armazenados em uma matriz que será utilizada para a verificação da inscrição em P. A matriz E5 armazena os vértices, o ponto aleatório e o centro da elipse para verificar a inscrição:

$$E5 = \begin{bmatrix} e5A1x & e5B1x & e5C1x & e5D1x & P5_x & C5_x & BA5_x \\ e5A1y & e5B1y & e5C1y & e5D1y & P5_y & C5_y & BA5_y \end{bmatrix} \tag{6.24}$$

6.1.2 Elipse posicionada no 2º quadrante de ε

O empacotamento da elipse tangente ao 2º quadrante de ε tem como base o ponto de tangência obtido pela transformação em $\mathbb{R}^2$ [8, 11, 41] - reflexão em torno do eixo das ordenadas - aplicada ao ponto $P5(P5_x, P5_y)$ resultando na equação paramétrica utilizada para calcular as coordenadas do ponto $P6(P6_x, P6_y)$, conforme mostra a Figura 6.14:

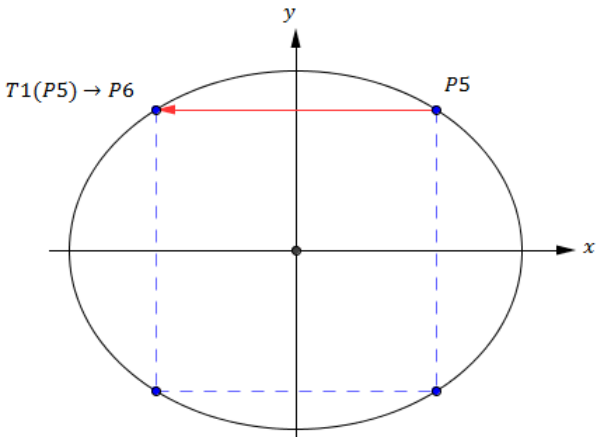


Figura 6.14: Ponto de tangência $P6$ resultado da transformação aplicada. Fonte: Elaborado pela autora.

As coordenadas do ponto $P6(P6_x, P6_y)$ são calculadas com a equação paramétrica 6.25, utilizando as coordenadas do centro de ε e $t = \frac{\pi}{4}$:

$$\begin{cases} P6_x = x_c - aw \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ P6_y = y_c - aw \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.25}$$

Determina-se dessa forma o ponto de tangência localizado no 2º quadrante e a direção de busca do centro $C6(C6_x, C6_y)$ mostrado na Figura 6.15. O comprimento do semi-eixo maior da elipse posicionada no 2º quadrante de ε é obtido pela Equação 6.26:

$$aw6 = 0.8 \cdot \lambda_6, \quad \lambda_6 \in \mathbb{R}^+ \tag{6.26}$$

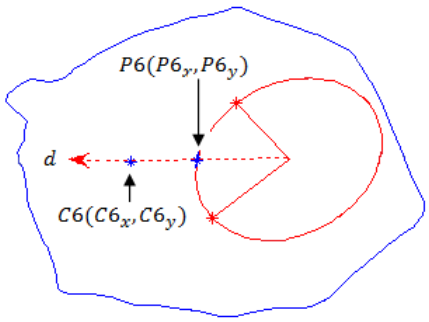


Figura 6.15: Ponto de tangência com direção de busca definida no 2º quadrante. Fonte: Elaborado pela autora.

A distância do centro $C(x_c, y_c)$ de ε ao ponto $P6(P6_x, P6_y)$ define o valor de λ_6 através da Equação 6.27:

$$\begin{cases} d6_x = |P6_x - x_c| \\ d6_y = |P6_y - y_c| \end{cases} \tag{6.27}$$

onde

$$\lambda_6 = \sqrt{(d6_x)^2 + (d6_y)^2} \tag{6.28}$$

e o semi-eixo menor da elipse, obtido pela Equação 6.29:

$$bw6 = aw6 \cdot \sqrt{1 - e^2} \tag{6.29}$$

O centro $C6(C6_x, C6_y)$ da elipse posicionada no 2º quadrante de ε é calculado assumindo $t = \frac{\pi}{4}$ e ângulo de rotação σ como parâmetros na equação paramétrica 6.30 resultante da simetria em $\mathbb{R}^2$ aplicada no ponto $P5(P5_x, P5_y)$:

$$\begin{cases} C6_x = P6_x - aw6 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - bw6 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ C6_y = P6_y - aw6 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + bw6 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.30}$$

Os vértices da elipse tangente ao 2º quadrante de ε com $t = \frac{\pi}{4}$ são calculados com base nas coordenadas do centro e ângulo de rotação:

Vértice E6A1:

$$\begin{cases} \text{e6A1}x = C6_x - aw6 \cdot c \\ \text{e6A1}y = C6_y + aw6 \cdot s \end{cases}$$

Vértice E6B1:

$$\begin{cases} \text{e6B1}x = C6_x + aw6 \cdot c \\ \text{e6B1}y = C6_y - aw6 \cdot s \end{cases}$$

Vértice E6C1:

$$\begin{cases} \text{e6C1}x = C6_x + bw6 \cdot s \\ \text{e6C1}y = C6_y + bw6 \cdot c \end{cases}$$

Vértice E6D1:

$$\begin{cases} \text{e6D1}x = C6_x - bw6 \cdot s \\ \text{e6D1}y = C6_y - bw6 \cdot c \end{cases}$$

Com os vértices calculados, temos que o eixo maior da elipse é definido por $\overrightarrow{E6A1E6B1}$ e o eixo menor dessa elipse é definido por $\overrightarrow{E6C1E6D1}$. A representação geométrica da elipse tangente ao 2º quadrante de ε é realizada com base na equação paramétrica 6.31:

$$\begin{cases} x6(t) = C6_x - aw6 \cdot (\cos(t) \cdot \cos(\sigma)) - bw6 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y6(t) = C6_y - aw6 \cdot (\cos(t) \cdot \sin(\sigma)) + bw6 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.31}$$

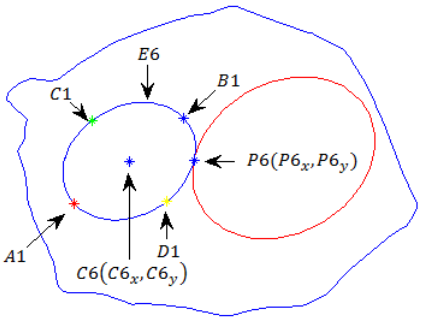


Figura 6.16: Elipse tangente ao 2º quadrante de ε. Fonte: Elaborado pela autora.

O ponto aleatório da elipse tangente ao 2º quadrante de ε é projetado no 2º quadrante, onde ocorre o maior contato com polígono. O ponto $BA6(BA6_x, BA6_y)$ está posicionado na mesma direção de busca (bissetriz do quadrante) utilizada no posicionamento do centro de ε, do ponto de tangência $P6(P6_x, P6_y)$ e do centro $C6(C6_x, C6_y)$ da elipse tangente:

$$\begin{cases} BA6_x = P6_x - 2 \cdot aw6 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) - 2 \cdot bw6 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ BA6_y = P6_y - 2 \cdot aw6 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) + 2 \cdot bw6 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.32}$$

temos dessa forma o posicionamento do ponto aleatório utilizado para verificação da inscrição, mostrado na Figura 6.17:

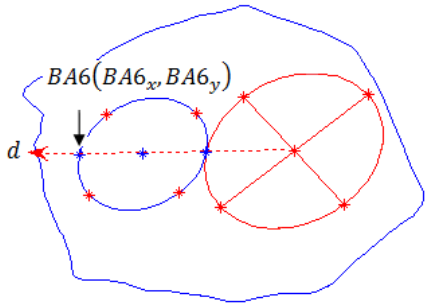


Figura 6.17: Elipse 6 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora.

A matriz utilizada para a verificação do empacotamento da elipse 6, no polígono P é descrita por:

$$E6 = \begin{bmatrix} e6A1x & e6B1x & e6C1x & e6D1x & P6_x & C6_x & BA6_x \\ e6A1y & e6B1y & e6C1y & e6D1y & P6_y & C6_y & BA6_y \end{bmatrix} \tag{6.33}$$

6.1.3 Elipse posicionada no 3º quadrante de ε

Para empacotar a elipse tangente ao 3º quadrante de ε é necessário definir o ponto de tangência que será a base para obter o centro e os vértices dessa elipse. O ponto de tangência é resultado de uma transformação em $\mathbb{R}^2$ - reflexão em torno da origem - aplicada ao ponto $P5(P5_x, P5_y)$, cuja equação paramétrica será utilizada para obter o ponto $P7(P7_x, P7_y)$:

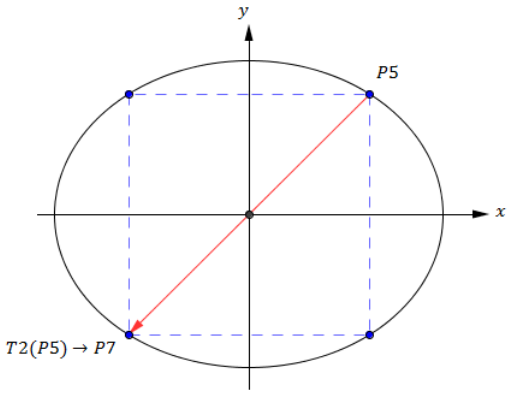


Figura 6.18: Ponto de tangência P7 obtido com a transformação aplicada. Fonte: Elaborado pela autora.

Temos que $P7(P7_x, P7_y)$ é descrito pela Equação 6.34, utilizando as coordenadas do centro e os semi-eixos de ε:

$$\begin{cases} P7_x = x_c - aw \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + bw \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ P7_y = y_c - aw \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - bw \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.34}$$

e assumindo a direção de busca d definida pela bissetriz do quadrante para o posicionamento do centro $C7(C7_x, C7_y)$ e do ponto aleatório $BA7(BA7_x, BA7_y)$ da elipse 7. A direção de busca, ponto de tangência e o centro são mostrados na Figura 6.19.

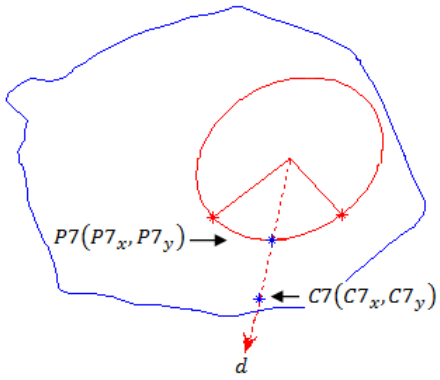


Figura 6.19: Ponto de tangência com direção de busca definida no 3º quadrante. Fonte: Elaborado pela autora.

Com o ponto de tangência calculado, o próximo passo é determinar o comprimento do semi-eixo maior da elipse 7 é definido pela Equação 6.35:

$$aw7 = 0.8 \cdot \lambda_7, \quad \lambda_7 \in \mathbb{R}^+ \tag{6.35}$$

sendo que λ_7 é calculado através da distância entre o centro $C(x_c, y_c)$ de ε e o ponto $P7(P7_x, P7_y)$:

$$\begin{cases} d7_x = |P7_x - x_c| \\ d7_y = |P7_y - y_c| \end{cases} \tag{6.36}$$

onde:

$$\lambda_7 = \sqrt{(d7_x)^2 + (d7_y)^2} \tag{6.37}$$

e o semi-eixo menor da elipse é obtido com a Equação 6.38:

$$bw7 = aw7 \cdot \sqrt{1 - e^2} \tag{6.38}$$

Para o cálculo do posicionamento do centro da elipse 7, ou seja, $C7(C7_x, C7_y)$ são utilizadas as informações do ponto de tangência, comprimento dos semi-eixos, ângulo de rotação e assumindo $\frac{\pi}{4}$:

$$\begin{cases} C7_x = P7_x - aw7 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + bw7 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ C7_y = P7_y - aw7 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - bw7 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.39}$$

Definindo os vértices da elipse tangente ao 3º quadrante de ε:

Vértice E7A1:

$$\begin{cases} \text{e7A1}x = C7_x - aw7 \cdot c \\ \text{e7A1}y = C7_y + aw7 \cdot s \end{cases}$$

Vértice E7B1:

$$\begin{cases} \text{e7B1}x = C7_x + aw7 \cdot c \\ \text{e7B1}y = C7_y - aw7 \cdot s \end{cases}$$

Vértice E7C1:

$$\begin{cases} \text{e7C1}x = C7_x + bw7 \cdot s \\ \text{e7C1}y = C7_y + bw7 \cdot c \end{cases}$$

Vértice E7D1:

$$\begin{cases} \text{e7D1}x = C7_x - bw7 \cdot s \\ \text{e7D1}y = C7_y - bw7 \cdot c \end{cases}$$

O eixo maior da elipse é definido por $\overrightarrow{E7A1E7B1}$ e o eixo menor dessa elipse é definido por $\overrightarrow{E7C1E7D1}$. Com a informação dos semi-eixos e centro podemos representá-la graficamente pela equação paramétrica 6.40:

$$\begin{cases} x7(t) = C7_x - aw7 \cdot (\cos(t) \cdot \cos(\sigma)) + bw7 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y7(t) = C7_y - aw7 \cdot (\cos(t) \cdot \sin(\sigma)) - bw7 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.40}$$

Para garantir o empacotamento da elipse 7 em P calcula-se o ponto aleatório $BA7(BA7_x, BA7_y)$ no 3º quadrante dessa elipse, posicionado na mesma direção utilizada no cálculo do ponto de tangência $P7(P7_x, P7_y)$, do centro $C(x_c, y_c)$ de ε e do centro da elipse tangente $C7(C7_x, C7_y)$:

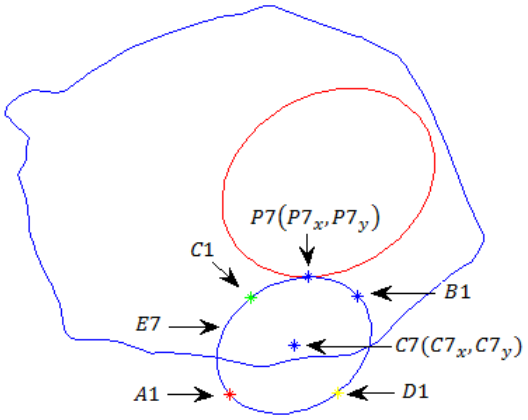


Figura 6.20: Elipse tangente ao 3º quadrante de ε. Fonte: Elaborado pela autora.

$$\begin{cases} BA7_x = P7_x - 2 \cdot aw7 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + 2 \cdot bw7 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ BA7_y = P7_y - 2 \cdot aw7 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - 2 \cdot bw7 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.41}$$

que pode ser visualizado na Figura 6.21:

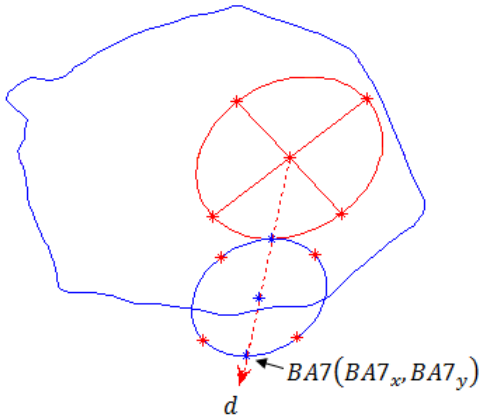


Figura 6.21: Elipse 7 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora.

