

**Volume 79, 2015**

**Editores**

**Fernando Rodrigo Rafaeli (Editor Chefe)**

Universidade Federal de Uberlândia - UFU  
Uberlândia, MG, Brasil

**Vanessa Avansini Botta Pirani (Editor Adjunto)**

Universidade Estadual Paulista - UNESP  
Presidente Prudente, SP, Brasil

**Alexandre Loureiro Madureira**

Laboratório Nacional de Computação Científica - LNCC  
Petrópolis, RJ, Brasil

**Edson Luiz Cataldo Ferreira**

Universidade Federal Fluminense - UFF  
Niterói, RJ, Brasil

**Jorge Manuel Vieira Capela**

Universidade Estadual Paulista - UNESP  
Araraquara, SP, Brasil

**Sandra Augusta Santos**

Universidade Estadual de Campinas - UNICAMP  
Campinas, SP, Brasil

A Sociedade Brasileira de Matemática Aplicada e Computacional - SBMAC publica, desde as primeiras edições do evento, monografias dos cursos que são ministrados nos CNMAC.

Para a comemoração dos 25 anos da SBMAC, que ocorreu durante o XXVI CNMAC em 2003, foi criada a série **Notas em Matemática Aplicada** para publicar as monografias dos minicursos ministrados nos CNMAC, o que permaneceu até o XXXIII CNMAC em 2010.

A partir de 2011, a série passa a publicar, também, livros nas áreas de interesse da SBMAC. Os autores que submeterem textos à série Notas em Matemática Aplicada devem estar cientes de que poderão ser convidados a ministrarem minicursos nos eventos patrocinados pela SBMAC, em especial nos CNMAC, sobre assunto a que se refere o texto.

O livro deve ser preparado em **Latex (compatível com o Miktex versão 2.9)**, as **figuras em eps** e deve ter entre **80 e 150 páginas**. O texto deve ser redigido de forma clara, acompanhado de uma excelente revisão bibliográfica e de **exercícios de verificação de aprendizagem** ao final de cada capítulo.

Veja todos os títulos publicados nesta série na página  
[http://www.sbmac.org.br/p\\_notas.php](http://www.sbmac.org.br/p_notas.php)

# INTRODUÇÃO À GERAÇÃO DE MALHAS TRIANGULARES

Sanderson L. Gonzaga de Oliveira  
sanderson@dcc.ufla.br

Departamento de Ciência da Computação  
Universidade Federal de Lavras



Sociedade Brasileira de Matemática Aplicada e Computacional

São Carlos - SP, Brasil  
2015

Coordenação Editorial: Maria do Socorro Nogueira Rangel

Coordenação Editorial da Série: Fernando Rodrigo Rafaeli

Editora: SBMAC

Capa: Matheus Botossi Trindade

Patrocínio: SBMAC

Copyright ©2015 by Sanderson L. Gonzaga de Oliveira. Direitos reservados, 2015 pela SBMAC. A publicação nesta série não impede o autor de publicar parte ou a totalidade da obra por outra editora, em qualquer meio, desde que faça citação à edição original.

**Catálogo elaborado pela Biblioteca do IBILCE/UNESP**  
**Bibliotecária: Maria Luiza Fernandes Jardim Froner**

Gonzaga de Oliveira, Sanderson L.

Introdução à geração de malhas triangulares - São Carlos, SP :  
SBMAC, 2015, 94 p., 21.5 cm - (Notas em Matemática  
Aplicada; v. 79)

e-ISBN 978-85-8215-066-5

1. Geração de malhas 2. Triangulação de Delaunay 3. Diagrama de Voronoi  
I. Gonzaga de Oliveira, Sanderson L. II. Título. III. Série

CDD - 51

A minha esposa, Tatiana,  
ao meu pai, Luiz e  
à memória de minha mãe, Selma.  
*Dedico*



# Agradecimentos

Agradeço ao comitê responsável pela série Notas em Matemática Aplicada da SBMAC a oportunidade de divulgação deste trabalho. Agradeço também aos revisores anônimos as valiosas contribuições.

Agradeço a todos os alunos que contribuíram para que esta obra pudesse ser escrita, e também por criarem e fornecerem figuras, principalmente a Guilherme Oliveira Chagas, Frederico Santos de Oliveira e Jéssica Renata Nogueira. Também agradeço as contribuições em descrições aos alunos Conrado J. Teixeira (da seção 3.4 à seção 3.8), e Jéssica R. Nogueira (da subseção 4.2.1 à subseção 4.2.4 e subseção 4.2.8). Ainda, agradeço o apoio da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (Fapemig).





# Conteúdo

<b>Prefácio</b>	<b>xiii</b>
<b>1 Malhas</b>	<b>1</b>
1.1 Introdução . . . . .	1
1.2 Conceitos básicos sobre malhas . . . . .	2
1.3 Geração de malhas . . . . .	3
1.4 Abordagens de métodos de discretização . . . . .	4
1.5 Conectividades . . . . .	6
1.5.1 Malhas uniformes, não uniformes e conformes . . . . .	6
1.5.2 Topologia de malhas . . . . .	8
1.6 Triangulações . . . . .	10
1.7 Notas bibliográficas . . . . .	14
1.8 Exercícios . . . . .	15
<b>2 Técnicas para adaptação de malhas</b>	<b>17</b>
2.1 Introdução . . . . .	17
2.2 Refinamento e simplificação locais . . . . .	18
2.3 Movimento de um número fixo de pontos . . . . .	23
2.3.1 Características básicas de movimento de malhas . . . . .	23
2.3.2 Suavização laplaciana . . . . .	24
2.4 Aumento na ordem dos polinômios das funções de forma . . . . .	26
2.5 Outros tipos de adaptação de malhas . . . . .	26
2.6 Notas bibliográficas . . . . .	27
2.7 Exercícios . . . . .	27
<b>3 Métricas de avaliação da qualidade de triângulos</b>	<b>29</b>
3.1 Introdução . . . . .	29
3.2 Notações e conceitos básicos . . . . .	30
3.3 Métricas relacionadas com a área do politopo . . . . .	31
3.4 Ângulos extremos . . . . .	32
3.5 Razões entre raios das circunferências circunscrita e inscrita . . . . .	33
3.6 Razões entre arestas . . . . .	34
3.7 Razões entre raio do circuncírculo e arestas . . . . .	35
3.8 Razões entre arestas e raio de círculo inscrito . . . . .	35
3.9 Considerações finais . . . . .	36
3.10 Notas bibliográficas . . . . .	37
3.11 Exercícios . . . . .	37

<b>4</b>	<b>Triangulação de Delaunay e diagrama de Voronoi</b>	<b>39</b>
4.1	Introdução . . . . .	39
4.2	Triangulação de Delaunay . . . . .	40
4.2.1	Circuncírculos vazios . . . . .	41
4.2.2	Arestas locais de Delaunay . . . . .	43
4.2.3	Propriedades . . . . .	45
4.2.4	Triangulação de Delaunay restrita . . . . .	47
4.2.5	Algoritmos para a triangulação de Delaunay . . . . .	48
4.2.6	Refinamento de Delaunay . . . . .	52
4.2.7	Tetraedros com vértices quase coplanares . . . . .	59
4.2.8	Triangulação de Pitteway e grafo de Gabriel . . . . .	59
4.3	Diagrama de Voronoi . . . . .	60
4.3.1	Dualidade entre diagrama de Voronoi e tesselação de Delaunay . . . . .	61
4.3.2	Diagrama de potência . . . . .	62
4.4	Notas bibliográficas . . . . .	64
4.5	Exercícios . . . . .	64
<b>5</b>	<b>Representações de malhas triangulares</b>	<b>65</b>
5.1	Introdução . . . . .	65
5.2	Formas básicas de armazenamento . . . . .	65
5.2.1	Matriz de adjacências e listas de adjacências . . . . .	66
5.2.2	Formato CSR e variações . . . . .	66
5.3	Estrutura de dados <i>winged-edge</i> . . . . .	68
5.4	Estrutura de dados DCEL . . . . .	69
5.5	Grafo para representar refinamento adaptativo . . . . .	70
5.5.1	Simplificação local . . . . .	73
5.5.2	Ordenação dos triângulos da malha . . . . .	75
5.6	Custos de armazenamento . . . . .	78
5.7	Notas bibliográficas . . . . .	79
5.8	Exercícios . . . . .	79





# Prefácio

A geração de malhas é utilizada em diversas áreas, tais como computação gráfica, projetos industriais, aplicações médicas, modelagem de meio porosos, modelagem de objetos deformáveis, modelagem molecular, vídeo games, indústria de filmes, processamento de imagens, sistemas de informação geográfica e análise de propagação de rádio. A geração de malhas é particularmente importante na resolução numérica de equações diferenciais parciais pelo método dos volumes finitos. Malhas também são utilizadas em discretizações por elementos finitos, que é uma ferramenta central em computação científica.

Como um tópico da geometria computacional, a geração de malhas também é interdisciplinar. Por isso, profissionais e estudantes de diferentes áreas do conhecimento podem considerar este texto útil para obterem conhecimentos sobre termos e conceitos básicos em geração de malhas, com enfoque em malhas triangulares.

Um dos objetivos buscados ao escrever este livro foi fazer, dentro do possível, com que o leitor não necessite consultar outras fontes para entender os conteúdos descritos. Por isso, não são necessários conhecimentos aprofundados em assuntos específicos para entendimento dos conceitos abordados; apesar de que conhecimentos básicos de geometria, de geometria computacional e de métodos numéricos poderão contribuir para o entendimento dos conteúdos. Com isso, graduandos a partir da segunda metade do curso em diante ou que tenham cursado disciplinas básicas de Geometria Analítica, Cálculo Numérico e Projeto e Análise de Algoritmos poderão acompanhar sem dificuldades os conteúdos descritos. Mesmo assim, como muitos termos básicos são explicados no texto, os assuntos podem ser compreendidos mesmo por quem não tenha cursado ainda essas disciplinas. Logo, a inclusão dessas explicações podem ser úteis a leitores menos experientes; embora esses trechos possam ser considerados triviais por leitores com alguma familiaridade em geração de malhas. Por outro lado, desenvolvimentos recentes em geração de malhas triangulares são abordados. Dessa forma, leitores mais experientes em geração de malhas poderão considerar o texto útil também. Além disso, após a apresentação de conceitos básicos em geração de malhas triangulares, fontes são citadas para que o leitor possa obter conhecimentos em tópicos específicos, de forma atualizada, mais aprofundada e mais abrangente do que descrito neste texto.

Este texto foi dividido em cinco capítulos. Naturalmente, são abordados conceitos básicos sobre malhas no primeiro capítulo deste texto. Como continuação do capítulo introdutório, técnicas para adaptação de malhas são descritas no capítulo 2. Métricas para avaliação da qualidade de triângulos são abordadas no capítulo 3. A triangulação de Delaunay e o diagrama de Voronoi são descritos no capítulo 4. Finalmente, representações de malhas triangulares são apresentadas no capítulo 5.

Como descrito, o livro não esgota o assunto em geração de malhas triangulares e, como indicado já em seu título, é apenas uma introdução. Desde já, agradecemos as sugestões para a melhoria do texto que venham a ser encaminhadas para

xiv

sanderson@dcc.ufla.br.

Lavras, 19 de agosto de 2015.