A matriz utilizada para a verificação da inscrição de elipse 7 no polígono P é descrita por:

$$E7 = \begin{bmatrix} e7A1x & e7B1x & e7C1x & e7D1x & P7_x & C7_x & BA7_x \\ e7A1y & e7B1y & e7C1y & e7D1y & P7_y & C7_y & BA7_y \end{bmatrix} \tag{6.42}$$

6.1.4 Elipse posicionada no 4º quadrante de ε

A transformação em $\mathbb{R}^2$ - reflexão em torno do eixo das abscissas [8, 41]- mostrada na Figura 6.22, quando aplicada ao ponto $P5(P5_x, P5_y)$, resulta no ponto $P8(P8_x, P8_y)$. Com esse ponto, determinado no 4º quadrante de ε, realiza-se o empacotamento da elipse tangente.

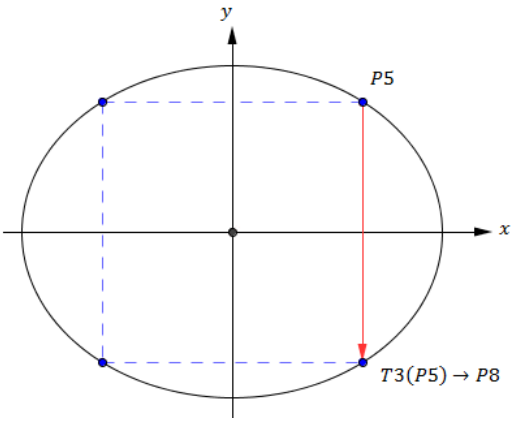


Figura 6.22: Ponto de tangência P8 resultado da transformação aplicada. Fonte: Elaborado pela autora.

O ponto de tangência $P8$ que coincide com a intersecção da bissetriz do 4º quadrante de ε é obtido através da equação paramétrica 6.43, com $t = \frac{\pi}{4}$.

$$\begin{cases} P8_x = x_c + aw \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + bw \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ P8_y = y_c + aw \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - bw \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.43}$$

O ponto $P8(P8_x, P8_y)$ determinará a direção de busca d para o posicionamento do centro $C8(C8_x, C8_y)$ e do ponto aleatório, conforme a Figura 6.23:

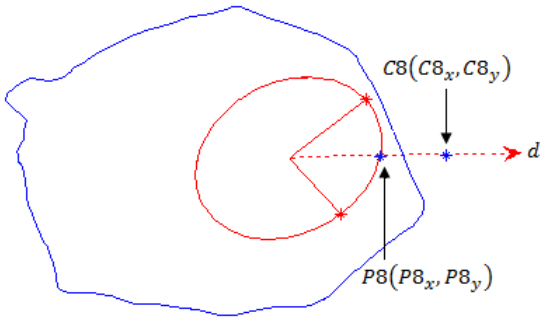


Figura 6.23: Ponto de tangência com direção de busca definida no 4º quadrante. Fonte: Elaborado pela autora.

O comprimento do semi-eixo maior da elipse posicionada no 4º quadrante de ε é calculado utilizando um escalar λ_8 dado pela Equação 6.44:

$$aw8 = 0.8 \cdot \lambda_8, \;\; \lambda_8 \in \mathbb{R}^+ \tag{6.44}$$

O valor desse escalar λ_8 para a elipse tangente ao ponto $P8(P8_x, P8_y)$ é definido com a Equação 6.45:

$$\begin{cases} d8_x = |P8_x - x_c| \\ d8_y = |P8_y - y_c| \end{cases} \tag{6.45}$$

de modo que

$$\lambda_8 = \sqrt{(d8_x)^2 + (d8_y)^2} \tag{6.46}$$

e o semi-eixo menor da elipse é calculado por:

$$bw8 = aw8 \cdot \sqrt{1 - e^2} \tag{6.47}$$

Temos que o centro $C8(C8_x, C8_y)$ da elipse tangente ao ponto $P8(P8_x, P8_y)$ é determinado pela equação paramétrica 6.48, assumindo $\frac{\pi}{4}$ e ângulo de rotação σ:

$$\begin{cases} C8_x = P8_x + aw8 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + bw8 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ C8_y = P8_y + aw8 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - bw8 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.48}$$

Definindo os vértices da elipse tangente ao 4º quadrante de ε com $t = \frac{\pi}{4}$:

Vértice E8A1:

$$\begin{cases} \text{e8A1}x = C8_x - aw8 \cdot c \\ \text{e8A1}y = C8_y + aw8 \cdot s \end{cases}$$

Vértice E8B1:

$$\begin{cases} \text{e8B1}x = C8_x + aw8 \cdot c \\ \text{e8B1}y = C8_y - aw8 \cdot s \end{cases}$$

Vértice E8C1:

$$\begin{cases} \text{e8C1}x = C8_x + bw8 \cdot s \\ \text{e8C1}y = C8_y + bw8 \cdot c \end{cases}$$

Vértice E8D1:

$$\begin{cases} \text{e8D1}x = C8_x - bw8 \cdot s \\ \text{e8D1}y = C8_y - bw8 \cdot c \end{cases}$$

Com a equação paramétrica 6.49 é possível representar geometricamente a elipse tangente ao 4º quadrante de ε. O eixo maior dessa elipse é definido por $\overrightarrow{E8A1E8B1}$ e o eixo menor por $\overrightarrow{E8C1E8D1}$.

$$\begin{cases} x8(t) = C8_x + aw8 \cdot (\cos(t) \cdot \cos(\sigma)) + bw8 \cdot (\sin(t) \cdot \sin(\sigma)), & t \in [0, 2\pi] \\ y8(t) = C8_y + aw8 \cdot (\cos(t) \cdot \sin(\sigma)) - bw8 \cdot (\sin(t) \cdot \cos(\sigma)) \end{cases} \tag{6.49}$$

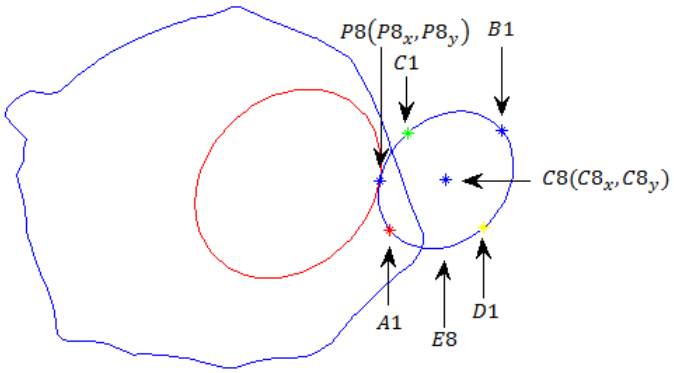


Figura 6.24: Elipse tangente ao 4º quadrante de ε. Fonte: Elaborado pela autora.

O ponto aleatório deve ser calculado no quadrante que possui pontos que não sejam interiores ao polígono, nesse caso o 4º quadrante. A representação da elipse na

Figura 6.24 mostra que não apenas o 4º quadrante possui pontos exteriores ao polígono. Porém, pela heurística construtiva, basta definir o ponto aleatório na direção de busca assumida. Garantindo que esse ponto é interior, pelas características da heurística, os demais pontos também pertencem ao polígono.

De modo que, os vértices e o centro também são utilizados para a verificação da inscrição. Definimos o ponto aleatório como $BA8(BA8_x, BA8_y)$ pela Equação 6.50 e mostrado na Figura 6.25, assumindo $t = \frac{\pi}{4}$:

$$\begin{cases} BA8_x = P8_x + 2 \cdot aw8 \cdot (\cos(\frac{\pi}{4}) \cdot \cos(\sigma)) + 2 \cdot bw8 \cdot (\sin(\frac{\pi}{4}) \cdot \sin(\sigma)) \\ BA8_y = P8_y + 2 \cdot aw8 \cdot (\cos(\frac{\pi}{4}) \cdot \sin(\sigma)) - 2 \cdot bw8 \cdot (\sin(\frac{\pi}{4}) \cdot \cos(\sigma)) \end{cases} \tag{6.50}$$

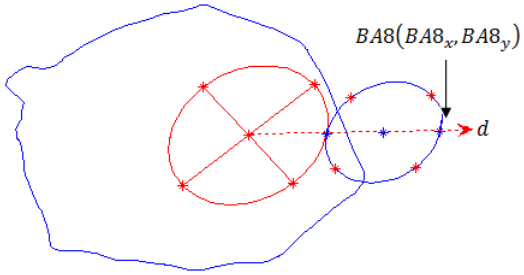


Figura 6.25: Elipse 8 - Ponto de tangência, centro e ponto aleatório. Fonte: Elaborado pela autora.

O ponto de tangência, centro e ponto aleatório estão posicionados na direção de busca d. Com os pontos e os vértices armazenados na matriz E8, realiza-se a verificação do empacotamento da elipse em P:

$$E8 = \begin{bmatrix} e8A1x & e8B1x & e8C1x & e8D1x & P8_x & C8_x & BA8_x \\ e8A1y & e8B1y & e8C1y & e8D1y & P8_y & C8_y & BA8_y \end{bmatrix} \tag{6.51}$$

Com o empacotamento das elipses nos vértices e quadrantes de ε, o próximo passo da heurística construtiva desenvolvida é verificar a inscrição de todas as elipses, caso existam elipses cujos pontos são exteriores ao contorno que define o polígono P, realiza-se o procedimento de redução gradativa do semi-eixo maior até que a devida elipse esteja empacotada em P. O processo de verificação da inscrição será descrito no Capítulo 7.

7 Verificação da inscrição das elipses no polígono P

Este capítulo trata do procedimento de verificação da inscrição das elipses no polígono. Primeiramente, situam-se os casos que podem ocorrer com o empacotamento: pontos interiores, exteriores ou coincidindo com os vértices ou arestas. Após, descreve-se a função inpolygon do Matlab utilizada para a verificação dos pontos em relação ao polígono. Por fim, é mostrada a redução gradativa, adotada na heurística construtiva, para redefinir o comprimento do semi-eixo maior da elipse.

A heurística construtiva desenvolvida realiza o posicionamento da elipse inicial pelo método de Hooke e Jeeves [4] buscando o contorno do polígono na direção assumida, por isso existe o contato de alguns quadrantes. Como pode ser visto na Figura 7.1, algumas elipses não estão totalmente inscritas no polígono P. Dessa forma, faz-se necessário definir um ponto em cada quadrante que tem probabilidade de conter pontos exteriores ao polígono. Esse ponto aleatório é calculado através da intersecção da bissetriz do quadrante com a curva que descreve a elipse e nas elipses tangentes aos quadrantes, esse ponto sempre estará posicionado na mesma direção de busca do ponto de tangência e do centro.

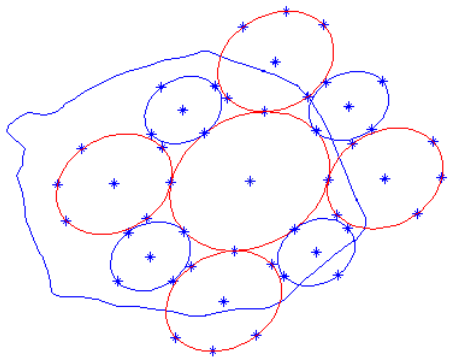


Figura 7.1: Heurística construtiva: elipses posicionadas nos vértices e quadrantes de ε sem a verificação da inscrição. Fonte: Elaborado pela autora.

As elipses que possuam pontos $P(x, y)$ exteriores ao contorno, tem esses pontos armazenados em uma matriz e os pontos interiores em outra matriz, depois do processo de verificação. Na metodologia analítica apresentada nos capítulos anteriores definia-se uma matriz com as coordenadas dos vértices, centro, ponto de tangência e pontos aleatórios. A matriz formada é utilizada no teste para a verificação desses pontos da elipse em relação ao polígono, sendo que os pontos podem estar contidos ou não no polígono ou ainda coincidir com um vértice. O posicionamento dos pontos [40] quando adaptado a heurística resulta nos casos mostrados na Figura 7.2:

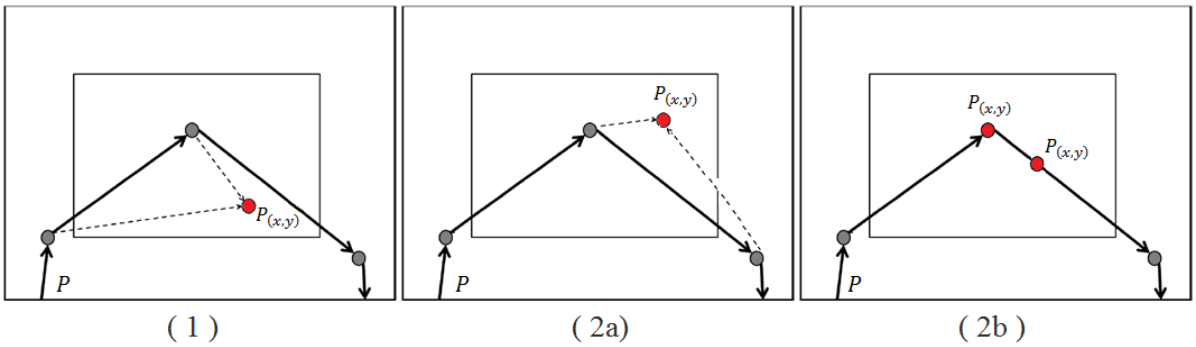


Figura 7.2: Posição relativa do ponto em relação ao polígono P. Fonte: Adaptado pela autora [40].

Com base na Figura 7.2, temos as situações (1) e (2):

- 1) $P_{(x,y)}(in) \mapsto P_{(x,y)} \in P$

Nesse caso os pontos da elipse são interiores, ou seja, a elipse está totalmente inscrita no polígono P e não existe a necessidade de realizar a redução gradativa do semi-eixo maior.

- 2) $P_{(x,y)}(\sim in) \mapsto P_{(x,y)} \notin P$

2a) Os pontos $P_{(x,y)}(\sim in)$ são exteriores ao polígono. Caso o ponto esteja situado na fronteira da região poligonal ou seja um vértice do polígono (2b), faz-se uma nova verificação da posição relativa, que resulta em duas situações: ou está na fronteira ou está posicionado dentro ou fora de P.

Quando a matriz de pontos exteriores não for vazia, aplica-se a redução gradativa do semi-eixo maior. Para realizar a verificação dos pontos utilizamos a função

inpoligon disponível na biblioteca de implementações do MatLab [34] com adaptações realizadas para garantir a eficiência da heurística. A função *inpoligon* possui uma sintaxe simples, descrita pelo MatLab:

Sintaxe:

$IN = inpolygon(X, Y, xv, yv)$

$[\ IN \quad ON\ ] = inpolygon\ (X, Y, xv, yv)$

onde (X, Y) são as coordenadas dos pontos que compõe a matriz de cada elipse e $xv, yv)$ as coordenadas dos vértices do polígono P.

Descrição:

$IN = inpolygon(X, Y, xv, yv)$ retorna uma matriz IN do mesmo tamanho de X e Y. Para cada elemento de IN é atribuído o valor de 0 ou 1 (verdadeiro ou falso) dependendo se o ponto $(X(p, q), Y(p, q))$ está dentro da região poligonal definida por vx, cujos vértices são especificados pelos vetores xv e yv. A atribuição do valor 0 ou 1 ocorre da seguinte maneira:

- Se $(X(p, q), Y(p, q))$ está dentro da região poligonal ou no limite do polígono implica em $IN(p, q) = 1$;
- Se $(X(p, q), Y(p, q))$ está fora da região poligonal implica em $IN(p, q) = 0$.

$[\ IN \quad ON\ ] = inpolygon\ (X, Y, xv, yv)$ retorna uma segunda matriz ON do mesmo tamanho de X e Y. Para cada elemento de ON é atribuído o valor 0 ou 1, avaliando se o ponto $(X(p, q), Y(p, q))$ está situado na fronteira da região poligonal, resultando em duas situações:

- Se $(X(p, q), Y(p, q))$ está na fronteira da região poligonal implica em $ON(p, q) = 1$;
- Se $(X(p, q), Y(p, q))$ está dentro ou fora da região poligonal implica em $ON(p, q) = 0$.

Na heurística proposta todas as elipses devem estar totalmente inscritas na região poligonal, ou seja, a matriz $[\ IN \quad ON\ ]$ de cada elipse dever conter apenas pontos interiores ao polígono. Retornando algum ponto exterior ao polígono, faz-se a redução gradativa do semi-eixo maior e realiza-se novamente o cálculo do semi-eixo menor, do

centro, dos vértices e dos pontos aleatórios até que todos os pontos sejam interiores ao polígono.

A estratégia adotada consiste em separar a matriz vx dos vértices em duas sub-matrizes: xv e yv, conforme mostrado em 7.1 e 7.2:

$$xv = \Big[xv; xv[1]\Big] \tag{7.1}$$

$$yv = \Big[yv; yv[1]\Big] \tag{7.2}$$

Da mesma forma, separamos a matriz dos pontos da elipse empacotada em duas sub-matrizes: x e y.

$$x = \Big[x; x[1]\Big] \tag{7.3}$$

$$y = \Big[y; y[1]\Big] \tag{7.4}$$

com $IN = inpolygon\ (x, y, xv, yv)$ e definindo $A = x(in)$ e $B = y(in)$.

Inicialmente, a ideia de diminuir a complexidade do problema levou a utilização de dois retângulos, visto que o número de vértices seria bem inferior ao do polígono utilizado na heurística. A Figura 7.3 mostra a utilização da função *inpolygon* em um teste para determinar quais os pontos são interiores ao retângulo 2 localizado no centro do retângulo 1:

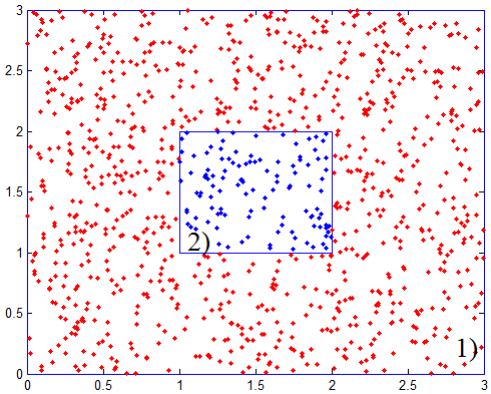


Figura 7.3: Função *inpolygon* aplicada ao problema de um retângulo. Fonte: [34].

Além de atribuir 0 ou 1 dependendo da localização do ponto em relação ao polígono, para demonstrar a eficiência do método destacaremos em vermelho (in) os pontos da elipse inscritos em P e em azul (in) os pontos exteriores ou localizados na fronteira do poligono, conforme mostra a Figura 7.4:

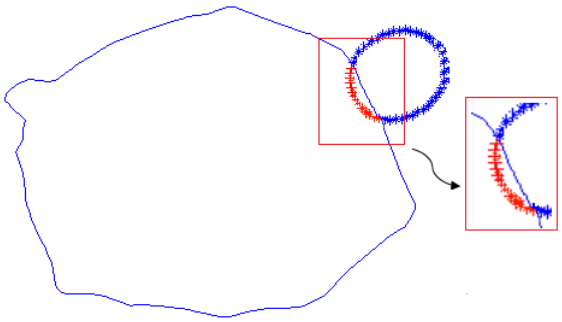


Figura 7.4: Heurística construtiva: exemplo da verificação da inscrição em ε. Fonte: Elaborado pela autora.

O tempo de processamento da verificação é linear [40], variando de acordo com o número de arestas do polígono e o número de pontos das elipses obtidas com a heurística. Para estimar o tempo de processamento, por exemplo no teste realizado, utilizando um polígono com 10.000 arestas, obteve-se o coeficiente angular $\alpha = 0.0000021$ [40]. A variação do tempo de processamento é mostrada na Figura 7.5, de modo uma grande quantidade de pontos exteriores ao polígono torna o processo mais lento.

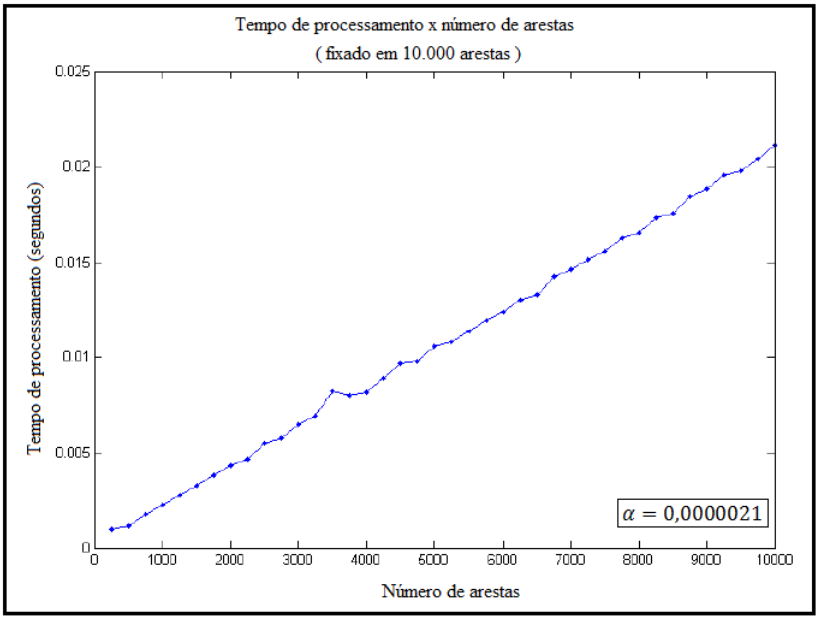


Figura 7.5: Tempo de processamento x número de arestas. Fonte: [40].

Caso o procedimento para a verificação da posição relativa dos pontos das elipses em relação ao polígono detectar pontos exteriores ao polígono, aplica-se a redução gradativa do semi-eixo maior da elipse parcialmente inscrita. A redução determina um novo comprimento para o semi-eixo maior aw, com esse parâmetro calcula-se novamente o semi-eixo menor bw, o centro $C(C_x, C_y)$, os vértices e os pontos aleatórios até que todos esses pontos estejam inscritos em P. Os parâmetros obtidos são armazenados na matriz da elipse empacotada e faz-se uma nova verificação dos pontos. A redução do semi-eixo maior, o cálculo das coordenadas e a verificação dos pontos é realizada recursivamente até que todos os pontos da matriz estejam contidos em P [39].

O comprimento do semi-eixo maior obtido pela redução é calculado com a condição adicional mostrada no Capítulo 3:

$$aw_i' = aw_i - \alpha \cdot aw_i \tag{7.5}$$

$$aw_i' = aw_i \cdot (1 - \alpha) \tag{7.6}$$

onde $\alpha = 0.1$ é o fator de redução.

Após a verificação da inscrição das elipses empacotadas em P, realiza-se as simulações para validação da heurística construtiva para o empacotamento de elipses tangentes em polígonos de n lados.

8 Simulações da heurística construtiva de empacotamento

Este capítulo descreve as simulações realizadas como validação da heurística desenvolvida para o empacotamento de elipses tangentes em um polígono de n lados. São apresentados os resultados de cada iteração e uma análise do comportamento da heurística após cada iteração realizada.

Como forma de validar a heurística desenvolvida para empacotar elipses tangentes em polígonos de n lados foram realizadas diversas simulações com contornos que descrevem polígonos convexos e côncavos, com o número de vértices diferente.

A heurística desenvolvida foi implementada com a linguagem de programação MatLab [34]. As simulações computacionais foram realizadas com o *software* MatLab versão R2010a (licença acadêmica) com um computador portátil (laptop), processador Intel® Core^{TM} i5 com 2.30 GHz de capacidade de processamento, disco rígido (HD) de 500 Gb e memória instalada (RAM) de 4 Gb.

8.1 Experimentos numéricos

Primeiramente, para cada polígono realizou-se o posicionamento inicial das elipses com base na elipse inicial. Realizada essa etapa e armazenados os parâmetros necessários para os cálculos posteriores, foram efetuadas 10 iterações em todos os polígonos utilizados. Os dados obtidos, com o posicionamento inicial e as iterações, serão utilizados para comparar o comportamento da heurística em cada tipo de polígono. As simulações da heurística construtiva foram realizadas com 7 polígonos, côncavos e convexos, sendo o número de pontos diferente em cada polígono.

Os valores obtidos em cada iteração para os parâmetros de cada elipse empa-

cotada estão listados a seguir. Na primeira coluna está especificado se a elipse é tangente ao vértice ou quadrante de ε, na segunda define-se a ordem de empacotamento. Nas demais colunas estão expressos os parâmetros relativos ao comprimento do semi-eixo maior (aw) em mm, semi-eixo menor (bw) em mm, excentricidade, ângulo de rotação $\sigma(^c)$, área em mm^2 e tempo de processamento dado em segundos. Na Tabela 8.1 é descrito o posicionamento de cada elipse empacotada em relação a tangência nos vértices e nos quadrantes.

Tabela 8.1: Legenda utilizada nas simulações na descrição da tangência

Elipse	**Posicionamento em relação a ε**
E1	Elipse tangente ao vértice B1 de ε
E2	Elipse tangente ao vértice C1 de ε
E3	Elipse tangente ao vértice A1 de ε
E4	Elipse tangente ao vértice D1 de ε
E5	Elipse tangente ao 1º Quadrante de ε
E6	Elipse tangente ao 2º Quadrante de ε
E7	Elipse tangente ao 3º Quadrante de ε
E8	Elipse tangente ao 4º Quadrante de ε

Para os resultados numéricos obtidos em cada iteração, utilizou-se o formato numérico padrão do Matlab [34] *(format short)*, o qual apresenta os valores numéricos com quatro casas decimais. Para realizar o ajuste de curva faz-se necessário calcular o tempo total de processamento, área total ocupada pelas elipses empacotadas, área do polígono, área e percentual de área restante.

O tempo de processamento foi calculado com os comandos *tic* (inicializa o contador de tempo) e *toc* (finaliza a contagem do tempo decorrido) do MatLab. Para uma melhor análise, a área de ε e a área da maior elipse tangente serão destacadas em cada tabela. Na simulação numérica descrita abaixo utilizou-se um polígono em 122 vértices, área do polígono $215,7690mm^2$, excentricidade de ε fixa em 0,593946, número de pontos para a discretização do polígono fixo em 50, tolerância $1E-3$ e número máximo de iterações fixado em 10.