Sanderson L. Gonzaga de Oliveira







# Capítulo 1

## Malhas

### 1.1 Introdução

Malhas são utilizadas em várias aplicações em diversas áreas da Ciência e da Engenharia: computação gráfica, projetos industriais, aplicações médicas, modelagem de meio porosos, modelagem de objetos deformáveis, modelagem molecular, vídeo games, indústria de filmes, processamento de imagens, sistemas de informação geográfica e análise de propagação de rádio são apenas algumas das áreas em que malhas são utilizadas. Malhas também são utilizadas em discretizações por elementos finitos, que é uma ferramenta central em computação científica. Em mais detalhes, com determinados métodos numéricos, equações diferenciais ou integro-diferenciais são resolvidas ao se substituir as derivadas ou integrais por expressões algébricas que envolvem a função incógnita. Isso significa que uma forma de se resolver as equações é *discretizar* as equações do modelo matemático, isto é, realiza-se uma aproximação da solução do problema por meio de um sistema de equações lineares que envolve um número finito de incógnitas. Dessa forma, as equações matemáticas são convertidas em formas adequadas para computação digital. Isso é chamado de discretização por envolver a criação de um modelo discreto, em que a solução é uma aproximação da solução do modelo contínuo original. Com isso, a solução aproximada de equações diferenciais parciais por um método de discretização, como os métodos dos elementos ou dos volumes finitos, requer, primeiro, a discretização das equações para substituir as equações diferenciais contínuas por um sistema de equações algébricas. A discretização é realizada pela conversão do domínio da solução do modelo matemático em um número finito de pontos. Esses pontos podem ser conectados entre si e essas conexões podem ser de diversas maneiras, de forma que geram subdomínios geometricamente simples e limitados. Esses pontos e suas conexões formam uma *malha computacional* e a geração dessa malha é o primeiro passo da computação. As soluções aproximadas nos vários subdomínios podem ser colocadas juntas para fornecerem uma solução aproximada do domínio computacional inteiro. O domínio computacional é onde o fenômeno físico em estudo ocorre e evolui. A discretização é, portanto, a substituição do modelo matemático (contínuo) por um conjunto finito de equações algébricas lineares.

Geometricamente, os subdomínios gerados na discretização são politopos, isto é, são primitivas geométricas no espaço euclidiano: intervalos no espaço unidimensional, polígonos no plano ou poliedros no espaço tridimensional. Em termos simples um *polígono* é uma figura plana limitada por segmentos de reta não alinhados na mesma reta e que se fecham. Por sua vez, *poliedro* é um sólido geométrico tri-

dimensional, cuja superfície é composta por um número finito de faces poligonais, formadas por três ou mais arestas. Um conjunto de politopos é uma *malha* (computacional). Isso significa que uma malha é um conjunto de pontos, e suas conexões, distribuídos ao longo de um campo de cálculo para a solução numérica de um conjunto de equações diferenciais parciais, ou seja, é a discretização geométrica do domínio de um problema. Esses politopos são chamados de células no método das diferenças finitas, de elementos no método dos elementos finitos ou de volumes no método dos volumes finitos. Especificamente, o termo volume é adimensional no contexto desse método.

Conceitos básicos sobre malhas são descritos no restante deste capítulo introdutório. Malha é definida na seção 1.2. A geração de malhas é abordada na seção 1.3. Na seção 1.4, são apresentados conceitos básicos sobre métodos de discretização. Na seção 1.5, são apresentados conceitos sobre conectividades e estruturas de dados. Triangulações são abordadas na seção 1.6.

Como continuação deste capítulo introdutório, técnicas básicas de adaptação de malhas são abordadas no capítulo 2. O foco deste texto é em malhas triangulares, isto é, malhas (bidimensionais) formadas por triângulos. Com isso, métricas de qualidade de triângulos são abordadas no capítulo 3. A triangulação de Delaunay e o seu dual geométrico, o diagrama de Voronoi, são explicados no capítulo 4. Finalmente, representações de malhas triangulares são apresentadas no capítulo 5.

## 1.2 Conceitos básicos sobre malhas

Em determinado sentido, malhas computacionais são grafos em que o aspecto geométrico é levado em consideração. Um grafo é um par  $G = (V, A)$ , em que  $V$  é um conjunto finito não vazio de objetos denominados vértices (ou pontos ou nós) e  $A$  é um conjunto de pares não ordenados de vértices de  $V$ , denominados como arestas [63]. O termo grafo é utilizado em áreas da matemática aplicada. Como o aspecto geométrico é relevante nas aplicações de métodos numéricos, o termo *malha* é comumente empregado no contexto de métodos numéricos. Com isso, é útil citar alguns termos básicos sobre grafos e malhas: arestas são *incidentes* a vértices e, se a aresta  $\overline{ab} \in A$ , então, o vértice  $a$  é *adjacente* ao vértice  $b$ . O *grau* (ou valência) de um vértice é o número de suas adjacências, ou seja, é o número de arestas incidentes ao vértice.

Para se ter uma descrição técnica de malha, é útil que antes sejam definidos alguns conceitos. A seguir, são definidas distância euclidiana entre dois pontos e bola fechada.

**Definição 1.1** (distância euclidiana entre dois pontos). *A distância euclidiana entre dois pontos  $p_1 = (x_1, x_2, \dots, x_d)$  e  $p_2 = (y_1, y_2, \dots, y_d)$ , no  $\mathbb{R}^d$ , define-se como  $\|p_1 - p_2\| = \sqrt{\sum_{i=1}^d (x_i - y_i)^2}$ .*

**Definição 1.2** (bola fechada). *No  $\mathbb{R}^d$ , o conjunto de todos os pontos, com centro no ponto  $c \in \mathbb{R}^d$ , com uma distância menor ou igual que o raio  $r > 0$ , é a bola fechada, definida como  $B_f(c, r) = \{p \in \mathbb{R}^d : \|p - c\| \leq r\}$ , no espaço euclidiano.*

Nesse contexto, instâncias de uma bola são aresta e disco nos espaços unidimensional e bidimensional, respectivamente. Como exemplo, se  $d = 2$ , então, pode-se escrever  $c = (x, y)$  e  $B_f(c, r) = \{(x', y') \in \mathbb{R}^2 : (x' - x)^2 + (y' - y)^2 \leq r^2\}$ . Com isso, pode-se, também, definir *bola aberta*.

**Definição 1.3** (bola aberta). No  $\mathbb{R}^d$ , o conjunto de todos os pontos, com centro no ponto  $c \in \mathbb{R}^d$ , com uma distância menor que o raio  $r > 0$ , é a bola aberta, definida como  $B_a(c, r) = \{p \in \mathbb{R}^d : \|p - c\| < r\}$ , no espaço euclidiano.

No plano euclidiano,  $B_a(c, r)$  é o *interior* de uma circunferência centrada no ponto  $c$  e de raio  $r$  ou, equivalentemente, o conjunto dos vetores no plano com origem em  $c$  e com raio menor que  $r$ . Com esses conceitos, pode-se, também, definir *fronteira* de uma bola.

**Definição 1.4** (fronteira). A *fronteira* de uma bola  $d$ -dimensional, com centro no ponto  $c \in \mathbb{R}^d$  e raio  $r > 0$ , é chamada de esfera  $(d-1)$ -dimensional e definida como  $S(c, r) = B_f(c, r) - B_a(c, r) = \{p \in \mathbb{R}^d : \|p - c\| = r\}$ , no espaço euclidiano.

Como exemplo, um círculo é uma esfera unidimensional. Com isso, os conceitos de domínios abertos e fechados podem ser também facilmente descritos. Em termos simples, compreende-se *domínio* como o conjunto de entrada de uma relação entre dois conjuntos.

**Definição 1.5** (domínios abertos e fechados). Sejam um ponto  $c \in \mathbb{R}^d$ , subconjuntos  $\Omega \subset \mathbb{R}^d$  e  $\Theta \subset \mathbb{R}^d$  centrados em  $c$ , e uma curva fechada  $\Phi$  seccionalmente contínua, em que  $\Phi = \Theta - \Omega$ , isto é,  $\Phi \not\subset \Omega$  e  $\Phi \subset \Theta$ . Os pontos da fronteira de  $\Theta$  podem ser definidos como  $\partial\Theta = \{p \in \mathbb{R}^d : p \notin \Omega \wedge p \in \Theta \wedge p \in \Phi\}$ . O subconjunto de pontos  $\Omega \subset \mathbb{R}^d$  é dito um domínio aberto se, dado um ponto  $c \in \Omega$ , então, existe um subconjunto aberto de pontos centrados em  $c$  inteiramente contidos em  $\Omega$ . Um subconjunto  $\Theta \subset \mathbb{R}^d$  é dito domínio fechado se seu complemento  $\mathbb{R}^d - \Theta = \{p \in \mathbb{R}^d : p \notin \Theta\}$  é aberto.

Com isso, em termos simples, um *domínio fechado* pode ser entendido como um conjunto fechado e conexo, isto é, que não pode ser particionado em subconjuntos. Ainda, *curva* é um mapeamento (uma função)  $\mathfrak{c} : \mathbb{R} \rightarrow \mathfrak{A}$ . Em termos simples, uma *curva* é um conjunto de pontos homeomórfico a uma linha. Homeomorfismo entre dois espaços topológicos  $\mathfrak{A}$  e  $\mathfrak{B}$  é um mapeamento  $\sigma : \mathfrak{A} \rightarrow \mathfrak{B}$  contínuo e com um mapeamento inverso  $\sigma^{-1} : \mathfrak{B} \rightarrow \mathfrak{A}$ . Em termos simples, homeomorfismo entre dois subconjuntos no espaço euclidiano indica que todos os pontos entre os dois espaços são abrangidos. Por sua vez, seja um par  $(\mathfrak{A}, \mathfrak{t}(\mathfrak{A}))$ . Escolhas diferentes da *topologia*  $\mathfrak{t}(\mathfrak{A})$  de um espaço (um conjunto)  $\mathfrak{A}$  corresponde a estruturas topológicas diferentes de  $\mathfrak{A}$ . Agora, considere a definição de malha fornecida por De Floriani, Kobbelt e Puppo [41].

**Definição 1.6** (malha). Considere um politopo convexo de dimensão  $k$  como um subconjunto do espaço euclidiano  $d$ -dimensional homeomórfico a uma bola de dimensão  $k$ , em que  $k \leq d$ . Seja  $M$  um conjunto finito de politopos de dimensões heterogêneas e embutidos no espaço euclidiano  $d$ -dimensional, em que  $d$  é a dimensão máxima dos politopos de  $M$ , tal que a fronteira de cada politopo em  $M$  forma uma coleção de politopos de dimensões menores, pertencentes a  $M$ . Com isso,  $M$  é uma malha de dimensão  $d$  se e, somente se, os interiores de cada par de politopos adjacentes, de dimensão  $d$  de  $M$ , são disjuntos e qualquer politopo  $k$  de  $M$ , com  $k < d$ , tem limite com pelo menos um politopo de dimensão  $d$  de  $M$ .

### 1.3 Geração de malhas

Procura-se encontrar uma divisão apropriada de um domínio contínuo no processo de *geração de malhas*. Na geração de uma malha adequada, leva-se em consideração o custo de geração da malha, a precisão da aproximação e o desempenho

da computação para obter a aproximação. Com isso, a malha deve ser gerada sob várias restrições, que podem ser difíceis de serem satisfeitas. Como exemplos, a malha deve ser suficientemente densa para que a aproximação numérica seja precisa e deve ser suficientemente refinada para se aproximar as alterações no gradiente da solução. Entretanto, a malha não pode ser tão densa de forma que a solução computacional tenha um custo computacional alto para ser obtida. Ainda, a diferença de espaçamento entre politopos adjacentes da malha deve ser suave. A malha deve ser gradual de forma que a diferença de área (ou volume) entre politopos não seja grande, pois isso pode acarretar em instabilidade numérica. Também, no processo de se alcançar a solução computacional, deve-se garantir uma continuidade adequada da solução nas interfaces de cada par de politopos adjacentes.

A malha também deve ser gerada com eficiência computacional. Isso porque a precisão de uma aproximação também pode ser prejudicada caso a escolha da malha conduza a códigos computacionais excessivamente complexos.

A geração da malha é um passo crítico em simulações numéricas porque a precisão e o desempenho da solução computacional, como em problemas que envolvem velocidade ou pressão de cisalhamento, são afetados pela resolução da malha, pela topologia da malha, isto é, malha regular (estruturada) ou irregular (não estruturada), abordadas na seção 1.5.2, na página 8, e pelo tipo de politopo, como triângulos ou quadriláteros em domínios bidimensionais, tetraedros, elementos piramidais, prismas ou hexaedros em domínios tridimensionais. Isso é atribuído à integração da equação diferencial parcial que descreve o problema físico no politopo. Além de serem utilizados na modelagem de domínios bidimensionais, triângulos e quadriláteros são utilizados também como malhas de superfície embutidas em domínios tridimensionais, que são bastante utilizadas em computação gráfica e em métodos de elementos de contorno, como exemplos. Malhas de triângulos ou de tetraedros são mais fáceis de serem geradas do que malhas de quadriláteros ou de hexaedros. Por outro lado, malhas de quadriláteros ou de hexaedros oferecem mais precisão na interpolação e aproximação do que malhas de triângulos ou de tetraedros. Entretanto, malhas de hexaedros podem ser extremamente difíceis de serem geradas para domínios geometricamente complicados [32, p. 3].

O sistema de equações lineares resultante da discretização da equação diferencial parcial pode ser mal condicionado se o politopo da malha for de má qualidade. Ocorre que, em malhas com politopos de má qualidade, como exemplos, com ângulos próximos de zero ou  $\pi$ , a solução computacional sendo gerada durante as iterações muda rapidamente, isto é, é altamente sensível, em resposta a variações pequenas dos valores dos coeficientes e dados do sistema de equações lineares resultante da discretização da equação diferencial parcial. Conseqüentemente, erros de arredondamento podem levar a aproximações irreais ou à instabilidade numérica. Isso pode gerar aproximações inadequadas e o modelo numérico não é convergente se for instável.

## 1.4 Abordagens de métodos de discretização

Em *diferenças finitas*, as derivadas das equações diferenciais parciais são representadas nos pontos por expressões algébricas obtidas por meio da expansão da série de Taylor das variáveis da solução em vários pontos adjacentes ao ponto da avaliação. Isso equivale a representar a solução por meio de polinômios entre os pontos [143, seção P-1.1].

A abordagem por diferenças finitas, utilizando pontos discretos, está associ-

ada, historicamente, com malhas cartesianas quadrangulares. Essa estrutura fornece pontos distribuídos uniformemente na forma de quadrângulos e proporciona uma forma fácil de identificação dos pontos adjacentes a serem utilizados na representação das derivadas. O método das diferenças finitas não é, contudo, limitado a malhas quadrangulares e tem sido aplicado em outros sistemas de coordenadas analíticas (cilíndricas, esféricas, elípticas etc.) que, ainda assim, formam malhas regulares [143, seção P-2.1]. Optou-se por utilizar as expressões *malha regular* e *malha irregular* em vez de *malha estruturada* e *malha não estruturada*, respectivamente; pois, de certa forma, uma malha não estruturada possui uma determinada estrutura. Esses conceitos são abordados na seção 1.5.2, na página 8.

De acordo com Cheng, Dey e Shewchuk [32, p. 10], a geração de malhas irregulares para métodos de elementos finitos foi iniciada em 1970 com o artigo de Frederick, Wong e Edge [52]. Na seção P-2, Thompson, Soni e Weatherhill [143] explicam que a abordagem por elementos finitos foi utilizada pela natureza de sua construção em politopos. De forma geral, essa abordagem é considerada adequada para regiões irregulares. Uma malha com tais politopos pode ser gerada para preencher qualquer região arbitrária e cada politopo é uma entidade em si própria e a representação é em politopos, não entre politopos.

O esquema clássico no método dos elementos ou dos volumes finitos inclui os seguintes passos para um caso simples em que um sistema de equações lineares é resolvido.

1. Definição de um domínio computacional.
2. Construção da malha computacional.
3. Um passo de interpolação, em que os elementos finitos são construídos a partir dos politopos da malha.
4. Montagem de um sistema de equações lineares, baseada na conectividade entre os elementos.
5. Resolução do sistema de equações lineares.
6. Término do método se a precisão da aproximação é suficiente ou se o número de iterações ou um tempo limite é ultrapassado; ou adaptação da malha e retorno ao passo 2, caso contrário.

Em métodos de discretização, como os métodos dos elementos, diferenças ou volumes finitos, deve-se levar em consideração o tamanho da menor aresta na malha. Politopos com arestas muito pequenas podem causar instabilidade nos métodos de integração numérica utilizados em simulações dependentes do tempo. Isso significa que há de se levar em consideração a estabilidade do método numérico utilizado para resolver a equação discretizada. A integração numérica é tipicamente dependente da condição Courant-Friedrichs-Lewy (CFL) [33, 34, 35]. Essa condição tem a forma

$$C = \Delta t \sum_{i=1}^d \frac{u_{x_i}}{\Delta x_i} \leq C_{max}$$

no caso  $d$ -dimensional, em que  $C$  é o número de Courant,  $u_{x_i}$  são as magnitudes da velocidade e cujas dimensões são comprimento/tempo,  $\Delta t$  é o passo de tempo e  $\Delta x_i$  são os comprimentos do intervalo espacial. O valor  $C_{max}$  depende do método numérico utilizado, especialmente se a abordagem é explícita ou implícita. Se uma abordagem explícita (ou marchante) é utilizada, um valor típico é  $C_{max} = 1$ . Abordagens implícitas, em que se obtêm um sistema de equações lineares, são, geralmente, menos sensíveis à instabilidade numérica e um valor maior

de  $C_{max}$  pode ser utilizado. De forma intuitiva, a condição CFL implica que o passo de tempo deve ser pequeno de forma que o ponto de avaliação de um politopo no passo de tempo seguinte não ultrapasse o comprimento do politopo no passo de tempo corrente, para  $C_{max} \leq 1$ . Para  $C_{max} > 1$ , o ponto de avaliação poderá ultrapassar mais de um comprimento de politopo em um só passo de tempo e isso poderá causar instabilidade no método numérico utilizado. Com isso, o passo de tempo é limitado pelo tamanho das arestas mais curtas da malha. Estabelecer um passo de tempo muito pequeno pode acarretar em alto custo computacional em uma simulação.

## 1.5 Conectividades

Uma malha pode ser composta por politopos de diferentes formas. Dependendo da aplicação, os politopos devem satisfazer propriedades específicas.

Os politopos de uma malha são os seus componentes básicos e são definidos pela sua natureza geométrica e uma lista de vértices. Essa lista, se considerada com algumas convenções, permite a definição completa do politopo, incluindo a definição de suas arestas e faces (em malhas tridimensionais). Em relação à natureza geométrica, no plano euclidiano, politopos podem ser triângulos, quadriláteros etc.; no espaço euclidiano tridimensional, podem ser hexaedros, tetraedros e os menos comuns, prismas e elementos piramidais. De modo geral, prismas e elementos piramidais são menos utilizados porque apresentam dois tipos de faces, triângulos e quadriláteros, e isso pode gerar malhas não conformes.

As malhas podem ser classificadas, principalmente, de acordo com a sua *conectividade*, definida a seguir.

**Definição 1.7.** (*conectividade de um politopo*) *A conectividade de um politopo é a definição das ligações (conexões) entre os vértices ao nível do politopo.*

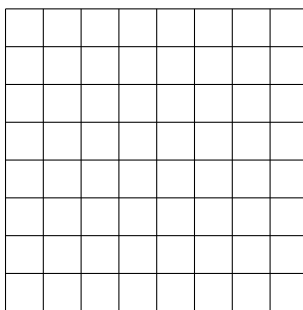
Essa conectividade possibilita a descrição da topologia desse politopo. Dessa forma, a topologia de um politopo de uma malha é a definição das relações entre seus vértices, arestas e faces.

A seguir, malhas uniformes, não uniformes e conformes são abordadas na subseção 1.5.1. Topologia de malhas são abordadas na subseção 1.5.2.

### 1.5.1 Malhas uniformes, não uniformes e conformes

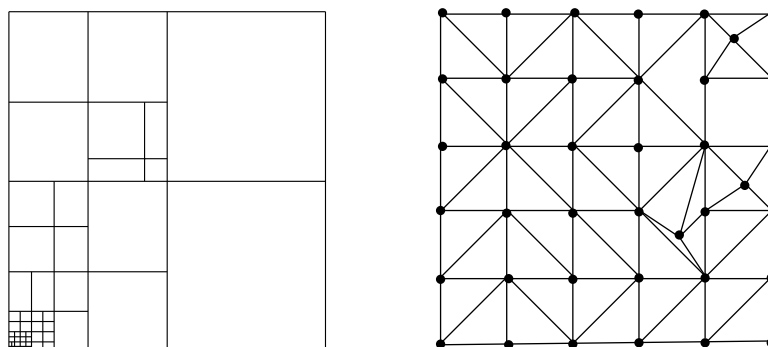
As malhas podem ser classificadas como uniformes ou não uniformes. Uma malha uniforme é, como o nome indica, constituída por vértices que são distribuídos de maneira uniforme no domínio. Veja um exemplo de malha uniforme com 64 polígonos na figura 1.1.

Uma malha uniforme pode ser pouco conveniente se a solução do modelo matemático varia muito entre os pontos na solução numérica. Uma forma de se resolver isso é, evidentemente, usar mais pontos, de modo que o espaçamento entre os pontos seja reduzido. No entanto, isso pode gerar grande esforço computacional, já que haverá mais pontos onde as equações devem ser avaliadas, ao se utilizar uma malha com espaçamento uniforme entre os pontos. Ainda, o aumento de pontos na malha pode ser demasiado se os pontos forem igualmente espaçados e fortes variações ocorrerem na solução em regiões variadas e dispersas do campo de cálculo, uma vez que muitos pontos serão desperdiçados em regiões de pequena variação da solução do modelo matemático. Uma alternativa natural é distribuir os pontos



**Figura 1.1:** Exemplo de malha uniforme, regular e conforme.

com espaçamentos diferentes. Com isso, gera-se uma malha com espaçamento não uniforme entre os pontos, isto é, os vértices da malha são distribuídos de maneira *não uniforme* no domínio [143, seção P-1.1]. As *malhas não uniformes* são caracterizadas pela ocorrência de refinamento adaptativo, que é descrito na subseção 2.2, na página 18. Exemplos de malhas não uniformes podem ser observados na figura 1.2.



**Figura 1.2:** Exemplos de malhas não uniformes, irregulares e não conformes. (Figuras adaptadas de [102].)

A estrutura de dados para representar a malha e a codificação podem ser mais difíceis de serem realizadas em uma malha não uniforme do que em uma malha uniforme. Para problemas com variações suaves na solução, é preferível utilizar uma malha uniforme em vez de uma malha não uniforme; pois, ao se utilizar uma malha uniforme, é possível obter uma solução tão eficiente quanto com uma malha não uniforme.

Uma malha é *conforme* se satisfaz duas condições: *i*) a interseção entre quaisquer dois polítopos da malha ou é um conjunto vazio ou é um elemento geométrico pertencente a ambos os polítopos (uma face, uma aresta, um vértice) e *ii*) tal elemento geométrico compartilhado é somente um e completo. Note que os vértices internos de uma malha conforme podem não ter a mesma valência. Um exemplo de malha conforme é mostrado na figura 1.1 e exemplos de malhas não conformes são mostrados na figura 1.2. Ainda, um malha de superfície (conforme) é um *manifold* se suas arestas internas são compartilhadas por exatamente dois componentes básicos da malha ou somente por um se a aresta pertence à fronteira de uma superfície

aberta.