Na Tabela 8.2 podemos observar os parâmetros de cada elipse empacotada no posicionamento inicial (iteração 0):

Tabela 8.2: Simulação numérica - Posição inicial

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	0,8755	0,7043	0,5939	1,7957	1,9373	0,6750
V	**E1**	0,4377	0,3522	0,5939	1,7957	0,4843	0,1844
V	**E2**	0,4226	0,3400	0,5939	1,7957	0,4514	3,6509
V	**E3**	0,4377	0,3522	0,5939	1,7957	0,4843	0,0464
V	**E4**	0,4226	0,3400	0,5939	1,7957	0,4514	0,0445
Q	**E5**	0,6356	0,5114	0,5939	1,7957	1,0212	0,0520
Q	**E6**	0,6356	0,5114	0,5939	1,7957	1,0212	0,0559
Q	**E7**	0,6356	0,5114	0,5939	1,7957	1,0212	0,0434
Q	**E8**	0,6356	0,5114	0,5939	1,7957	1,0212	0,0429

Na iteração inicial mostrada na Tabela 8.2 é possível observar claramente o comportamento da heurística desenvolvida. A elipse $E1$ tangentes ao vértice $B1$ e a elipse $E3$ tangente ao vértice $A1$ de ε possuem a mesma área no empacotamento inicial. Já as elipses $E2$ e $E4$, tangentes respectivamente aos vértices $C1$ e $D1$ de ε possuem a mesma área conforme definido na heurística. As elipses tangentes aos quadrantes de ε possuem todas a mesma área visto que o parâmetro λ utilizado em cada elipse é o mesmo, considerando a distância do centro $C(x_c, y_c)$ ao ponto de tangência definido pela intersecção da bissetriz do quadrante com a curva.

Tabela 8.3: Simulação numérica - Iteração 1

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	1,2818	1,0312	0,5939	3,9907	4,1527	0,6556
V	**E1**	0,6409	0,5156	0,5939	3,9907	1,0382	0,0989
V	**E2**	0,6187	0,4978	0,5939	3,9907	0,9676	3,6099
V	**E3**	0,6409	0,5156	0,5939	3,9907	1,0382	0,0381
V	**E4**	0,6187	0,4978	0,5939	3,9907	0,9676	0,0442
Q	**E5**	0,9306	0,7487	0,5939	3,9907	2,1889	0,0583
Q	**E6**	0,9306	0,7487	0,5939	3,9907	2,1889	0,0651
Q	**E7**	0,9306	0,7487	0,5939	3,9907	2,1889	0,0677
Q	**E8**	0,9306	0,7487	0,5939	3,9907	2,1889	0,0668

No decorrer da heurística, a direção de busca assumida no posicionamento das elipses tangentes a ε visa empacotar as elipses o mais próximo do contorno, sempre rotacionando ε no sentido anti-horário. Após a iteração inicial, as elipses começam a se posicionar próximo do contorno de modo que não há necessidade de aplicar o processo de redução do semi-eixo maior e o tempo de processamento se mantém estável comparado ao tempo expresso na Tabela 8.2. Com base na Tabela 8.3 podemos observar a variação no ângulo de rotação, aumento dos semi-eixos e da área das elipses.

Tabela 8.4: Simulação numérica - Iteração 2

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	1,8767	1,5098	0,5939	4,6053	8,9016	0,7701
V	**E1**	0,9384	0,7549	0,5939	4,6053	2,2254	0,0902
V	**E2**	0,9059	0,7288	0,5939	4,6053	2,0741	3,6425
V	**E3**	0,4488	0,3611	0,5939	4,6053	0,5091	8,1056
V	**E4**	0,9059	0,7288	0,5939	4,6053	2,0741	0,0366
Q	**E5**	1,3625	1,0962	0,5939	4,6053	4,6922	0,0381
Q	**E6**	0,7241	0,5825	0,5939	4,6053	1,3252	5,8570
Q	**E7**	1,2263	0,9865	0,5939	4,6053	3,8007	1,3139
Q	**E8**	1,3625	1,0962	0,5939	4,6053	4,6922	0,0460

O tempo de processamento para o empacotamento de cada elipse mostrado na Tabela 8.4 é calculado em segundos, as elipses que possuem um tempo de processamento bem mais elevado como as elipses $E2, E3$ e $E6$ passaram pelo processo de redução do semi-eixo maior. O valor de σ obtido indica que ε fez um giro de aproximadamente 263.8642^{o} no sentido anti-horário do sistema de eixos cartesianos original.

A cada iteração realizada, a área total das elipses aumenta reduzindo o percentual de área restante e demandando um baixo tempo de processamento.

Tabela 8.5: Simulação numérica - Iteração 3

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	2,7477	2,2105	0,5939	6,7636	19,0814	0,8943
V	**E1**	1,3738	1,1053	0,5939	6,7636	4,7704	0,0812
V	**E2**	1,3263	1,0670	0,5939	6,7636	4,4460	3,6277
V	**E3**	1,3738	1,1053	0,5939	6,7636	4,7704	0,0445
V	**E4**	1,3263	1,0670	0,5939	6,7636	4,4460	0,0429
Q	**E5**	1,7954	1,4444	0,5939	6,7636	8,1470	1,8906
Q	**E6**	1,9949	1,6049	0,5939	6,7636	10,0581	0,0413
Q	**E7**	1,9949	1,6049	0,5939	6,7636	10,0581	0,0477
Q	**E8**	1,9949	1,6049	0,5939	6,7636	10,0581	0,0472

Considerando o ângulo de giro estimado em 387.5257º, o posicionamento das elipses empacotadas é semelhante ao posicionamento na iteração inicial. A variação que ocorre é no tamanho das elipses, sendo que a redução do semi-eixo maior foi necessária apenas na elipse $E5$, localizada no 1º quadrante de ε. Isso é facilmente observado comparando a sua área às áreas das elipses tangentes aos quadrantes de ε.

Tabela 8.6: Simulação numérica - Iteração 4

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	4,0229	3,2364	0,5939	7,3128	40,9028	1,0855
V	**E1**	0,6312	0,5078	0,5939	7,3128	1,0070	12,6117
V	**E2**	1,2741	1,0250	0,5939	7,3128	4,1025	9,7529
V	**E3**	2,0114	1,6182	0,5939	7,3128	10,2257	0,0444
V	**E4**	1,9419	1,5622	0,5939	7,3128	9,5304	0,0406
Q	**E5**	0,8249	0,6636	0,5939	7,3128	1,7198	19,0595
Q	**E6**	2,9207	2,3497	0,5939	7,3128	21,5604	0,0430
Q	**E7**	2,9207	2,3497	0,5939	7,3128	21,5604	0,0787
Q	**E8**	0,9166	0,7374	0,5939	7,3128	2,1232	18,1769

Na iteração 4, mostrada na Tabela 8.6, as elipses $E1, E2, E4, E5$ e $E8$ possuem pontos exteriores ao polígono P, sendo que a direção de busca tende a posicionar ε na direção do contorno, apenas nas elipses $E3, E6$ e $E7$ não ocorre o redimensionamento do

semi-eixo maior. Quando aplicada a verificação da inscrição e posteriormente a redução do semi-eixo, ocorre um aumento no tempo de processamento das elipses empacotadas.

Tabela 8.7: Simulação numérica - Iteração 5

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	4,9655	3,9948	0,5939	7,8917	62,3174	1,1689
V	**E1**	0,6311	0,5077	0,5939	7,8917	1,0066	16,8965
V	**E2**	2,3969	1,9283	0,5939	7,8917	14,5201	3,6797
V	**E3**	2,0110	1,6179	0,5939	7,8917	10,2216	5,3929
V	**E4**	0,6769	0,5446	0,5939	7,8917	1,1582	15,6717
Q	**E5**	0,4870	0,3918	0,5939	7,8917	0,5994	28,3136
Q	**E6**	3,6051	2,9003	0,5939	7,8917	32,8483	0,0438
Q	**E7**	1,1313	0,9102	0,5939	7,8917	3,2348	22,4600
Q	**E8**	0,7423	0,5971	0,5939	7,8917	1,3925	25,9846

A tendência da heurística a partir da 5ª iteração é que ε comece a ocupar grande parte do polígono, deixando pouco espaço para empacotar as demais elipses. Com pouco espaço disponível para alocação das elipses tangentes aumenta a probabilidade das elipses não estarem totalmente inscritas. Na iteração 5, a elipse $E6$ posicionada no 2º quadrante de ε possui a maior área entre as elipses tangentes. O ângulo de giro nessa iteração é de 452.1611º em relação ao sistema de eixos original.

Tabela 8.8: Simulação numérica - Iteração 6

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	6,1291	4,9309	0,5939	8,0142	94,9435	1,3481
V	**E1**	0,6310	0,5076	0,5939	8,0142	1,0062	22,1636
V	**E2**	2,9585	2,3801	0,5939	8,0142	22,1220	3,6115
V	**E3**	0,7011	0,5640	0,5939	8,0142	1,2422	28,6305
V	**E4**	0,1720	0,1384	0,5939	8,0142	0,0748	25,3599
Q	**E5**	1,0180	0,8190	0,5939	8,0142	2,6191	31,0902
Q	**E6**	2,9195	2,3488	0,5939	8,0142	21,5431	13,9141
Q	**E7**	0,0902	0,0726	0,5939	8,0142	0,0206	39,5954
Q	**E8**	0,7421	0,5970	0,5939	8,0142	1,3919	33,6012

As elipses $E4$ e $E7$ possuem área inferior a $1mm^2$, o procedimento para empacotar essas elipses se torna lento, elevando o tempo total de simulação. Na próxima iteração, com a rotação realizada por ε, elas passam a ocupar espaços ociosos do polígono. O ângulo de rotação, obtido e mostrado na Tabela 8.8, na iteração 6 é $\sigma = 8,0142$, o que representa 459.1798º no sentido anti-horário.

Tabela 8.9: Simulação numérica - Iteração 7

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	6,3779	5,1311	0,5939	7,7980	102,8102	1,5201
V	**E1**	0,3489	0,2807	0,5939	7,7980	0,3077	25,8579
V	**E2**	2,7708	2,2291	0,5939	7,7980	19,4035	6,4414
V	**E3**	1,2355	0,9939	0,5939	7,7980	3,8578	22,3089
V	**E4**	0,1989	0,1600	0,5939	7,7980	0,1000	26,2007
Q	**E5**	0,3324	0,2674	0,5939	7,7980	0,2793	38,9541
Q	**E6**	3,0381	2,4442	0,5939	7,7980	23,3281	14,4636
Q	**E7**	0,2992	0,2407	0,5939	7,7980	0,2262	39,2924
Q	**E8**	0,2423	0,1950	0,5939	7,7980	0,1484	39,8586

Conforme a Tabela 8.9, a área de ε é $102,8102mm^2$ ocupando cerca de 47 % da área do polígono. As elipses com área inferior a $1mm^2$ resultam em modelos de lapidação pequenos, mesmo assim não são descartadas. A maior elipse tangente está posicionada no 2º quadrante ε.

Tabela 8.10: Simulação numérica - Iteração 8

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	6,5191	5,2447	0,5939	7,8550	107,4133	2,2316
V	**E1**	0,4403	0,3542	0,5939	7,8550	0,4900	25,7254
V	**E2**	2,8321	2,2785	0,5939	7,8550	20,2723	6,4889
V	**E3**	1,0229	0,8229	0,5939	7,8550	2,6444	25,9684
V	**E4**	0,1482	0,1192	0,5939	7,8550	0,0555	27,3618
Q	**E5**	0,4661	0,3750	0,5939	7,8550	0,5491	38,7020
Q	**E6**	2,7948	2,2484	0,5939	7,8550	19,7418	17,5539
Q	**E7**	0,2006	0,1614	0,5939	7,8550	0,1017	41,1314
Q	**E8**	0,3058	0,2460	0,5939	7,8550	0,2364	40,1459

O procedimento adotado na heurística assume que a cada iteração as elipses variam de tamanho dependendo do espaço ocioso. Isso facilmente é observado nas iterações 6, 7 e 8. Na iteração 6, as elipses $E4$ e $E7$ possuem área muito pequena, com o movimento de rotação realizado por ε, o novo espaço que a elipse tangente vai ocupar é maior fazendo com que a área aumente na iteração 7. O novo movimento de rotação empacota as elipses mais próximo do contorno de modo que a área das elipses $E4$ e $E7$ volte a diminuir na iteração 8, conforme verificado na Tabela 8.10.

Pelas deformidades do contorno, qualquer variação no ângulo de rotação altera o empacotamento. Na iteração 7 onde a área das elipses $E4$ e $E7$ aumenta, o valor do ângulo de rotação é 446.7924º passando para 450.0583º na iteração 8. A variação do ângulo é inferior a 5º e resulta em uma mudança significativa na área das elipses posicionadas próximo do contorno do polígono.

Tabela 8.11: Simulação numérica - Iteração 9

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	6,6515	5,3512	0,5939	6,9526	111,8194	1,9260
V	**E1**	0,1566	0,1260	0,5939	6,9526	0,0620	28,8639
V	**E2**	1,2439	1,0007	0,5939	6,9526	3,9106	21,4351
V	**E3**	2,1820	1,7554	0,5939	6,9526	12,0336	12,4351
V	**E4**	0,6611	0,5318	0,5939	6,9526	1,1045	23,2037
Q	**E5**	0,9943	0,7999	0,5939	6,9526	2,4986	34,7396
Q	**E6**	1,8709	1,5052	0,5939	6,9526	8,8468	26,7688
Q	**E7**	2,0788	1,6724	0,5939	6,9526	10,9220	24,9045
Q	**E8**	0,1842	0,1482	0,5939	6,9526	0,0858	42,1190

O empacotamento das elipses realizado na iteração 9, mostrada na Tabela 8.11, utiliza mais de 70% da área do polígono. Com a análise das iterações anteriores é possível observar que a elipse tangente com maior área varia, por exemplo, na iteração 7 era a elipse $E6$, na iteração 8 era a elipse $E2$ e na iteração 9 a elipse $E3$. Essa característica define a heurística construtiva de empacotamento, onde o tamanho das elipses não é fixado e sim calculado com base nos parâmetros de ε.