### 1.5.2 Topologia de malhas

Malhas podem ser classificadas como regulares (ou estruturadas), irregulares (ou não estruturadas) e versões híbridas de malhas regulares e irregulares. A diferença básica entre malhas regulares e irregulares é na forma da *estrutura de dados* que descreve mais apropriadamente a malha. Em termos simples, estrutura de dados é a forma de armazenamento e organização dos dados na memória do sistema de computação.

Em geral, o desempenho da computação é em função da quantidade do número de pontos de avaliação no modelo numérico e da quantidade de adjacências entre os pontos. Com isso, busca-se um número mínimo de pontos de avaliação para a precisão desejada na aproximação. A relação entre o número de vértices e o número de politopos depende da topologia da malha e do tipo dos politopos. Se os pontos de avaliação das equações forem nos vértices, então, o número de vértices indica o número de *graus de liberdade* do modelo, ou seja, é o número de incógnitas do sistema de equações lineares gerado na aplicação do método numérico. Em um modelo mais genérico com  $m$  graus de liberdade por vértice, há  $m$  equações acopladas, produzidas para cada vértice. Claramente, a utilização de uma malha regular ou irregular depende da aplicação.

#### Malhas regulares

Uma malha regular é caracterizada por uma conectividade regular dos vértices e pode ser expressa como uma estrutura de dados matricial (bidimensional, tridimensional etc.). Em uma malha regular, a regularidade da conectividade da matriz que representa a malha permite percorrer os vértices adjacentes a um vértice apenas pelos índices da estrutura de dados e pode-se detectar as adjacências apenas ao se verificar o arranjo do armazenamento da estrutura de dados. Em particular, o mesmo número de politopos incide a cada vértice interior de uma malha regular. Mais especificamente, uma malha regular de quadrângulos ou de hexaedros consiste em um conjunto de coordenadas e conectividades de forma que a malha de polígonos ou de poliedros pode ser mapeada em uma matriz. Por exemplo, uma matriz tridimensional  $M_{i,j,k}$  pode ser utilizada para armazenar as coordenadas dos pontos de uma malha regular tridimensional, em que cada índice pode ser escolhido para descrever e representar a posição dos pontos em uma direção. Com isso, os pontos adjacentes ao ponto representado na entrada  $M_{i,j,k}$  são representados nas seguintes entradas. Respectivamente, as entradas  $M_{i-1,j,k}$ ,  $M_{i+1,j,k}$ ,  $M_{i,j+1,k}$ ,  $M_{i,j-1,k}$ ,  $M_{i,j,k-1}$  e  $M_{i,j,k+1}$  representam os pontos à esquerda, à direita, acima, abaixo, na frente e atrás do ponto representado na entrada  $M_{i,j,k}$ .

Em geral, somente quadrângulos e hexaedros são utilizados em malhas regulares. Exemplo de uma malha regular e conforme é mostrado na figura 1.1. Há autores que denominam malha regular como *grid*, mas há autores que podem utilizar o termo *grid* para se referirem a qualquer tipo de malha.

Thompson e Weatherhill [143, seção 1.2] explicam que malhas regulares fornecem uma representação natural das condições de contorno de derivadas normais e permitem aproximações simples baseadas nas direções predominantes do problema, por exemplo, paralelas ou normais a uma direção de fluxo ou limite. Malhas regulares e conformes têm sido amplamente aplicadas, por exemplo, em dinâmica de fluidos computacionais. Se o problema for em dinâmica de fluidos computacional e o



movimento do fluido é altamente direcional, a utilização de uma malha regular com poliedros bem organizados ao longo da direção principal do fluxo contribui para se obter soluções numéricas adequadas. Uma malha regular com hexaedros alinhados ao longo da linha central do fluxo apresenta arestas predominantemente alinhadas com o fluxo. Um bom alinhamento entre as arestas da malha e o fluxo, como em malhas regulares de hexaedros, apresenta baixos erros de difusividade numérica. Em malhas regulares com hexaedros alinhados longitudinalmente, pode-se “esticar” longitudinalmente os hexaedros sem degradar seus formatos, isto é, não há alteração dos ângulos entre as arestas do poliedro ao se gerar malhas anisotrópicas. Por sua vez, malhas de tetraedros não podem fornecer alinhamento de arestas com o fluxo [43, p. 39-40]. Grosso modo, malhas isotrópicas não são associadas a uma direção em particular e malhas anisotrópicas são associadas a uma direção em particular.

Uma malha regular também leva a uma estrutura simples do conjunto de dados e permite a utilização de representações de divisões e fluxo direcional de tempo. O processo de resolução de sistemas das equações lineares, que são gerados na aplicação de métodos numéricos, como exemplos, pelos métodos dos elementos finitos, das diferenças finitas ou dos volumes finitos, é simples e rápido em malhas regulares (bem como em malhas irregulares), devido à facilidade em determinar as adjacências dos vértices.

Uma das grandes vantagens resultantes da utilização de malhas regulares é que sua implementação é relativamente simples. A principal desvantagem da utilização de malhas regulares é o fato de que nem sempre é possível garantir que uma malha aceitável seja produzida com essa abordagem.

### Malhas irregulares

Uma malha irregular é caracterizada por uma conectividade irregular, isto é, as conectividades dos vértices não possuem uma estrutura regular em toda a malha. Uma malha irregular não pode ser diretamente expressa como uma estrutura de dados matricial (bidimensional, tridimensional etc.). O número de politopos incidentes a um vértice interior da malha não é necessariamente constante em uma malha irregular. Por isso, as adjacências de um vértice em uma malha irregular devem ser explicitamente armazenadas. Exemplos de malhas irregulares são mostrados na figura 1.2. Métodos de discretização de equações de fluxo de fluidos que são baseados em esquemas integrais, tais como volumes finitos ou elementos finitos, são candidatos naturais para serem utilizados com malhas irregulares [143, seção 35.1].

A principal vantagem da abordagem irregular é que fornece uma forma conveniente para a discretização de domínios geométricos com formatos complicados ou de fluxos com características complexas. Dependendo do problema, uma malha irregular pode permitir um número menor de politopos para cobrir o domínio em relação a uma malha regular.

Métodos que utilizam malhas irregulares, naturalmente, oferecem a possibilidade de incorporação de adaptações: uma malha irregular pode ser melhor adaptada para a representação do domínio do problema se comparada a uma malha regular. Em malhas irregulares, é possível adicionar e remover vértices e politopos como a geometria exigir ou em um esquema de adaptação de fluxo, como em fluxo de gradientes ou em erros evolutivos [143, seção P-3.1]. Por outro lado, os métodos empregados são bastante sensíveis à qualidade da malha empregada. Essa dependência é tão grande que a qualidade da malha tem que ser levada seriamente em consideração. Técnicas de adaptação de malhas são descritas no capítulo 2.

Por outro lado, a falta de quaisquer características direcionais globais em uma malha irregular torna a implementação do algoritmo de varredura dos pontos de avaliação mais difícil de implementar do que em malhas regulares. Ainda, malhas irregulares podem exigir um esquema de armazenamento mais complicado do que se malhas regulares forem utilizadas. Isso ocorre porque, como descrito, as adjacências nas malhas irregulares devem ser explicitamente armazenadas. Pode ser necessário armazenar uma referência para cada um dos vértices adjacentes em todos os vértices que compõem uma malha irregular. Ainda, para que não haja aumento de vértices da malha e, conseqüentemente, aumento do custo computacional da resolução, pode-se definir um movimento de vértices da malha para regiões com maior variação da solução do modelo matemático ou na direção de um sistema de fluxo unidirecional. Em geral, esse movimento de vértices “estica” longitudinalmente os politopos e pode degradar o formato dos politopos de malhas irregulares ao alterar os ângulos entre as arestas ou faces de um politopo. Por exemplo, em um problema de pressão de cisalhamento, De Santis [43, p. 44] comparou o desempenho de malhas regulares e irregulares. Esse autor mostrou que malhas irregulares de tetraedros e prismas exigiram como seis vezes mais poliedros e o desempenho da utilização da malha irregular foi 14 vezes pior em relação à utilização de malha regular de poliedros. Ainda, o erro da malha irregular não diminuiu abaixo de 5%; mas o erro diminuiu progressivamente com a utilização da malha regular. Com esse exemplo, tem-se uma noção de que a escolha da topologia da malha é altamente dependente da aplicação.

### Versões híbridas

Uma *malha semi-regular* é obtida a partir de um *refinamento adaptativo* dos politopos de uma malha inicialmente irregular. Refinamento adaptativo de malhas é descrito na subseção 2.2, na página 18. Não devem ser gerados novos politopos irregulares ao se refinar a malha. Portanto, uma malha semi-regular é regular em partes e caracteriza-se por possuir alguns politopos irregulares que constituíam a malha inicial [41].

Em particular, uma *malha regular multi-bloco* é uma coleção de malhas irregulares de blocos regulares. Mais especificamente, uma malha regular multi-bloco é composta de vários domínios e cada domínio é uma malha regular.

## 1.6 Triangulações

Triangulações são utilizadas para representar partes do espaço contínuo de uma forma que permita a algoritmos numéricos computar características do espaço. Há uma utilização intensa de malhas triangulares. Antes de se apresentar uma definição de triangulação, considere a definição de *simplex* a seguir.

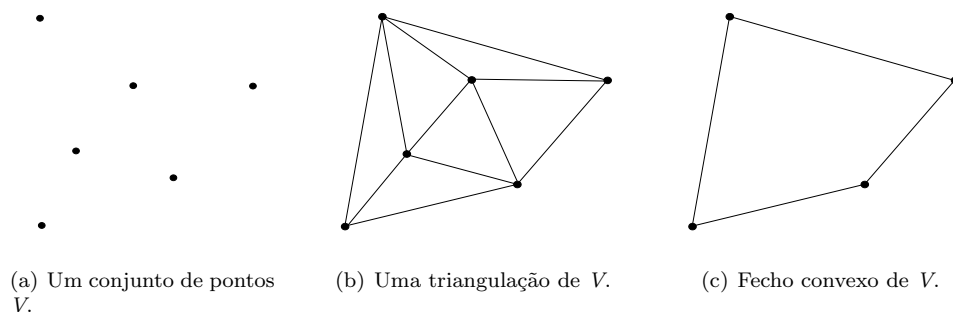
**Definição 1.8** (*simplex*). *Um simplex  $k$ -dimensional, embutido no espaço euclidiano  $d$ -dimensional, é constituído de pontos no espaço euclidiano  $k$ -dimensional que pode ser expresso como a combinação convexa de  $k + 1$  pontos afim-independentes.*

Os pontos  $a_1, a_2, \dots, a_n \in \mathbb{R}$  são afim-independentes se os  $k - 1$  pontos  $a_2 - a_1, \dots, a_k - a_1$  são linearmente independentes. Em especial, afim-independência é independência linear ao se desconsiderar a origem. Em particular, no espaço euclidiano, instâncias de um *simplex* são: vértice (*0-simplex*) no espaço zero-dimensional, aresta (*1-simplex*) no espaço unidimensional, triângulo (*2-simplex*) no plano e tetraedro (*3-simplex*) no espaço tridimensional.