Tabela 8.12: Simulação numérica - Iteração 10

		aw	***bw***	***e***	σ	***A***	***Tp***
	ε	7,5383	6,0646	0,5939	7,0693	143,6253	1,9173
V	**E1**	0,7760	0,6243	0,5939	7,0693	1,5221	27,1735
V	**E2**	1,1419	0,9187	0,5939	7,0693	3,2955	26,2188
V	**E3**	0,5657	0,4551	0,5939	7,0693	0,8089	38,2878
V	**E4**	0,3225	0,2595	0,5939	7,0693	0,2629	30,1186
Q	**E5**	1,0142	0,8159	0,5939	7,0693	2,5995	40,3579
Q	**E6**	1,3912	1,1192	0,5939	7,0693	4,8915	36,9712
Q	**E7**	0,8215	0,6609	0,5939	7,0693	1,7056	42,1319
Q	**E8**	0,4851	0,3902	0,5939	7,0693	0,5947	45,1757

Com o número máximo de iterações estabelecido em 10, a busca pelo empacotamento de elipses tangentes a ε é finalizada. A área de ε mostrada na Tabela 8.12 ocupa cerca de 60% da área poligonal. Para validar a heurística desenvolvida, foram realizadas mais que 10 iterações constatando-se que a variação significativa está diretamente ligada a ε.

Os polígonos utilizados para realizar as simulações tem entre 120 e 400 pontos. Para verificar a aplicabilidade da heurística desenvolvida em polígonos com um número de vértices inferior a 120, utilizou-se um polígono com 23 vértices.

O empacotamento de elipses realizado em cada iteração é mostrado na Figura 8.1, sendo que todas as elipses estão totalmente inscritas em P sem que haja segmentos ou pontos fora da região de fronteira do polígono.

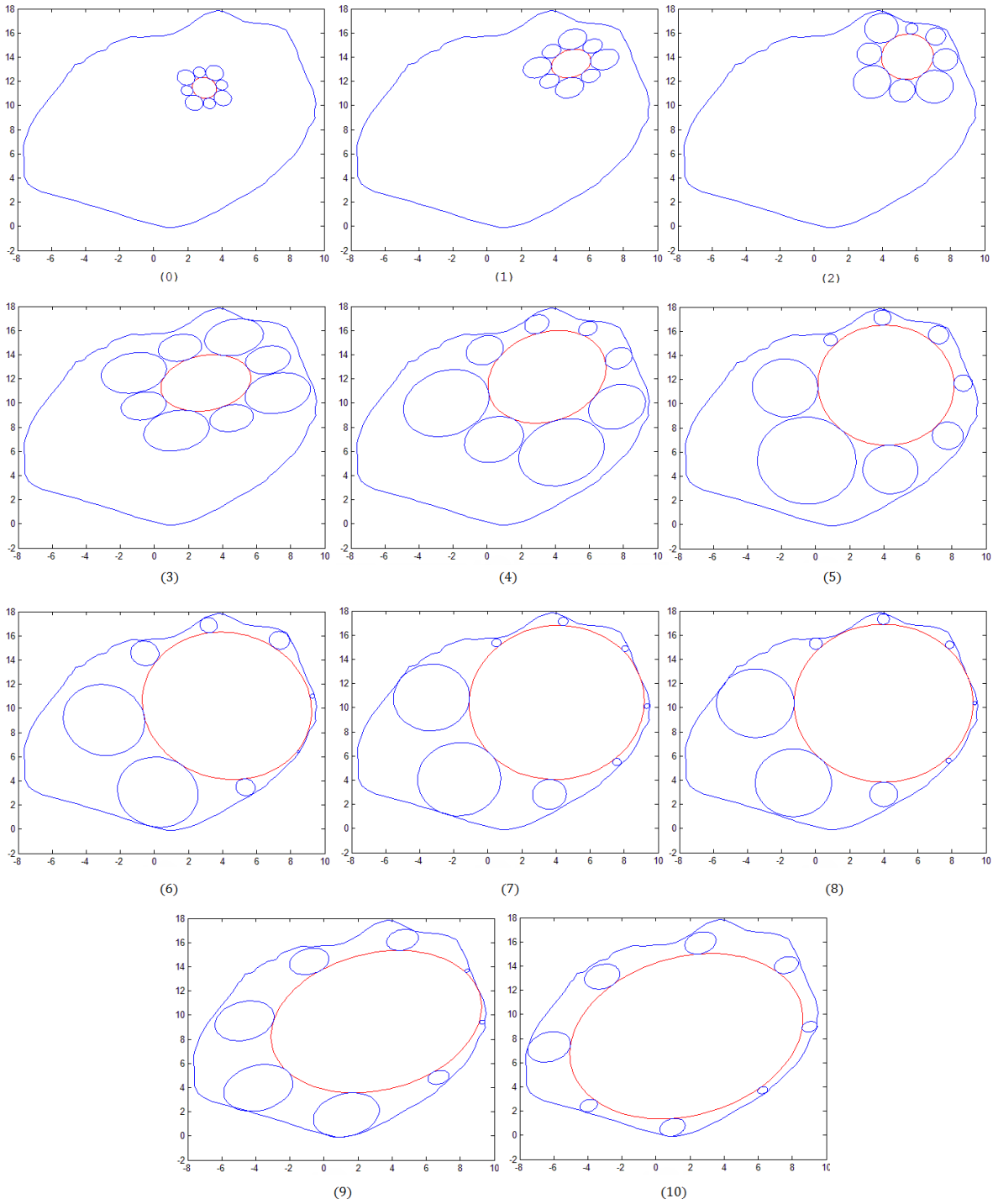


Figura 8.1: Empacotamento de elipses tangentes a uma elipse inicial em um polígono de n lados. Fonte: Heurística construtiva.

A análise dos gráficos e o ajuste de curva realizado para verificar o comportamento exponencial da heurística serão descritos no Capítulo 9.

9 Análise dos resultados obtidos com as simulações

Este capítulo é dedicado à análise dos resultados obtidos com as simulações da heurística construtiva de empacotamento de elipses. Algumas questões sobre o ajuste de curva realizado com a série de dados obtida em cada simulação são discutidas.

Os resultados obtidos através das simulações realizadas apontam que a cada iteração o percentual de área restante decresce, objetivo esse da heurística desenvolvida. Com o propósito inicial de desenvolver uma metodologia analítica e computacional com a qual fosse possível encontrar a solução viável para o problema de empacotamento de elipses - idênticas ou não - sem sobreposição, esse estudo também mostrou-se satisfatório quando analisado o tempo de processamento total da heurística.

Em todas as simulações realizadas, o tempo de processamento total da heurística não ultrapassou 500 segundos. Por ser considerado um procedimento de buscas aleatórias, a heurística converge para a solução ótima em um tempo de processamento viável.

Na Tabela 9.1 estão os resultados gerais computados após cada iteração da simulação mostrada no Capítulo 8. O polígono utilizado na simulação possui 122 vértices e área poligonal de $215,7690mm^2$.

A cada iteração realizada, o tempo total de processamento (ttp) aumenta devido a verificação da inscrição e a área total (at) das elipses empacotadas cresce, consequentemente a área restante (ar) decresce.

Para a análise dos dados obtidos em cada simulação realizada foi utilizado o código disponível no Apêndice C e validado com o *Toolbox Curve Fitting*^{TM} [34], considerando o ajuste de curva exponencial, dado pela Equação 9.1:

$$y = a \cdot e^{bx} \tag{9.1}$$

Tabela 9.1: Resultados obtidos com a heurística construtiva

I	**Ap**	**Ttp**	**At**	**Ar**	**Pr**
0	215,7690	4,7954	7,8932	207,8757	96,3418
1	215,7690	4,7047	16,9199	198,8491	92,1583
2	215,7690	19,8998	30,2945	185,4744	85,9597
3	215,7690	6,7174	75,8355	139,9335	64,8534
4	215,7690	60,8931	112,7322	103,0368	47,7533
5	215,7690	119,6117	127,2988	88,4702	41,0023
6	215,7690	199,3145	144,9635	70,8055	32,8154
7	215,7690	214,8979	150,4614	65,3076	30,2674
8	215,7690	225,3092	151,5046	64,2645	29,7839
9	215,7690	216,3957	151,2833	64,4857	29,8865
10	215,7690	288,3527	159,3059	56,4631	26,1683

Tabela 9.2: Séries de dados utilizada no ajuste de curvas.

Iteração	**Percentual de área restante (%)**	**Tempo de processamento (s)**
0	96,3418	4,7954
1	92,1583	4,7047
2	85,9597	19,8998
3	64,8534	6,7174
4	47,7533	60,8931
5	41,0023	119,6117
6	32,8154	199,3145
7	30,2674	214,8979
8	29,7839	225,3092
9	29,8865	216,3957
10	26,1683	288,3527

A Tabela 9.2 expressa as duas séries de dados (percentual de área restante e tempo de processamento) mostradas na Tabela 9.1.

O ajuste da série de dados da Tabela 9.2 em curvas parametrizadas [42] mostrado na Figura 9.1, além de determinar a tendência da heurística, gera uma função que serve para comparações da heurística em cada polígono e para uma possível generalização do processo.

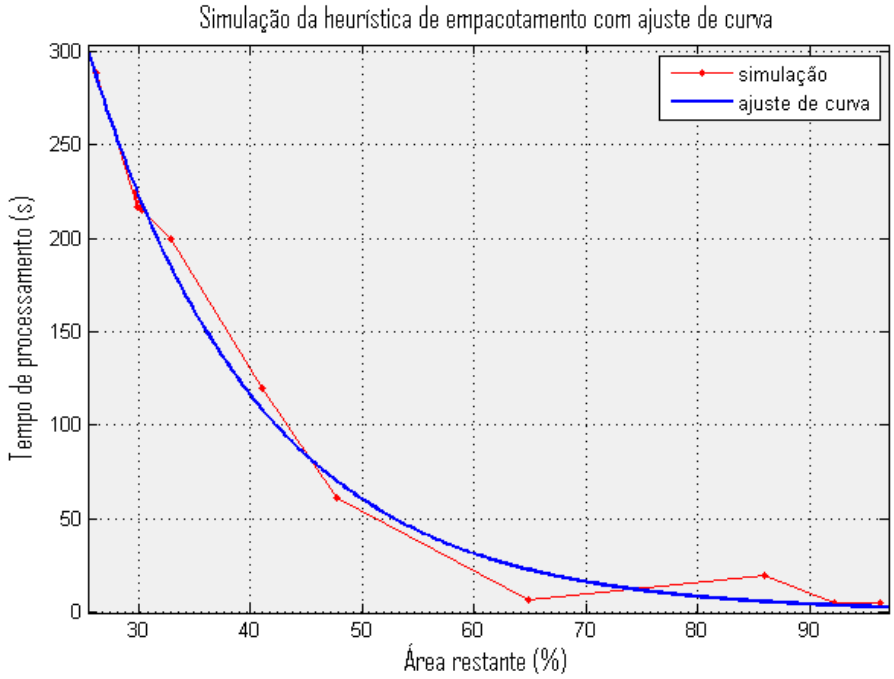


Figura 9.1: Simulação mostrada no Capítulo 8 com ajuste de curva . Fonte: Elaborado pela autora.

O gráfico mostrado na Figura 9.1 demonstra o comportamento exponencial da redução de área restante em função do tempo de processamento decorrido. Os resultados numéricos mostrados nessa simulação foram detalhados no Capítulo 8 e o tempo total de processamento da simulação não ultrapassa 300 segundos, garantindo uma convergência rápida para a solução ótima.

A função obtida através do ajuste de curva exponencial é dada pela Equação 9.2:

$$y = 137,9 \cdot e^{-1,403x} \tag{9.2}$$

Com a função exponencial 9.2 é possível determinar o percentual de área

restante dado um tempo de processamento decorrido ou ainda estabelecer qual o tempo gasto para se obter a solução ótima após cada iteração.

A simulação mostrada abaixo evidencia ainda mais o comportamento exponencial da heurística de empacotamento A Tabela 9.3 expõe os dados obtidos da simulação realizada com um polígono de 121 vértices e área poligonal $467,1110mm^2$, que serviram de base para o ajuste de curva realizado.

Tabela 9.3: Séries de dados utilizada no ajuste de curvas.

Iteração	**Percentual de área restante (%)**	**Tempo de processamento (s)**
0	94,6204	4,5300
1	88,4684	4,6430
2	77,3639	13,7553
3	61,5777	59,5862
4	47,6632	148,0900
5	35,1403	251,9465
6	27,2375	404,3782
7	23,9907	451,5450
8	22,6661	468,1797
9	22,2781	473,3256
10	22,2781	474,0371

Os resultados expostos na Tabela 9.3 mostram que na iteração final o percentual de área restante é inferior a 25 % da área do polígono e faz-se o ajuste de curva conforme pode ser visto na Figura 9.2.

Com o ajuste de curva realizado através da série de dados da Tabela 9.3, temos a função exponencial 9.3 obtida :

$$y = 51,33 \cdot e^{-1,796x} \tag{9.3}$$

A solução ótima é obtida na iteração 9, sendo que na iteração 10 o percentual de área restante não se altera e demanda mais tempo de processamento. Essa simulação tem um tempo total de processamento inferior a 500 segundos para obter o empacotamento ideal das elipses tangentes a ε, que pode ser observado na Figura 9.2.

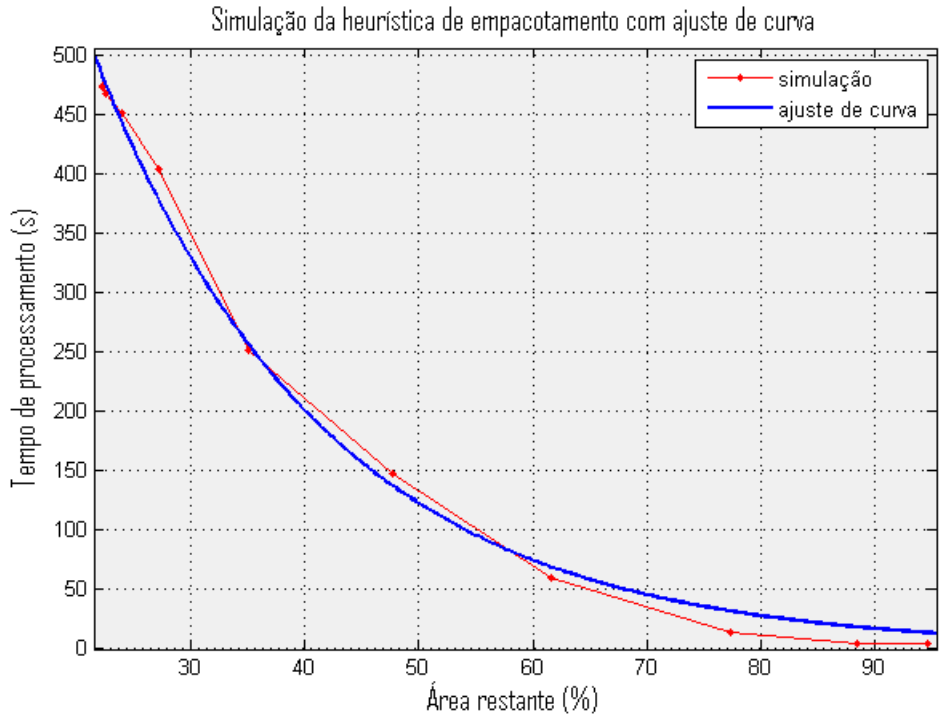


Figura 9.2: Simulação realizada e o ajuste de curva . Fonte: Elaborado pela autora.

Outro fator que deve ser destacado na análise dos resultados obtidos é a fácil adaptação da heurística em polígonos com grandes concavidades. A heurística desenvolvida realiza buscas em direções aleatórias mas não emprega nenhum método que utilize a informação do vetor gradiente $\nabla aw\,(x_c, y_c, \sigma)$ e $\nabla aw\,(C_x, C_y, \sigma)$ uma vez que não podemos obtê-los analiticamente, e também queremos evitar o caminho alternativo de determinar esses gradientes computacionalmente ou cair em mínimos e máximos locais. A Figura 9.3 mostra duas situações em que a heurística empacota as elipses em regiões de contorno com concavidades acentuadas.

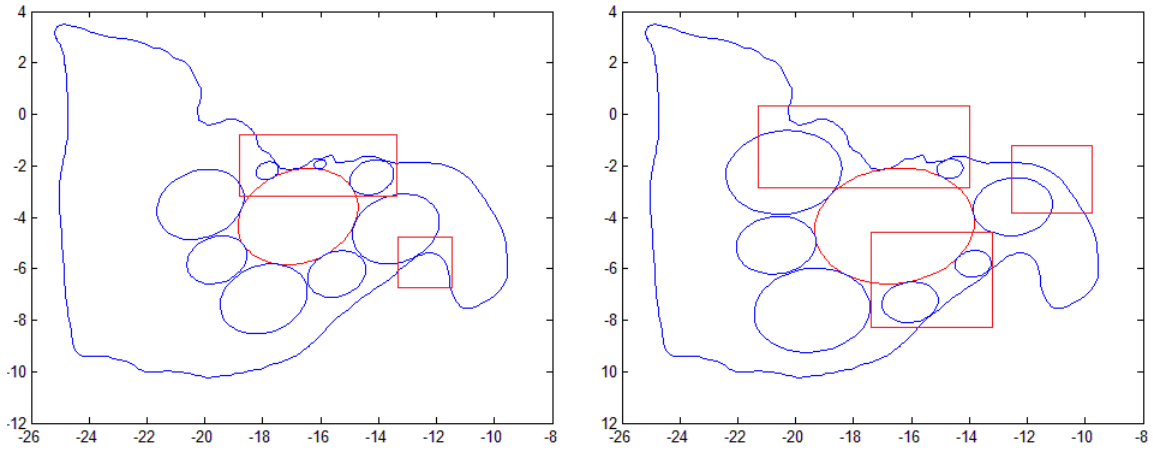


Figura 9.3: Elipses empacotadas restritas ao contorno com concavidades. Fonte: Elaborado pela autora.

As regiões variam de iteração para iteração, destacando a facilidade movimentar e deslocar as elipses empacotadas na heurística, não mantendo nenhuma elipse fixa nas concavidades dos polígonos. Na Figura 9.3 os elipses tangenciam o contorno mas não ultrapassam a fronteira, fato que se deve a verificação da inscrição no polígono.

10 Conclusão

Finalizando, este capítulo apresenta os principais aspectos do trabalho desenvolvido, seguindo com uma discussão das principais contribuições alcançadas e, posteriormente, com sugestões de trabalhos futuros.

10.1 Considerações finais

No decorrer da pesquisa buscou-se desenvolver uma metodologia eficaz e com baixo tempo computacional, que fosse capaz de posicionar múltiplas elipses tangentes a elipse inicial obedecendo o contorno formado pelos vértices do polígono, bem como, maximizar o aproveitamento volumétrico da gema. O estudo inicialmente foi dirigido para a implementação da busca pelos pontos de tangência sem a utilização das transformações aplicadas em equações paramétricas, porém, pela complexidade das matrizes de pontos obtidos em cada quadrante preferiu-se adotar o método de intersecção da bissetriz do quadrante com a curva, resultando no valor do ângulo utilizado na equação paramétrica modificada.

O procedimento do cálculo dos vértices da elipse inicial demandou bastante tempo, visto que da forma como foram ordenados na heurística desenvolvida assumiu-se o ângulo de rotação negativo, evitando assim a dedução de novas fórmulas para os vértices.

A verificação da inscrição utilizando a função *inpolygon* do Matlab tornou o procedimento de empacotamento muito mais rápido. Sem a utilização dessa função, a verificação deveria ser realizada através da intersecção de cada aresta do polígono com cada elipse empacotada. Considerando um polígono com um número de pontos pequeno isso não seria tão complexo, mas com polígonos com muitos vértices o sistema a ser revolvido envolveria uma grande quantidade de variáveis.

O entendimento de conceitos, presentes na Ciência da Computação e estudados na área de Computação Gráfica, foi necessário para a compreensão dos algoritmos do

método de Hooke e Jeves e da linguagem de programação utilizada. Dessa forma, é possível afirmar que este trabalho foi um complemento na apropriação do conhecimento. Outro fator importante foi o estudo do método de otimização de Hooke e Jeeves, método que não é estudado no decorrer dos cursos de formação na área de Matemática nem em termos de pós-graduação.

A utilização de recursos computacionais e *softwares*, como o MatLab, para a verificação da eficiência do método e conceitos utilizados na resolução do problema de empacotamento de elipses, são fundamentais, no sentido de tornar a simulação mais rápida e mais precisa.

Com o software MatLab desenvolveu-se as verificações algébricas do comprimento dos semi-eixos das elipse, excentricidade e a área, além da plotagem dos polígonos com as elipses empacotados, validação da heurística, sendo que a cada iteração o resultado convergiu para a solução ótima.

Com relação ao método de otimização de Hooke e Jeeves existem poucas referências disponíveis sobre o assunto, sendo o livro de Mokhtar S. Bazaraa sobre programação não-linear a referência mais utilizada para a compreensão dos algoritmos.

A restrição de não sobrepor elipses e evitar que as elipses não estivessem totalmente inscritas em P foi obedecida, gerando elipses totalmente empacotadas e sem nenhuma sobreposição.

A realização de um trabalho envolvendo uma situação real que foi estudada inicialmente no Projeto 3D Gemas, resolvida e utilizada no contexto não só acadêmico, mas no contexto de pesquisa e desenvolvimento tecnológico, mostra a grande importância da Matemática interligada com outras áreas do conhecimento, como propõe o Programa de Pós-Graduação em Modelagem Computacional.

Os resultados obtidos são satisfatórios, considerando que a heurística construtiva foi capaz de determinar a solução ótima para os polígonos testados. Como não há um método ou heurística com a qual se possa estabelecer uma comparação, expõe-se os resultados obtidos com a esperança de que possam ser confrontados no futuro com outras heurísticas.

Acredita-se que o objetivo do presente trabalho, o qual consiste em desenvolver uma metodologia analítica e computacional, com a qual fosse possível encontrar

a solução viável para o problema de empacotamento de elipses, idênticas ou não, sem sobreposição e tangentes a cada vértice e quadrante da elipse inicial tenha sido alcançado, conforme mostrado na simulação apresentada e na análise dos resultados. No problema de empacotamento *(np-difícil)* proposto obteve-se a maximização da área total das elipses empacotadas em P e a minimização do tempo de processamento computacional.

10.2 Contribuições do trabalho

A metodologia analítica apresentada nos capítulos anteriores, serviu de fundamentação para o desenvolvimento do algoritmo que realizasse o empacotamento de elipses tangentes a uma elipse inicial com excentricidade fixa, levando em consideração o tempo de processamento do empacotamento total.

Essa heurística caracteriza por ser extensível aos mais diversos polígonos, côncavos ou convexos. Outra contribuição importante desse estudo, esta ligada a não utilização de derivadas, principalmente para obter os pontos de tangência. Destaca-se o fato de que considerando a uma elipse rotacionada em relação aos eixos normais, derivando-se implicitamente a função que a define é possível obter os pontos extremos (vértices) dessa curva no plano cartesiano. Para evitar a utilização das derivadas, optou-se por definir os vértices com base no comprimento dos semi-eixos, ângulo de rotação e informações do centro da elipse inicial.

Cabe ressaltar como contribuição do trabalho a qualidade das soluções obtidas, em um tempo computacional considerado baixo para problemas de empacotamento e buscas em direções com o método de Hooke e Jeeves. A implementação da heurística foi desenvolvida no decorrer do mestrado e sem o uso de uma heurística já existente.

Além disso, destaca-se a grande importância tecnológica e econômica das aplicações práticas no setor de lapidação da região noroeste do estado.

10.3 Propostas para trabalhos futuros

Pretende-se utilizar os métodos empregados na construção da heurística e os resultados obtidos nesse trabalho para desenvolver estudos que auxiliem na elaboração de outras heurísticas para a inscrição dos demais modelos de lapidação.

Como uma extensão desse trabalho desenvolvido propõe-se a aplicação de uma rotação σ_i nas elipses tangentes a ε buscando reposicionar e aumentar a área da elipse onde a transformação foi aplicada. Caso seja possível aplicar a rotação σ_i, utiliza-se a condição adicional inversa para calcular o novo comprimento do semi-eixo maior aw da elipse E_i, ou seja:

◦ **Condição adicional para redução do semi-eixo maior** aw**:**

$$aw' = aw \cdot (1 - \alpha), \quad \alpha \in [0, 1] \tag{10.1}$$

◦ **Condição adicional inversa para aumentar o semi-eixo maior** aw**:**

$$aw' = aw \cdot (1 + \alpha), \quad \alpha \in [0, 1] \tag{10.2}$$

Nas simulações realizadas, consideramos o tamanho do passo da condição adicional como $\alpha = 0,1$. Com o tamanho do passo pequeno, o semi-eixo maior é reduzido gradativamente e a verificação da inscrição é realizada.

A condição adicional inversa será estabelecida visando aumentar o tamanho das elipses que não tangenciam as arestas do polígono P. Com a transformação de rotação σ_i podendo ser realizada, resulta em uma nova equação paramétrica.

Sendo:

$$A = \begin{bmatrix} aw \cdot \cos(t) \cdot \cos(\sigma) & -bw \cdot \sin(t) \cdot \sin(\sigma) \\ aw \cdot \cos(t) \cdot \sin(\sigma) & bw \cdot \sin(t) \cdot \cos(\sigma) \end{bmatrix} \; t \in [0, 2\pi] \tag{10.3}$$

temos:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} xc \\ yc \end{bmatrix} + A \cdot \begin{bmatrix} \cos(\sigma_i) & \sin(\sigma_i) \\ -\sin(\sigma_i) & \cos(\sigma_i) \end{bmatrix} \tag{10.4}$$

onde σ_i é o ângulo de rotação da elipse E_i no sentido anti-horário.

Baseado em aspectos da Computação Gráfica, o mais indicado nesse caso é que sejam transladadas até a origem as elipses que forem rotacionadas [19]. Primeiramente transladam-se as elipses até a origem, aplica-se a rotação e translada novamente para

a posição inicial do centro dessa elipse. Para esse processo utiliza-se a composição de transformações lineares [8, 41].

Outra etapa futura, baseia-se em gerar e empacotar mais elipses tangentes e posicionadas nas elipses tangentes aos vértices e quadrantes de ε de modo a maximizar o percentual de área ocupada no polígono. Para isso, faz-se de forma recursiva a criação de elipses tangentes em cada elipse E_k com $k = 1, 2, ..., 8$ posicionada no vértice ou quadrante da elipse inicial. Como inquietação final, otimizar o algoritmo da heurística desenvolvida.

Referências Bibliográficas

[1] Arenales, M. N.; Morabito, R.; Yanasse, H. H. **Problemas de corte e empacotamento**. In Livro-texto de Mini-curso do XXXVI SBPO (2004), XXXVI SBPO - Simpósio Brasileiro de Pesquisa Operacional.

[2] Balbo, A. R. et al. **Cálculo Numérico Computacional: teoria, métodos numéricos, implementação computacional e programação orientada a objetos - Teoria sobre Programação Linear.** Unesp, Departamento de Matemática. Bauru, 2012.

[3] Barboza, F. L. M.; Gurmendi, A. C. **Economia mineral do Brasil**. Brasília: DNPM, 1995. p.280, mapas.

[4] Bazaraa, M. S. et al. **Nonlinear Programming:** Theory and Algorithms. 3rd ed. New Jersey: John Wiley e Sons, 2006.

[5] Beckel, C. C. **Modelo Matemático:** Algoritmo de inscrição de secções cônicas em polígonos irregulares de n-lados. [Trabalho de conclusão de curso]. Universidade de Passo Fundo, Curso de Matemática LP, Inst. de Ciências Exatas e Geociências, Soledade, 2010.

[6] Beckel, C. C.; Oro, N. T.; Emmendorfer, L. R. **An adaptation of the Hooke and Jeeves optimization algorithm for the maximization of the dimensions of an ellipse inscribed in a given non-convex polygon.** 5th LNCC Meeting on Computational Modeling. Petrópolis, RJ, 2012.

[7] Beckel, C. C.; Silva, J. T.; Ipar, C. E. A. E. L.**Verificação da influência do número de pontos em uma malha triangular resultante do processo de digitalização tridimensional de uma gema.** In: 34º Congresso Nacional de Matemática Aplicada e Computacional - XXXIV CNMAC, Águas de Lindóia. Anais XXXIV CNMAC, 2012.