Em uma malha de “simplexos”  $d$ -dimensionais, todo  $k$ -simplex, com  $k < d$ , é gerado por um subconjunto de vértices de algum  $d$ -simplex [41]. Triângulo é o polígono mais simples no plano euclidiano: triângulos são os polígonos fechados com o menor número de vértices e arestas. Em especial, *triângulo* pode ser descrito como o conjunto de todos os pontos internos ao polígono fechado e convexo formado por três arestas e *tetraedro* é um poliedro convexo composto por quatro faces triangulares, seis arestas, e cada três faces triangulares são incidentes a cada um dos quatro vértices do poliedro. Considere a definição de triangulação de um conjunto de pontos a seguir.

**Definição 1.9** (triangulação de um conjunto de pontos). *Seja um conjunto finito de pontos  $V$  no plano euclidiano. Uma triangulação  $T$  é um conjunto finito de simplexos (ou triângulos) no plano, tal que: i)  $V$  é o conjunto de vértices de  $T$ ; ii) para quaisquer dois triângulos  $t_1, t_2 \in T$ ,  $t_1 \cap t_2 = \emptyset$  ou a interseção de  $t_1$  e de  $t_2$ , ou seja,  $t_1 \cap t_2 = \overline{ab}$  é elemento do conjunto de arestas de  $T$ , ou  $t_1 \cap t_2 = a$  é elemento do conjunto  $V$ ; iii)  $((\overline{ab} = t_1 \cap t_2) \Rightarrow (t_1 \cup t_2 - \overline{ab} = \emptyset)) \wedge ((a = t_1 \cap t_2) \Rightarrow (t_1 \cup t_2 - a = \emptyset))$ , isto é, os interiores de cada par de triângulos adjacentes é vazio; iv) a união de todos os triângulos em  $T$  é a combinação convexa de  $V$ .*

Em termos simples, uma triangulação é a partição de um domínio fechado em triângulos. Veja um exemplo de triangulação na figura 1.3(b).

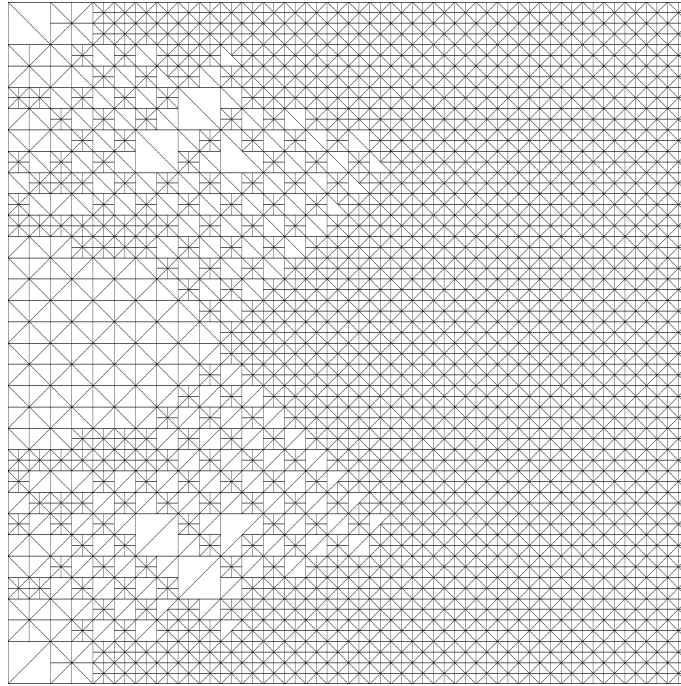


**Figura 1.3:** Pode-se obter uma triangulação a partir de um conjunto de pontos.

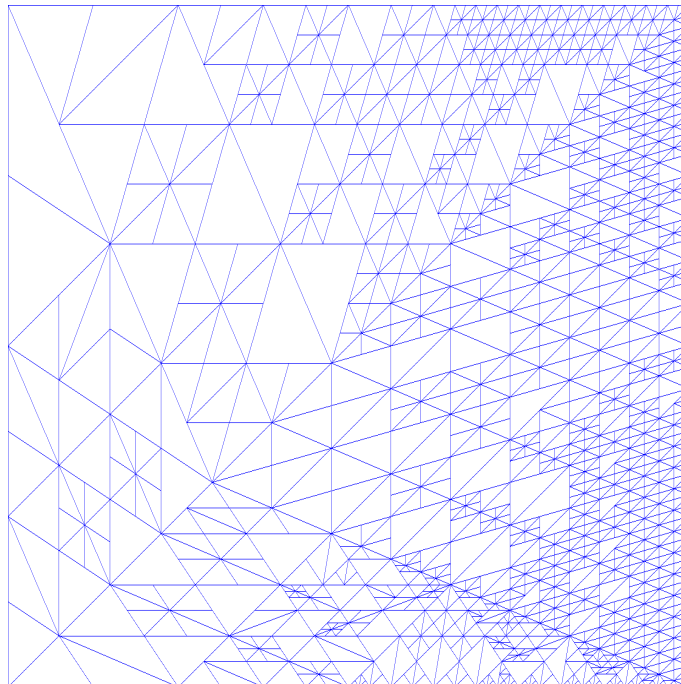
Em relação à conectividade e topologia de triângulos, uma numeração ordenada dos vértices permite calcular a área da superfície de um triângulo em sentido positivo, ou direcional. Essa numeração permite avaliar, também, as direções normais de cada aresta.

Pode-se gerar uma triangulação em qualquer conjunto de pontos. Se  $n$  pontos são colineares, então, existem  $n - 1$  arestas e zero triângulo (note que a definição 1.9 ainda é válida). Além disso, não existe uma só solução para a triangulação de um conjunto de pontos. Porém, deseja-se que os triângulos possuam formatos convenientes para a aplicação. Por isso, a qualidade da triangulação é importante. Com isso, nos últimos 40 anos, foram propostos diversos métodos de geração de malhas triangulares. Exemplos de malhas triangulares não conformes, formadas por triângulos retângulos isósceles e escalenos são mostrados nas figuras 1.4 e 1.5, respectivamente. Em particular, em uma triangulação *restrita*, determinadas arestas devem constar na triangulação, em que a fronteira do domínio é um exemplo.

Podem ser utilizados diferentes domínios na geração de uma triangulação. Os domínios para malhas triangulares podem ser subdivididos em quatro tipos de



**Figura 1.4:** Malha triangular não conforme, formada por 6668 triângulos retângulos isósceles gerada com refinamento adaptativo [69].



**Figura 1.5:** Malha triangular não conforme, formada por 1697 triângulos escalenos gerada com refinamento adaptativo [67].

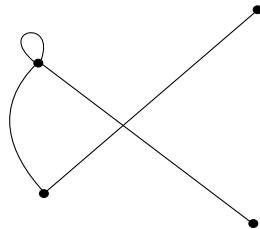
domínio de entrada [16]. Os quatro tipos de entrada e as definições dessas entradas são os seguintes [16].

- Polígono simples: o domínio é uma região poligonal fechada do plano e sua fronteira é formada por arestas simples. A triangulação deve utilizar as arestas da fronteira como arestas da triangulação. No entanto, arestas da fronteira podem ser subdivididas e podem formar várias arestas colineares na triangulação. Em mais detalhes, triangulação de um polígono  $p$  é a decomposição de  $p$  em triângulos por diagonais (que não se interceptam) entre vértices de  $p$  e estão no interior de  $p$ .
- Regiões não contínuas: essas regiões diferem do polígono apresentado anteriormente; pois, na fronteira podem haver regiões disjuntas.
- Conjunto de pontos: os vértices da triangulação são exatamente os pontos de entrada e o domínio da triangulação pode ser considerado como o fecho convexo dos pontos, definido a seguir.

**Definição 1.10.** (*fecho convexo*) O fecho convexo de um conjunto  $\mathfrak{A} = \{a_1, \dots, a_n\}$  em um espaço vetorial é o conjunto de todas as combinações convexas de elementos de  $\mathfrak{A}$ , isto é, é o conjunto de todas as somas  $\alpha_1 a_1 + \dots + \alpha_n a_n$ , com  $\alpha_i \geq 0$  e  $\sum_{i=1}^n \alpha_i = 1$ , para  $n$  arbitrário.

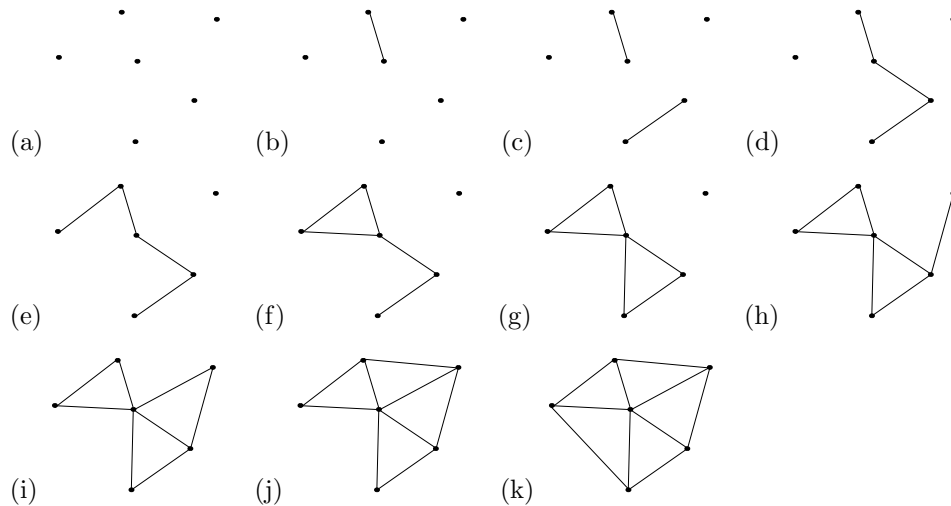
Em termos simples, o fecho convexo de um conjunto de  $n$  pontos é o menor polígono convexo que contém esses pontos. Veja a figura 1.3(c) para exemplificação de fecho convexo de um conjunto de pontos.

- *Planar straight line graph* (PSLG, ou grafo planar com arestas não curvas): a entrada é um conjunto de vértices e segmentos de reta no plano que se intersectam somente nos vértices. Os segmentos devem estar presentes na triangulação final e devem ser utilizados como arestas na triangulação, apesar de que, dependendo da aplicação, pode ser permitido que as arestas do PSLG sejam subdivididas e formem várias arestas colineares na triangulação. Já foram mostrados exemplos de PSLG, nas figuras 1.1 a 1.5, com exceção da figura 1.3(a). Mostra-se um exemplo que não é um PSLG na figura 1.6, ao considerar que os pontos estão destacados.



**Figura 1.6:** Um exemplo que não é um PSLG, pois há um laço, ocorre cruzamento entre arestas e há uma aresta curva.

Dois exemplos de triangulações são os seguintes. Uma triangulação gulosa é, como o nome indica, uma triangulação que utiliza a abordagem gulosa para a geração da malha. Ao se utilizar a abordagem gulosa no projeto de um algoritmo, em termos simples, encontra-se, entre todas as alternativas em um determinado passo do algoritmo, a alternativa que melhor atenda às condições naquele passo. Não se pode garantir que sempre será encontrada a solução ótima do problema quando se utiliza uma abordagem gulosa; mas, geralmente, projeta-se um algoritmo eficiente para o problema [63]. Com essa abordagem, diversas variações de algoritmos podem ser projetados. Uma delas é inserir arestas entre vértices mais próximos até que o resultado final seja uma triangulação. Na figura 1.7, pode ser observado um exemplo de geração da malha, utilizando-se a triangulação gulosa: arestas são inseridas entre vértices mais próximos até se obter uma triangulação. Um algoritmo guloso alternativo poderia ser projetado de forma a inserir uma primeira aresta e, em seguida, arestas seriam inseridas a partir de vértices com arestas incidentes, até se obter a triangulação final.



**Figura 1.7:** Exemplo do processo de geração de malha utilizando a triangulação gulosa com seis vértices. Note que, a cada passo, ocorre a inserção de uma aresta válida.

A triangulação MinMax foi proposta por Babuska e Aziz [5]. É necessário que os ângulos dos triângulos que compõem a triangulação não sejam próximos a  $180^\circ$  para a geração dessa triangulação.

Em termos simples, com a *técnica avanço de fronteira (advancing front method)*, a geração da malha irregular é realizada de forma progressiva, um polítopo por vez, tipicamente, a partir da construção de polítopos na fronteira, em direção às regiões internas do domínio. A região onde há novos polítopos e região ainda sem polítopos é chamada de *fronteira*. Em geral, nesses métodos, polítopos de ótima qualidade são os primeiros a serem gerados, isto é, na fronteira do domínio, e os piores polítopos são os últimos a serem gerados, tipicamente, nas regiões mais internas do domínio.

## 1.7 Notas bibliográficas

Desde a publicação de Thompson, Warsi e Mastin [142], uma quantidade razoável de livros inteiramente dedicados à geração de malhas tem sido publicada. Um livro

sobre geração de malhas fortemente recomendado é *Handbook of Grid Generation*, editado por Thompson, Soni e Weatherhill [143]: são 37 capítulos e contribuições de 30 outros pesquisadores, além de textos dos editores. Além dessa obra, outras fontes excelentes sobre geração de malhas, em diversos aspectos importantes e não abrangidos neste presente texto, são os livros *Mesh Generation, application to Finite Elements* de Frey e George [56] e *Delaunay Mesh Generation* de Cheng, Dey e Shewchuk [32]. Em particular, três técnicas importantes no contexto de geração de malhas triangulares não foram abordadas neste texto. Um exemplo de excelente texto sobre *triangulação de polígonos* é o capítulo 3 do livro *Computational Geometry*, de Berg e colaboradores [15]. Outra técnica importante e não abordada o suficiente neste texto é avanço de fronteira. Exemplos em que essa técnica é descrita em detalhes é no capítulo 6 do livro de Frey e George [56] e no capítulo 17 (e em várias outras partes) do livro de Thompson, Soni e Weatherhill [143]. Ainda, outra técnica importante e não abordada neste texto é varredura e o livro *Computational Geometry*, de Berg e colaboradores [15] é um ótimo texto sobre o assunto.

As técnicas de extrusão e *paving* também não são abordadas neste texto por serem mais utilizadas com malhas quadrilaterais. Ótimos livros que abordam extrusão e *paving* são *Mesh Generation, application to Finite Elements*, de Frey e George [56, p. 279], e *Handbook of Grid Generation*, editado por Thompson, Soni e Weatherhill [143, P-3.2.3], respectivamente. Finalmente, definições básicas de conceitos matemáticos são descritas em diversos livros, e o livro *Dictionary of applied math for engineers and scientists*, editado por Previato [118], é um bom exemplo.

## 1.8 Exercícios

1. Explique o que são:
  - (a) bola fechada:
  - (b) bola aberta:
  - (c) domínio fechado:
  - (d) domínio aberto:
  - (e) fronteira de um domínio:
2. Explique os seguintes termos.
  - (a) Grafo:
  - (b) Vértices:
  - (c) Aresta:
  - (d) Adjacência:
  - (e) Incidência:
  - (f) Grau de um vértice:
  - (g) Valência de um vértice:
  - (h) Malha:
  - (i) Polígono:
  - (j) Politopo convexo:
  - (k) Poliedro:

- (l) Politopo:
  - (m) *Simplex*:
3. Descreva as características dos seguintes tipos de triângulos.
    - (a) Retângulos:
    - (b) Isósceles:
    - (c) Equiláteros:
    - (d) Escalenos:
  4. Explique o que é combinação convexa.
  5. Explique o que é fecho convexo de um conjunto de pontos.
  6. Explique o que é homeomorfismo.
  7. O fecho convexo de  $n$  pontos no espaço euclidiano  $d$ -dimensional é homeomórfico a uma bola no espaço euclidiano  $d$ -dimensional ou no espaço euclidiano  $(d - 1)$ -dimensional? Explique.
  8. Explique o que é um grafo planar.
  9. Explique o que é um PSLG.
  10. Explique a(s) diferença(s) entre malha e grafo.
  11. Explique as diferenças entre os seguintes tipos de malhas.
    - (a) Malhas uniformes e não uniformes.
    - (b) Malhas regulares e irregulares.
    - (c) Malhas isotrópicas e anisotrópicas.
  12. A utilização de uma malha irregular é sempre melhor que a utilização de uma malha regular? Por quê?
  13. Explique o que são malhas triangulares.
  14. Cite três tipos diferentes de domínio para a geração de uma triangulação.



## Capítulo 2

# Técnicas para adaptação de malhas

### 2.1 Introdução

Com a adaptatividade de uma malha computacional, alterações na malha são realizadas à medida que a solução do modelo matemático é calculada. A utilização de técnicas de adaptação é uma forma eficaz de se garantir uma solução numérica com boa precisão em regiões críticas do domínio a um custo computacional aceitável. É importante que, com os critérios de adaptação, sejam resolvidas tanto as características descontínuas da solução do modelo matemático, isto é, ondas de choque, contato etc., quanto as características suaves. Adaptações dinâmicas, juntamente com o tratamento de configurações reais em três dimensões por meio de estruturas de malhas compostas, são assuntos bastante investigados em geração de malhas computacionais.

Em formas típicas de adaptatividade, a malha é localmente refinada pela inserção seletiva de vértices e/ou é movida para concentrar vértices em uma região selecionada, com o objetivo de ser apresentada uma aproximação numérica adequada à solução do modelo matemático. Essas adaptações podem reduzir oscilações de resoluções inadequadas em gradientes grandes, permitindo representações nítidas e adequadas de fronteiras de choques e de camadas limites, como exemplos.

Procedimentos computacionais permitem o cálculo de uma aproximação inicial para a solução de um determinado problema. Essa aproximação, geralmente, é melhorada com a adaptação da malha. Por exemplo, métodos podem ser utilizados para prever as características desejadas da malha, ou seja, o tamanho e a forma dos polítopos, de forma que uma nova malha adaptada possa ser gerada. O objetivo final do processo de adaptação é definir as características da malha ideal para ser utilizada pelo método numérico. Como exemplo, isso pode ser definido como uma malha em que o número de graus de liberdade necessário para se atingir um nível específico de precisão é mínimo. Alternativamente, pode ser interpretado como a malha em que um determinado número de vértices é distribuído de tal forma que a maior precisão possível é alcançada [143, subseção 35.4.1].

Em um processo adaptativo, a decisão fundamental consiste em determinar quando e onde adaptar a malha. Para isso, é necessário um estimador de erro que guiará o processo. É o valor do estimador que permite decidir quando se torna necessário redefinir a malha para se continuar a simulação, quer por imposição de

um erro máximo tolerável, quer pela própria malha não ser mais aceitável por conter politopos demasiadamente distorcidos [92, p. 67]. A avaliação da qualidade em malhas triangulares é um tópico relevante e isso é tratado no capítulo 3.

Critérios de adaptação baseiam-se na avaliação do erro na aproximação da solução das equações que regem ou são construídos para detectar as características do modelo matemático em análise. Esses estimadores são intimamente ligados às equações a serem resolvidas [143, seção P-3.3]. O erro calculado, estimado a partir da aproximação corrente, é analisado para que se alcance uma distribuição espacial ideal dos vértices da malha. Estimadores de erros sofisticados podem ser utilizados para algumas classes de problemas. Os estimadores de erros podem ser aplicados para fornecer uma solução numérica que pode ter um pré-limite sobre os *erros de discretização*, que é a diferença entre a aproximação computada e a solução da equação diferencial parcial [32, p. 5]. Cao, Huang e Russell [29] apresentam vários exemplos de indicadores de erro.

A malha corrente é, então, modificada com o objetivo de atender, tão próximo quanto possível, a uma distribuição ótima. A malha resultante é utilizada para produzir uma aproximação nova, em que o erro da aproximação deve ser diminuído e o procedimento pode ser repetido várias vezes até que o usuário esteja satisfeito com a qualidade da solução calculada [143, subseção 35.4.1]. Esse esquema é mais empregado em malhas irregulares, em que, como exemplo, com a inserção de vértices e as conexões subsequentes, as características de formato dos politopos da malha não são alterados.

Com o princípio de equidistribuição [37], busca-se rearranjar os vértices de uma malha de forma que uma determinada medida seja distribuída equitativamente ao longo de cada subintervalo da malha. Essa medida pode ser, por exemplo, uma medida de erro da discretização que será comparada com a medida de um politopo desejável, hipoteticamente ótimo. A diferença entre cada politopo da malha e o politopo desejável será, aproximadamente, a mesma para todos os politopos existentes. Algumas restrições topológicas, como cantos não convexos em problemas multidimensionais, podem impedir um movimento de vértices ideal. Por isso, não se pode garantir que a equidistribuição seja satisfeita para todos os politopos da malha [105]. Thompson, Soni e Weatherhill [143, seção P-3.3] explicam que, uma vez que um critério de adaptação é estabelecido, o princípio de equidistribuição pode ser atingido por meio de uma variedade de técnicas, como a inserção de vértices, simplificação, movimento de vértices, *remeshing* ou combinações dessas técnicas.

Há três estratégias básicas que podem ser empregadas em malhas adaptáveis dinamicamente, acopladas com as equações diferenciais parciais dos problemas físicos: refinamento e simplificação locais de vértices, abordados na seção 2.2; movimento ou redistribuição de um número fixo de vértices, descrito na seção 2.3; aumento local na ordem do método, comentado na seção 2.4. Claramente, combinações dessas abordagens são possíveis e combinações entre refinamento e movimento de vértices são abordadas também na seção 2.4. Finalmente, a troca e o colapso de arestas são abordadas na seção 2.5.

## 2.2 Refinamento e simplificação locais

No *refinamento adaptativo de malhas* no contexto de volumes finitos, ou *refinamento  $h$*  no contexto de elementos finitos, vértices são inseridos localmente na estrutura de vértices nas regiões do domínio onde é necessária uma aproximação melhor à solução do modelo matemático, como em regiões de erro ou gradiente da solução

relativamente grande. A inserção local de vértices pode ser realizada em qualquer tipo de malha. O objetivo principal da inserção de novos vértices na malha, com o refinamento adaptativo, é obter uma qualidade maior da solução numérica ao custo do aumento do custo computacional nos cálculos. Em geral, se uniforme, a malha se torna não uniforme com a inserção de vértices. Pontos também podem ser removidos e, com isso, obtém-se uma *simplificação* da malha. Transformações topológicas são operações que alteram a conectividade da malha ao remover elementos (vértices, arestas etc.) da malha ou alterá-los de forma que fiquem com uma configuração diferente da original. Os estudos sobre refinamento adaptativo de malhas começaram na década de 1970 e ainda é um tópico ativo de pesquisa. Publicações recentes no assunto incluem Nie, Li e Wang [97], Goffin et al. [62], Baker et al. [6] e Gonzaga de Oliveira, Kischinhevsky e Tavares [69].

O refinamento em malhas irregulares é mais fácil de ser implementado do que em malhas regulares. Isso porque a inserção de vértices em uma malha irregular envolve uma reconexão local dos politopos e a malha resultante tem o mesmo esquema da malha inicial. Dessa forma, o método resolutor pode ser utilizado na malha refinada como foi utilizado na malha inicial.