[8] Boldrini, J. L. et al. **Álgebra linear.** 3. ed. ampl. rev. São Paulo: Harbra, 1986.

[9] Brusso, M. J. et al. **Tecnologia 3D Gemas: otimização do aproveitamento de gemas coradas digitalizadas tridimensionalmente**. In: Tecnologias para o setor de Gemas, Joias e Mineração. 1 ed. Porto Alegre: IGEO/UFRGS, 2010, p. 40-52.

[10] Camponogara, E. **Métodos de otimização:** Teoria e prática. [apostila técnica]. 2006. Disponível em < http://www.das.ufsc.br/ camponog/Disciplinas/DAS-9011/>. Acesso em: 10 out. 2010.

[11] Caroli, A. de; Callioli, C. A.; Feitosa, M. O. **Matrizes, vetores, geometria analítica: teoria e exercícios.** São Paulo: Nobel, 1984.

[12] Cherri, A. C.; Arenales, M. N. **O problema de corte de estoque com reaproveitamento das sobras de material - heurística FFD modificada.** XXXVII Simpósio Brasileiro de Pesquisa Operacional, Gramado - RS, Brasil, 2009.

[13] Christiansen, G. K. . **Toy Buiding Brick Patent**. Oct. 24. Disclaimer filed Mar. 31, 1978, Interlego A.G. Billund, Denmark, 1961.

[14] Coelho, D. G. **Um algoritmo evolutivo multiobjetivo aplicado ao problema de corte bidimensional guilhotinado.** Dissertação (Mestrado em Modelagem Matemática e Computacional) - Programa de Mestrado em Modelagem Matemática e Computacional, Centro Federal de Educação Tecnológica de Minas Gerais, Minas Gerais. 2011.

[15] Coelho, L. S.; Mariani, V. C. **Concepção Híbrida de Otimização por Nuvem de Partículas Aplicada ao Problema de Weber.** [artigo científico]. Disponível em < www.dcc.ufla.br/ infocomp/artigos/v4.3/art04.pdf>. Acesso em: 25 ago. 2012.

[16] Constantino, A. A.; Junior, A. M. G. **Um algoritmo genético híbrido para o problema de corte industrial bidimensional.** Acta Scientiarum - Technology, Maringá, v. 24, p. 1727-1731, 2002.

[17] Dyckhoff, H. **A typology of cutting and packing problems.** European Journal of Operational Research 44, 2 (1990), p. 145 -159.

[18] Fialho, A. B. **Instrumentação Industrial: Conceitos, Aplicações e Análises.** São Paulo: Editora Érica, 2002.

[19] Foley, J. D. et al. **Computer graphics: principles and practice.** 2. ed em C. Ed. Addison-Wesley Publishing Company. Massachusetts, 1996.

[20] Formagin, S. **Aplicação estrutural de concretos de altíssimo desempenho reforçados com fibras de aço.** Universidade Iguaçu. Revista de Ciência e Tecnologia, v. 6, nº. 1. Rio de Janeiro: Gráfica Universitária, 2006.

[21] Gentil, J. M. P. **Sistemas de equações não-lineares e problemas de empacotamento.** Trabalho de Formatura Supervisionado. Universidade de São Paulo, São Paulo, 2008.

[22] Gilmore, P. C.; Gomory, R. E. **A linear programming approach to the cutting-stock problem**. Operations Research 9, 1961, p. 849-859 e Operations Research, 11, 1963 p. 863-888.

[23] Hooke, R.; Jeeves, T. A. **This week's citation classic.** Disponível em < www.garfield.library.upenn.edu/classics1980/A1980JK10100001.pdf>. Acesso em: 10 set. 2011.

[24] Horn, J. **Resource-based fitness sharing.** In Guervós, J. J. M., Adamidis, P., Beyer, H.-G., Martín, J. L. F.-V., and Schwefel, H.P. (Ed.s), Parallel Problem Solving From Nature (PPSN VII, Lecture Notes in Computer Science, Vol. 2439, p. 381 - 390). 2002. Berlin/Heidelberg: Springer.

[25] Horn, J. **Coevolving species for shape nesting.** In Schaeffer, J. D. (Ed.), The 2005 IEEE Congress on Evolutionary Computation (IEEE CEC 2005, p. 1800 - 1807). Piscataway, NJ: IEEE Press.

[26] Jones, M. T. **Artificial intelligence : a systems approach.** Infinity Science Press LLC. Hingham, Massachusetts, 2008.

[27] Kantorovich, L. V. **Mathematical methods of organizing and planning production** -(traduzido de um trabalho em russo datado de 1939). Management Science 6 (1960), 366 - 422.

[28] Kahan, M. **Miscalculating Area and Angles of a Needle-like Triangle.** University of California, Berkeley, EUA. 2000.

[29] Kolman, B. **Introdução à álgebra linear com aplicações.** 8. ed. Rio de Janeiro: LTC, 2006.

[30] Leithold, L. **O Cálculo com Geometria Analítica**. 3. ed. São Paulo: Harbra, v.1. 1994.

[31] Lima, K. F.; Mariani, V. C. **Métodos Computacionais - Matlab:** Uma introdução. 2008.

[32] Martins, T. C. **Estudo do Recozimento Simulado e do Polígono de Obstrução Aplicados ao Problema de Empacotamento Rotacional de Polígonos Irregulares Não-Convexos em Recipientes Fechados.** [Tese de doutorado]. Escola Politécnica da Universidade de São Paulo. Dep. de Eng. de Mecatrônica e de Sistemas Mecânicos. São Paulo, SP. 2007, 103 p.

[33] Maschio, C.; Santos, A. A.; Schiozer, D. J. **Aplicação do método Simplex no processo de ajuste de histórico assistido.** In: RIO OIL e GAS EXPO AND CONFERENCE, XI, 2006, Rio de Janeiro. Anais. Rio de Janeiro: Instituto Brasileiro de Petróleo e Gás - IBP, 2006. p. 01-08.

[34] MatLab. **MatLab and Simulink for Technical Computing.** R2012a, Mathworks Inc., 2002.

[35] MeshLab. **Meshlab**. Disponível em http://meshlab.sourceforge.net/. Acesso em: 01 jul. 2012.

[36] Milanes, A. et al. **Introdução ao MATLAB e ao pacote GAAL.** Disponível em: <http://www.mat.ufmg.br/gaal/>. Acesso em: 19 de ago. 2012.

[37] Miranda, A. T.; Albuquerque, M. P. **Introdução ao Matlab.** Centro Brasileiro de Pesquisas Físicas - CBPF. Disponível em: < www.cbpf.br/>. Acesso: 05 de ago. 2012.

[38] Miyazawa, F. K. **Algoritmos de Empacotamento Tridimensional: novas estratégias e análises de desempenho.** Master's thesis, Universidade de São Paulo, São Paulo, 1993.

[39] Oro, N. T.; Beckel, C. C.; Emmendorfer, L. R. **Heurística para o posicionamento de elipses tangentes no interior de um polígono de n lados.** V Conferência

Sul em Modelagem Computacional - Anais da V Conferência Sul em Modelagem Computacional VMCSul, Rio Grande. 2012. v. 1. p. 289-295.

[40] Petker, Jared. **Point-in-Polygon Detection.** University of California, Merced, 2010.

[41] Poole, D. **Álgebra linear.** São Paulo, SP: Thomson Learning: Cengage Learning, 2004. xxvi, 690 p. ISBN 85- 221-0359-3

[42] Ruggiero, M. A. G., Rocha, V. L. **Cálculo numérico: aspectos teóricos e computacionais**. S ao Paulo : Makron Books, 1988.

[43] Silva, I.N.; Cherri, A. C.; Junior, D. J. A. **Inferência *fuzzy* para o problema de corte de estoque com sobras aproveitáveis de material.** Revista Brasileira de Pesquisa Operacional, Rio de Janeiro, v. 31, n. 1, 2011.

[44] Spadotto, A. F. **Desenvolvimento de programa computacional aplicado ao empacotamento do palhiço de cana-de-açúcar.** Dissertação (Mestrado em Agronomia) - Faculdade de Ciências Agronômicas, Universidade Estadual Paulista Julio de Mesquita Filho, São Paulo. 2008.

[45] Steinbruch, A.; Winterle, P. **Álgebra Linear.** 2ª Ed. São Paulo: Pearson Makron Books, 1995.

[46] Steinbruch, A.; Winterle, P. **Geometria Analítica.** São Paulo: Pearson Makron Books, 2000.

[47] Telló, M. (Org.). **Aterramento elétrico:** impulsivo, em baixa e alta freqüências - com apresentação de casos. [livro]. Disponível em: <http://books.google.com.br/books?id= Tefs6meL6-YC>. Acesso em: 12 jun. 2012.

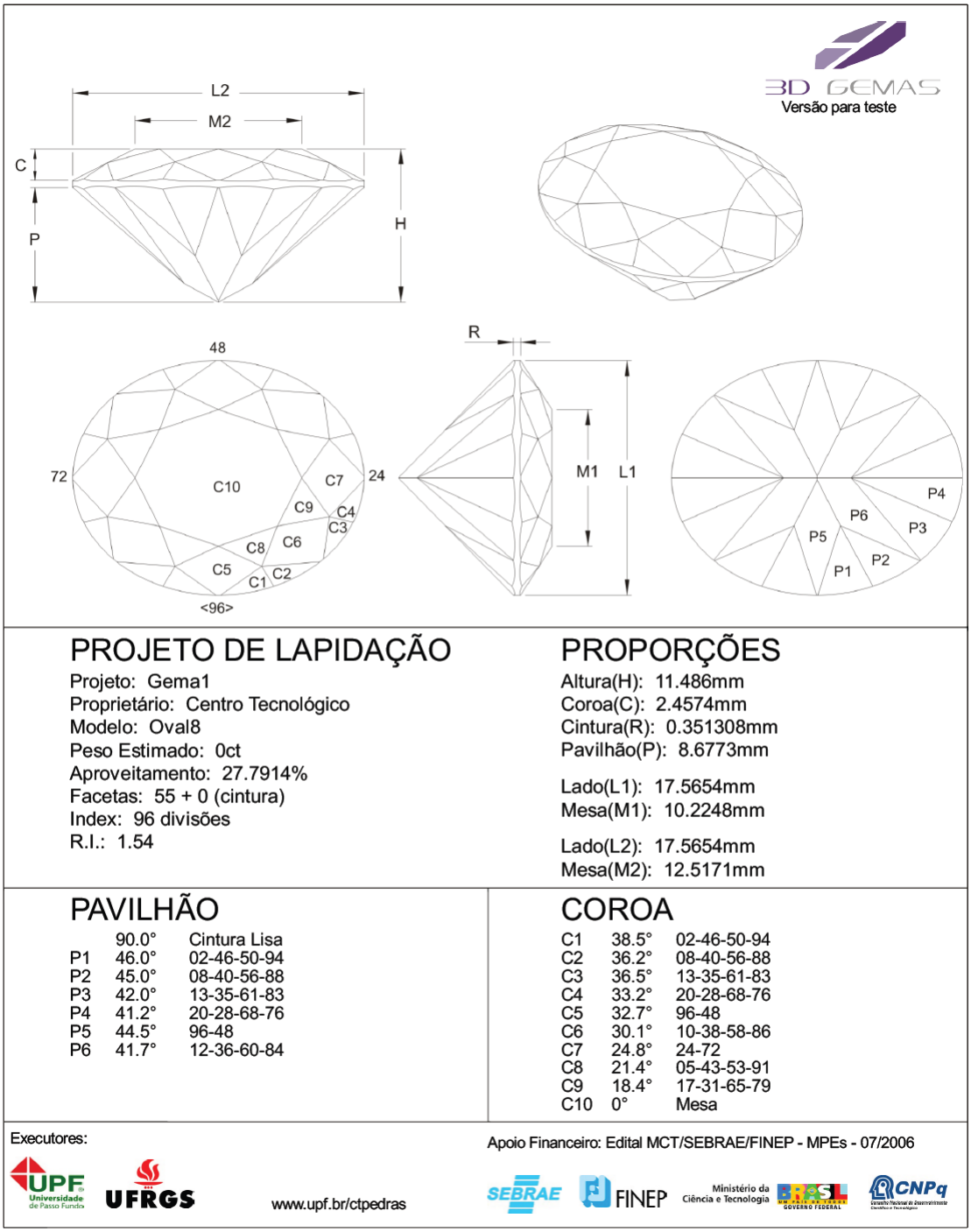
[48] Winterfeld, A. **Large-scale semi-infinite optimization applied to industrial gemstone cutting.** PhD Thesis, Technical University of Kaiserslautern, 2007, 155 p.

[49] Xavier, E. C. **Algoritmos para Problemas de Empacotamento.** Tese (Doutorado em Ciências da Computação) - Instituto de Computação, Universidade Estadual de Campinas, São Paulo. 2006.

[50] Zaro, M. A.; Borchardt, I. G. **Instrumentação:** guia de aulas práticas, Ed. UFRGS, 1982.

Apêndices

A Projeto de lapidação virtual



PROJETO DE LAPIDAÇÃO

Projeto: Gema1
Proprietário: Centro Tecnológico
Modelo: Oval8
Peso Estimado: 0ct
Aproveitamento: 27.7914%
Facetas: 55 + 0 (cintura)
Index: 96 divisões
R.I.: 1.54

PROPORÇÕES

Altura(H): 11.486mm
Coroa(C): 2.4574mm
Cintura(R): 0.351308mm
Pavilhão(P): 8.6773mm

Lado(L1): 17.5654mm
Mesa(M1): 10.2248mm

Lado(L2): 17.5654mm
Mesa(M2): 12.5171mm

PAVILHÃO

	90.0°	Cintura Lisa
P1	46.0°	02-46-50-94
P2	45.0°	08-40-56-88
P3	42.0°	13-35-61-83
P4	41.2°	20-28-68-76
P5	44.5°	96-48
P6	41.7°	12-36-60-84

COROA

C1	38.5°	02-46-50-94
C2	36.2°	08-40-56-88
C3	36.5°	13-35-61-83
C4	33.2°	20-28-68-76
C5	32.7°	96-48
C6	30.1°	10-38-58-86
C7	24.8°	24-72
C8	21.4°	05-43-53-91
C9	18.4°	17-31-65-79
C10	0°	Mesa

Executores:

UPF Universidade de Passo Fundo — UFRGS — www.upf.br/ctpedras

Apoio Financeiro: Edital MCT/SEBRAE/FINEP - MPEs - 07/2006

SEBRAE — FINEP — Ministério da Ciência e Tecnologia — BRASIL GOVERNO FEDERAL — CNPq

3D GEMAS
Versão para teste

PRÉ-FORMA

Projeto: Gema1
Proprietário: Centro Tecnológico
Modelo: Oval8
Identificação: ametista
Peso Gema Bruta: g
Ângulo Coroa: 41.0°
Ângulo Pavilhão: 48.0°

PROPORÇÕES PRÉ-FORMA

Altura(H): 11.1347mm
Coroa(C): 2.4574mm
Pavilhão(P): 8.6773mm
Cintura(R): 0.351308mm

Lado(L1): 17.5654mm
Lado(L2): 17.5654mm

Executores:
UPF Universidade de Passo Fundo
UFRGS
www.upf.br/ctpedras

Apoio Financeiro: Edital MCT/SEBRAE/FINEP - MPEs - 07/2006
SEBRAE FINEP Ministério da Ciência e Tecnologia BRASIL GOVERNO FEDERAL CNPq

B Pseudo-código do Algoritmo Heuristic

Algoritmo Heuristic

1. Início do algoritmo
2. Heuristic = **ler** função busca_unidimensional
3. centro_poligono
4. dist_ponto_poligono
5. elipse_interior_poligono
6. hjeeves_simula
7. maior_elipse_hjeeves
8. plota_elipse
9. plota_poligono
10. calcula_area_poligonal_2d
11. calcula_area_tri2d
12. **parametros** = $[vx, aw, x_c, y_c, \sigma, exc, tol, n, maxiter]$
13. **Para** Elipse Inicial **calcular** bw, area_elipse
14. **Defina** $x = xc$, $y = yc$, $a = aw$, $b = bw$, $rot = -\sigma$, $s = \sin(rot)$, $c = \cos(rot)$
15. **Calcula** vertices A1, B1, C1, D1
16.
17. % Elipses tangentes aos vertices Elipse Inicial
18. % Elipse 1 - Vertice B1
19. **Calcula** $(d1_x, d1_y), \lambda_1, aw1, bw1, (C1_x, C1_y)$
20. **calcula** vertices
21. plota_elipse
22. **calcula** pontos aleatorios PA1, PA2

23. **gera** matriz E1
24. % Verifica empacotamento elipse 1
25. **assuma** tamanho_1 = dimensao da matriz pontos exteriores ao poligono
26. **se** tamanho_1 $>$ 0
27. **calcula** $aw1 = aw1 \cdot (1 - \alpha)$
28. **repita** o procedimento **enquanto** tamanho_1 $>$ 0
29. **se** tamanho_1 = 0
30. **plota** elipse 1
31. **fim** do empacotamento elipse 1
32.
33. % Elipse 2 - Vertice C1
34. **Calcula** $(d2_x, d2_y), \lambda_2, aw2, bw2, (C2_x, C2_y)$
35. **calcula** vertices
36. plota_elipse
37. **calcula** pontos aleatorios PA3, PA4
38. **gera** matriz E2
39. % Verifica empacotamento elipse 2
40. **assuma** tamanho_2 = dimensao da matriz pontos exteriores ao poligono
41. **se** tamanho_2 $>$ 0
42. **calcula** $aw2 = aw2 \cdot (1 - \alpha)$
43. **repita** o procedimento **enquanto** tamanho_2 $>$ 0
44. **se** tamanho_2 = 0
45. **plota** elipse 2
46. **fim** do empacotamento elipse 2
47.
48. % Elipse 3 - Vertice A1
49. **Calcula** $(d3_x, d3_y), \lambda_3, aw3, bw3, (C3_x, C3_y)$
50. **calcula** vertices

51. plota_elipse
52. **calcula** pontos aleatorios PA5, PA6
53. **gera** matriz E3
54. % Verifica empacotamento elipse 3
55. **assuma** tamanho_3 = dimensao da matriz pontos exteriores ao poligono
56. **se** tamanho_3 > 0
57. **calcula** $aw3 = aw3 \cdot (1 - \alpha)$
58. **repita** o procedimento **enquanto** tamanho_3 > 0
59. **se** tamanho_3 = 0
60. **plota** elipse 3
61. **fim** do empacotamento elipse 3
62.
63. % Elipse 4 - Vertice D1
64. **Calcula** $(d4_x, d4_y), \lambda_4, aw4, bw4, (C4_x, C4_y)$
65. **calcula** vertices
66. plota_elipse
67. **calcula** pontos aleatorios PA7, PA8
68. **gera** matriz E4
69. % Verifica empacotamento elipse 4
70. **assuma** tamanho_4 = dimensao da matriz pontos exteriores ao poligono
71. **se** tamanho_4 > 0
72. **calcula** $aw4 = aw4 \cdot (1 - \alpha)$
73. **repita** o procedimento **enquanto** tamanho_4 > 0
74. **se** tamanho_4 = 0
75. **plota** elipse 4
76. **fim** do empacotamento elipse 4
77.
78. % Elipses tangentes ao quadrantes Elipse Inicial

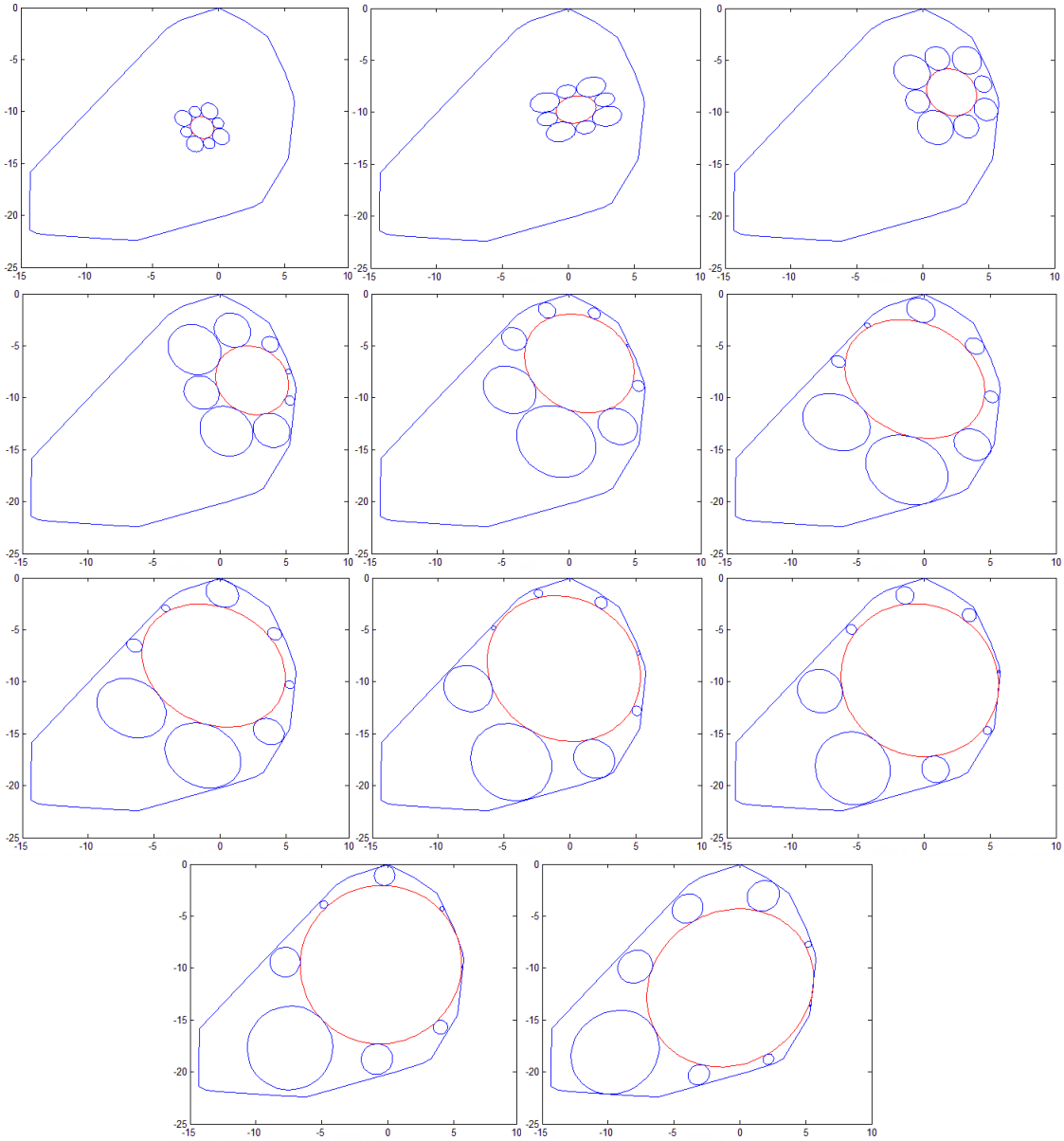
79. % Elipse 5 - tangente 1º quadante
80. **Calcula** ponto de tangência $P5(P5_x, P5_y)$
81. **Calcula** $(d5_x, d5_y), \lambda_5, aw5, bw5, (C5_x, C5_y)$
82. **calcula** vertices
83. plota_elipse
84. **calcula** ponto aleatorio BA5
85. **gera** matriz E5
86. % Verifica empacotamento elipse 5
87. **assuma** tamanho_5 = dimensao da matriz pontos exteriores ao poligono
88. **se** tamanho_5 > 0
89. **calcula** $aw5 = aw5 \cdot (1 - \alpha)$
90. **repita** o procedimento **enquanto** tamanho_5 > 0
91. **se** tamanho_5 = 0
92. **plota** elipse 5
93. **fim** do empacotamento elipse 5
94.
95. % Elipse 6 - tangente 2º quadante
96. **Calcula** ponto de tangência $P6(P6_x, P6_y)$
97. **Calcula** $(d6_x, d6_y), \lambda_6, aw6, bw6, (C6_x, C6_y)$
98. **calcula** vertices
99. plota_elipse
100. **calcula** ponto aleatorio BA6
101. **gera** matriz E6
102. % Verifica empacotamento elipse 6
103. **assuma** tamanho_6 = dimensao da matriz pontos exteriores ao poligono
104. **se** tamanho_6 > 0
105. **calcula** $aw6 = aw6 \cdot (1 - \alpha)$
106. **repita** o procedimento **enquanto** tamanho_6 > 0

107. **se** tamanho_6 = 0
108. **plota** elipse 6
109. **fim** do empacotamento elipse 6
110.
111. % Elipse 7 - tangente 3º quadante
112. **Calcula** ponto de tangência $P7(P7_x, P7_y)$
113. **Calcula** $(d7_x, d7_y), \lambda_7, aw7, bw7, (C7_x, C7_y)$
114. **calcula** vertices
115. plota_elipse
116. **calcula** ponto aleatorio BA7
117. **gera** matriz E7
118. % Verifica empacotamento elipse 7
119. **assuma** tamanho_7 = dimensao da matriz pontos exteriores ao poligono
120. **se** tamanho_7 > 0
121. **calcula** $aw7 = aw7 \cdot (1 - \alpha)$
122. **repita** o procedimento **enquanto** tamanho_7 > 0
123. **se** tamanho_7 = 0
124. **plota** elipse 7
125. **fim** do empacotamento elipse 7
126.
127. % Elipse 8 - tangente 4º quadante
128. **Calcula** ponto de tangência $P8(P8_x, P8_y)$
129. **Calcula** $(d8_x, d8_y), \lambda_8, aw8, bw8, (C8_x, C8_y)$
130. **calcula** vertices
131. plota_elipse
132. **calcula** ponto aleatorio BA8
133. **gera** matriz E8
134. % Verifica empacotamento elipse 8

135. **assuma** tamanho_8 = dimensao da matriz pontos exteriores ao poligono
136. **se** tamanho_8 > 0
137. **calcula** $aw8 = aw8 \cdot (1 - \alpha)$
138. **repita** o procedimento **enquanto** tamanho_8 > 0
139. **se** tamanho_8 $= 0$
140. **plota** elipse 8
141. **fim** do empacotamento elipse 8
142. **print** parametros
143. **Fim para**

C Algoritmo de ajuste de curva - Exponencial

03/01/13 11:45 C:\Users\Cássia Cris Beckel\Documents\Ajust...\ajuste_curva.m 1 of 1

```
% Ajuste de curvas - exponencial
% Cássia Cris Beckel

% phi(x)=alfa1+exp(-alfa2*x)

set (gcf, 'Color', 'white')

X =input('entre com os valores de x: ')
Y =input('entre com os valores de y: ')
n = length(X) %comprimento de X.
Z = log(Y)

% construção da matriz A

A(1,1)=n
A(1,2)=sum(X) %soma todo o vetor X
A(2,1)=A(1,2)
A(2,2)=sum(X.^2) % soma dos quadrados de cada elemento de X

% construção do vetor independente

b(1,1)=sum(Z)
b(2,1)=sum(Z.*X)

% solução do sistema A*alfa=b

gama=inv(A)*b

% calculo dos alfas
alfa(1)=exp(gama(1))
alfa(2)=-gama(2)

% grafico da função exponencial
x1=min(X)-0.1:0.001:max(X)+0.1;
phix=alfa(1)*exp(-alfa(2)*x1);

% subplot (1,2,1), plot(x1,phix),grid;
% subplot (1,2,2), plot(X,Y,'+r'),grid;

plot(x1,phix);
hold on;

plot(X,Y,'+r'),grid;
Xlabel('Percentual de área restante')
Ylabel('Tempo de processamento (s)')

% calculo dos valores ajustados

phi=polyval(phix,X); %valores ajustados
erro=sum((Y-phix).^2)

% previsões

%Xpre=[0.1 0.05 0.01];
%Ypre=polyval(C,Xpre)
```

D Simulação - Polígono com 23 vértices