Em uma malha regular, em geral, a inserção de vértices poderá interferir na regularidade no esquema matricial que representa a malha. Além disso, podem ocorrer vértices não conformes. Portanto, refinar malhas regulares poderá exigir uma modificação na base da estrutura de dados. Também, o surgimento de vértices não conformes poderá exigir uma propagação do refinamento e, conseqüentemente, um aumento exagerado no número de vértices na malha que, por sua vez, pode acarretar em um custo computacional muito alto da resolução. Claramente, o refinamento de vértices em malhas regulares não é um processo tão natural como o processo aplicado em malhas irregulares e, portanto, não é tão amplamente empregado. Trabalhos foram desenvolvidos para implementar refinamento em malhas regulares e os resultados mostram que os benefícios obtidos podem não compensar o esforço adicional das modificações na estrutura de dados e no método resolutor. Veja detalhes sobre esse assunto na subseção P-3.3.1 de Thompson, Soni e Weatherhill [143]. Pode-se utilizar, inicialmente, uma malha regular e uniforme e que se torna não uniforme com adaptações sucessivas.

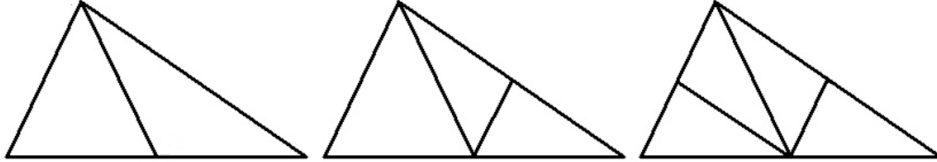
Uma vantagem da utilização do refinamento adaptativo por inserção de vértices é que a estrutura original de vértices fixos é preservada. Na abordagem por inserção de vértices em regiões de erro ou gradiente grande, não há, evidentemente, esgotamento dos vértices em outras regiões da malha e, portanto, nenhum aumento formal de erro ocorre. O erro global tende a diminuir já que o erro é reduzido localmente na região em que houve o refinamento adaptativo. Claramente, o refinamento com êxito aumenta a resolução de uma análise.

As desvantagens da utilização do refinamento adaptativo por inserção de vértices são que o tempo de processamento e de armazenamento aumentam com o refinamento. Ainda, a estrutura de dados e a codificação podem ser complicadas. Também, o refinamento adaptativo por inserção de vértices fornece um retorno decrescente em refinamentos sucessivos.

Definido o politopo a ser refinado, deve-se, ainda, definir como o politopo será refinado. Em particular, há diversas formas de se definir como subdividir *triângulos* e algumas são descritas a seguir.

- Uma das primeiras propostas para se subdividir triângulos foi a de traçar a mediana do triângulo, a partir do ponto médio da maior aresta, ou bisseção do triângulo. A *mediana* de um triângulo é o segmento de reta que conecta

um vértice do triângulo ao ponto médio da aresta oposta a esse vértice. Em qualquer triângulo, uma mediana divide o triângulo em duas regiões de áreas iguais. Em particular no triângulo retângulo, a mediana que parte do vértice no ângulo reto divide a hipotenusa em dois segmentos do mesmo tamanho da mediana. Essa subdivisão de triângulo, exemplificada na imagem à esquerda da figura 2.1, é denotada como subdivisão 2T-LE (*2-triangle longest-edge partition*).

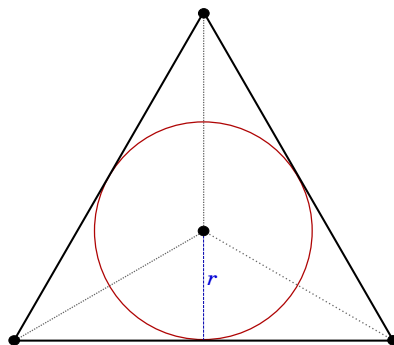


**Figura 2.1:** Formas possíveis de um triângulo ser subdividido: à esquerda, a subdivisão 2T-LE (*2-triangle longest-edge partition*); ao meio, a subdivisão 3T-LE (*3-triangle longest-edge partition*); à direita, a subdivisão 4T-LE (*4-triangle longest-edge partition*), que é a subdivisão por meio de particionar o triângulo pela mediana e, em seguida, subdividir os dois triângulos resultantes também por suas medianas [122].

Para se manter a conformidade da malha com a subdivisão 2T-LE, também deve-se subdividir o triângulo adjacente, isto é o triângulo que compartilha a maior aresta com o triângulo refinado, a menos que a aresta esteja na fronteira do domínio. Isso significa que o refinamento deve ser propagado até que uma aresta dividida seja a maior aresta também do triângulo adjacente ou uma aresta da fronteira seja dividida.

Essa subdivisão do triângulo faz com que a qualidade dos novos triângulos não seja deteriorada em relação ao triângulo original. Rosenberg and Stenger [123] provaram que os ângulos interiores não se tornam zero com esse esquema de refinamento. Stynes [135] demonstrou que os formatos dos triângulos produzidos por sucessivas subdivisões pelo esquema 2T-LE tendem a ser próximos de um triângulo equilátero. Ainda, Stynes [135] provou que os triângulos resultantes satisfazem a uma propriedade de regularidade no formato dos triângulos. Essas propriedades são compartilhadas por todos os esquemas baseados em bisseção, como os esquemas 3T-LE e 4T-LE, exemplificados nas imagens ao meio e à direita na figura 2.1, criando-se três e quatro triângulos, respectivamente [64]. Velho and Zorin [148] apresentaram uma subdivisão 4-8, que também utiliza a bisseção.

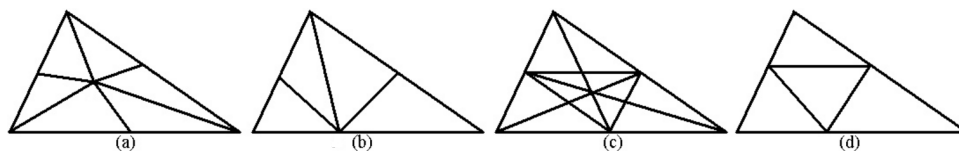
- Uma forma simples de se particionar um triângulo é inserir o ponto na coordenada do centro da circunferência inscrita do triângulo e os três vértices do triângulo são ligados a esse ponto, como exemplificado na figura 2.2. Com isso, tem-se a vantagem de que sempre o ponto a ser inserido pertence ao triângulo. A desvantagem dessa forma de se particionar um triângulo é que os três novos triângulos podem ser de má qualidade.
- O circuncentro do triângulo também pode ser inserido. Circuncentro é o centro do circuncírculo do triângulo, que é o único círculo que passa pelos três vértices do triângulo. Em termos simples, o circuncentro é a intersecção das mediatrizes das arestas de um triângulo. O circuncentro é equidistante aos três vértices do triângulo. Portanto, essa distância é o raio, ou circunraio, de



**Figura 2.2:** Inserção do centro da circunferência inscrita do triângulo. Esse ponto é ligado aos três vértices do triângulo, formando três novos triângulos, que substituem o triângulo original.

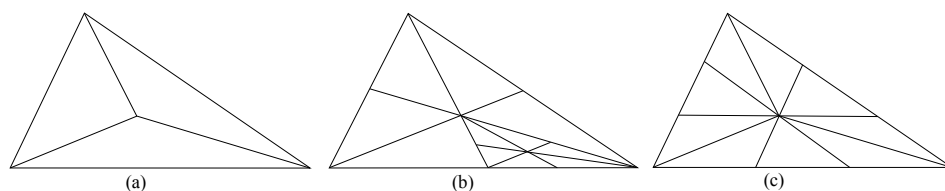
uma circunferência circunscrita ao triângulo, que tangencia seus vértices. Também em termos simples, a mediatriz de uma aresta é a reta perpendicular a essa aresta que passa exatamente no ponto médio da aresta. Claramente, a mediatriz de uma aresta de um triângulo é a reta que passa pelo ponto médio de uma aresta do triângulo e também pelo seu circuncentro. A desvantagem de se inserir o circuncentro é que o novo ponto pode ficar fora do triângulo e é necessário procurar em qual triângulo está o novo ponto. Mesmo se o circuncentro for externo ao triângulo, a solução numérica no triângulo original será, eventualmente, melhorada.

- Outra forma de se particionar um triângulo é calcular os raios das circunferências inscrita (inraio)  $r$  e circunscrita (circunraio)  $R$  e as coordenadas dos respectivos centros  $x_i$  e  $x_c$ . As coordenadas do novo ponto são  $x_n = x_c + (1 - \rho)x_i$ , em que  $\rho = \frac{2r}{R}$ . Se o triângulo for equilátero, os centros das circunferências inscrita e circunscrita coincidem, tem-se  $\rho = 1$  e o circuncentro é inserido. A desvantagem dessa abordagem é que o ponto também pode estar fora do triângulo. O novo ponto pode estar próximo de uma aresta ou de outro ponto e isso causará polítopos de má qualidade. Pode-se decidir ignorá-lo e prosseguir para o polítopo seguinte a ser refinado.
- Powell e Sabin [117] mostraram várias formas de se particionar triângulos. São mostradas algumas formas possíveis de se subdividir um triângulo na figura 2.3. Na subdivisão mostrada na figura 2.3(a), pode-se inserir qualquer ponto interior ao triângulo, criando-se seis novos triângulos. Na subdivisão mostrada na figura 2.3(b), que é uma generalização da subdivisão 4T-LE, pode-se inserir qualquer ponto na maior aresta do triângulo, criando-se quatro novos triângulos que substituem o triângulo original. Na subdivisão mostrada na figura 2.3(c), os pontos médios das arestas são ligados aos pares e as medianas também são traçadas. Essa subdivisão cria 12 novos triângulos. Na subdivisão mostrada na figura 2.3(d), quaisquer pontos em cada aresta são inseridos e ligados aos pares. Essa subdivisão cria quatro novos triângulos que substituem o triângulo original.
- Outra subdivisão possível de triângulo é traçar segmentos de reta dos vértices do triângulo até seu baricentro, como mostrado na figura 2.4(a). Entretanto,



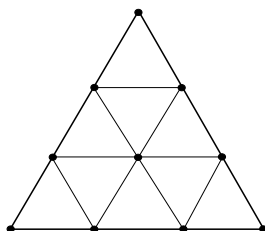
**Figura 2.3:** Subdivisões de Powell-Sabin [117].

essa subdivisão pode piorar a qualidade dos triângulos resultantes em relação ao triângulo original. O mesmo pode ocorrer com a subdivisão que cria seis novos triângulos (figura 2.3(a)), como exemplificado na figura 2.4(b). De forma similar, os triângulos resultantes podem ser de má qualidade com a subdivisão que gera nove novos triângulos, como mostrado na figura 2.4(c). Essa subdivisão divide cada aresta do triângulo em três partes de mesmo tamanho e liga-os ao baricentro do triângulo, além de ligar o baricentro aos vértices do triângulo. Veja o artigo de Plaza e Rivara [115], para detalhes sobre essas subdivisões de triângulos.



**Figura 2.4:** Subdivisões ternárias de triângulos: (a) subdivisão ao inserir o baricentro e ligá-lo aos vértices do triângulo, criando-se três novos triângulos; (b) subdivisão ao inserir o baricentro do triângulo e ligá-lo aos vértices do triângulo e também aos pontos médios das arestas, criando-se seis novos triângulos; (c) subdivisão das arestas em três partes iguais e também inserção do baricentro do triângulo, ligá-lo aos seis pontos criados nas arestas e também aos vértices do triângulo, criando-se nove novos triângulos.

- Ainda, outra subdivisão de triângulo é dividir cada uma das três arestas em três partes iguais e ligar esses seis pontos ao baricentro do triângulo e, também, traçar três novas arestas que ligam esses seis pontos aos pares em arestas adjacentes, formando-se nove novos triângulos, como mostrado na figura 2.5 [56, p. 564].



**Figura 2.5:** Cada uma das três arestas do triângulo é dividida em três partes iguais e os seis pontos são conectados ao baricentro do triângulo e, também, três novas arestas são inseridas ao ligar os seis pontos aos pares entre arestas adjacentes, formando-se nove novos triângulos [56, p. 564].

Em termos simples, o *baricentro* é o centro de massa de um polígono. Considere a definição de *baricentro* de um *simplex* a seguir.

**Definição 2.1.** (*baricentro de um simplex*) O *baricentro* (ou *centro de massa*) de um *simplex* composto pelos pontos  $a_1, \dots, a_n$  é  $\frac{a_1 + \dots + a_n}{n}$ .

## 2.3 Movimento de um número fixo de pontos

No movimento de malhas no contexto de volumes finitos, ou refinamento  $r$  no contexto de elementos finitos, pontos são movimentados (ou redistribuídos) de forma que se concentrem nas regiões de erro ou gradiente relativamente grande da solução. Isso significa que malhas móveis são adaptadas continuamente e vértices são movidos na direção de regiões em que há maior variação na solução do modelo matemático. Depois de se assegurar que há pontos suficientes na malha, o movimento de pontos pode fornecer o mecanismo necessário para se alcançar uma resolução alta com um custo computacional baixo. A seguir, nas subseções 2.3.1 e 2.3.2, são abordadas características básicas de movimento de malhas e a suavização laplaciana, respectivamente.

### 2.3.1 Características básicas de movimento de malhas

Nessa abordagem, os pontos são movidos de regiões com erro ou gradiente na solução relativamente pequeno para as regiões de erro ou gradiente grande. A redistribuição dos pontos deve ser realizada de forma a não diminuir seriamente o número de pontos em outras regiões de possíveis gradientes também significativos. Isso significa que o aumento do espaçamento que ocorre em alguma região não deve piorar a solução numérica nas outras regiões do domínio.

Com o movimento dos vértices da malha, a aproximação global não é melhorada no sentido matemático formal [143, subseção 1.3.6.1]. A redistribuição adaptativa dos pontos tem as suas raízes no princípio da equidistribuição de erros [143, subseção 1.3.6.4].

Essa abordagem de adaptatividade da malha pode ser, em geral, utilizada para problemas transientes por causa da mobilidade da malha, que facilita lidar com integradores de tempo. Entretanto, sua limitação está na dificuldade em definir um intervalo de tempo adequado, uma vez que os pontos variam de posição ao longo do tempo, podendo ocorrer o entrelaçamento de arestas. Ainda, a aplicabilidade do movimento de malhas é limitada devido ao número fixo de graus de liberdade e a uma conectividade fixa dos polítopos da malha. Com isso, o movimento de malhas é, tipicamente, utilizado para acelerar o processo computacional em vez de ser utilizado para se alcançar uma precisão prescrita [105].

Como afirmam Huang e Russell [78], os métodos que utilizam malhas móveis, ou simplesmente, métodos de malhas móveis, ainda estão em uma fase relativamente inicial de desenvolvimento. Muitos deles estão em estágio experimental e, quase todos, requerem uma justificativa matemática adicional. Como também explicam esses autores, uma análise rigorosa dos métodos de malhas móveis, para resolver equações diferenciais parciais dependentes do tempo, só foi realizada para alguns modelos simples de problemas e também afirmam que muitas formas de se melhorar sua eficiência e robustez serão, sem dúvida, desenvolvidos. Como, por exemplo, ainda são necessários mais estudos numéricos sistemáticos de como se reduzir os custos na resolução de todo um sistema de malhas e equações diferenciais parciais, bem como estudos em como se equilibrar a adaptação espacial e temporal de uma malha.

Explicam, também, que um fator importante nos métodos de malhas móveis está na escolha adequada de uma função de densidade da malha. Essa função controla a concentração de pontos da malha por meio do princípio de equidistribuição e, tipicamente, avalia a dificuldade na aproximação numérica espacial do problema sendo resolvido. De acordo com Huang e Russell [78], a seleção da função de densidade da malha pode ser baseada na estimativa de erro de interpolação, na invariância de escala ou em uma estimativa de erro *a posteriori*, com o limite ótimo para o erro de interpolação ou o erro da solução, também obtido pela malha equidistribuída correspondente [105].

A abordagem por redistribuição tem as vantagens de não aumentar o tempo de processamento e de armazenamento durante as iterações da resolução. As desvantagens do movimento de vértices da malha são o possível esgotamento da possibilidade de se movimentar vértices em certas regiões e a possível piora na qualidade dos polítopos. Hasssan e Probert [143, subseção 35.4.5] explicam que, em alguns casos, a melhoria obtida pelo movimento de pontos pode ser mínima. Isso pode ocorrer porque o algoritmo pode não permitir a inserção de novos pontos e, então, a qualidade da aproximação final é dependente da topologia da malha inicial. Por outro lado, uma solução numérica pode combinar o movimento de pontos, o refinamento adaptativo por inserção de vértices e a simplificação. Esses procedimentos devem ser implementados para que ocorram de forma dinâmica, ou seja, aplicados em intervalos regulares na simulação. Essa abordagem também oferece a possibilidade de se utilizar o movimento de pontos e refinamento para, de forma independente, capturarem as características em análise.

### 2.3.2 Suavização laplaciana

Uma forma de melhoria da qualidade de malhas é por *suavização*. *Suavização* é a tarefa de mover vértices para melhorar a qualidade dos polítopos incidentes aos vértices. A suavização não altera a conectividade da malha, isto é, a topologia da malha.

Uma técnica para movimentos de vértices que é eficaz e aplicável a todos os tipos de malhas baseia-se na formulação laplaciana ponderada. Essa técnica é chamada de *suavização laplaciana* [80] no contexto de melhoria de qualidade de malhas e é utilizada para reduzir a distorção de polítopos ao ajustar a posição dos vértices. A suavização laplaciana é chamada dessa forma por causa da sua interpretação como um operador diferencial laplaciano finito [32, p. 28]. Essa técnica foi uma das primeiras formas de se melhorar a qualidade de malhas e é, possivelmente, a técnica mais simples e conhecida para suavização de malhas.

A forma básica de suavização laplaciana é mover um vértice para o baricentro do polígono formado pelos seus vértices incidentes. Isso é feito para todos os pontos internos da malha. A suavização laplaciana opera heurísticamente e não garante a melhoria da qualidade dos polítopos. Essa técnica pode ser aplicada após outra forma de adaptação da malha ter sido realizada, como a inserção de novos vértices ou o movimento de vértices da malha.

Uma forma típica para a suavização laplaciana no plano euclidiano, aplicada no ponto  $p$  é  $p' = p + \beta \frac{\sum_{i=1}^n \varphi(p, p_i)(p_i - p)}{\sum_{i=1}^n \varphi(p, p_i)}$ , em que a posição corrente do ponto é  $p = (x, y)$ ,

$p'$  é a nova posição do ponto  $p$  após a suavização,  $p_i$  é um ponto adjacente ao ponto  $p$  e  $\beta$  é um parâmetro de controle de intensidade de movimento, definido no intervalo  $0 < \beta \leq 1$ . A intensidade do movimento pode ser estabelecida de modo que, quanto



mais  $\beta$  é próximo a zero, menos movimento dos vértices da malha ocorre e isso poderá causar lentidão na definição da melhor malha para o problema, pois vários passos podem ser necessários até que se tenha a malha desejada. Com  $\beta = 1$ , há um movimento total dos vértices dado pela fórmula e isso poderá causar entrelaçamento de arestas. Os somatórios são considerados sobre todas as  $n$  arestas  $\overline{pp_i}$ . A função adaptativa de peso  $\mathbf{p}(p, p_i)$  entre os pontos  $p$  e  $p_i$  pode ser considerada como uma medida da atividade e formas possíveis são mostradas a seguir.

- Uma abordagem típica é  $\mathbf{p}(p, p_i) = |\phi(p_i) - \phi(p)|$ , em que  $\phi(p)$  e  $\phi(p_i)$  são valores da solução corrente nos vértices  $p$  e  $p_i$ , respectivamente. Nesse caso, as intensidades dos movimentos são consideradas localmente e regiões com variações distintas podem ter movimentos de vértices com intensidades similares.
- Uma abordagem é mostrada por Thompson e Weatherhill [143, subseção 1.4.5.2], em que  $\mathbf{p}(p, p_i) = \chi \left| \frac{\phi(p_i) - \phi(p)}{\phi(p_i) + \phi(p)} \right|$ , para  $0 < \chi \leq 1$ . Nessa abordagem, é necessário estabelecer também o parâmetro  $\chi$ .
- Com o intuito de evitar possíveis distorções de sucessivas iterações, Taubin [137, 138] propôs utilizar  $\mathbf{p}(p, p_i) = |\phi(p_i) - \phi(p)|$  e combinar duas suavizações sucessivas:  $p' = p + \lambda \Delta p$  e  $p'' = p' - \mu \Delta p'$ , em que  $\Delta p = \frac{\sum_{i=1}^n \mathbf{p}(p, p_i)(p_i - p)}{\sum_{i=1}^n \mathbf{p}(p, p_i)}$  e os parâmetros quantificadores do movimento satisfazem  $0 < \lambda < \mu$ . Esse esquema é chamado de suavização  $\lambda/\mu$ . Taubin, Shang e Golub [139] analisaram as propriedades desse esquema e mostraram como minimizar seu tempo de execução. Kobbelt e colaboradores [81] propuseram utilizar a função  $\lambda/\mu$  de Taubin [137, 138] com  $\lambda = \mu = 1$ , intitulando-a como suavização bi-laplaciana.
- Uma variação simplificada é  $p' = p + \frac{\beta}{(\Delta\phi)_{max}} \sum_{i=1}^n (p_i - p) \cdot |\phi(p_i) - \phi(p)|$ , em que  $(\Delta\phi)_{max}$  é a variação máxima de  $|\phi(p_i) - \phi(p_j)|$  ao considerar todos as arestas da malha. Nessa abordagem, tem-se a vantagem de movimentar os vértices considerando-se uma medida de erro global, com a desvantagem de se ter que encontrar a variação máxima de toda a malha. Gonzaga de Oliveira, Oliveira e Chagas [71] mostraram que essa variação pode ser melhor que as quatro primeiras opções. Isso porque o movimento é proporcional à variação local da região, isto é, regiões com mais variação na solução têm vértices movimentados com mais intensidade e regiões com menos variação na solução têm vértices movimentados com menos intensidade.

A suavização laplaciana, geralmente, funciona bem em triangulações, mas não é confiável para tetraedros, quadriláteros e hexaedros [32, p. 28]. Cheng, Dey e Shewchuk [32, p. 28] também explicam que suavizadores baseados em otimização e mais sofisticados que a suavização laplaciana começaram a ser publicados na década de 1990 [28]. Apesar de ser um procedimento mais lento que a suavização laplaciana, suavizações melhores são obtidas pelo algoritmo de Freitag, Jones e Plassmann [53, 54, 55]. Em vez de movimentar o ponto para o baricentro do polígono formado pelos vértices adjacentes do ponto, o método baseado em otimização desses autores computa, para cada ponto de Steiner, uma nova posição que maximiza o ângulo mínimo em triângulos adjacentes. Os autores utilizaram um método de descida mais

acentuada<sup>1</sup> para resolver o problema da posição ótima do ponto. Em termos simples, métodos de descida mais acentuada (*steepest descent*, ou *ascent* se o problema for de maximização) é uma classe de algoritmos em que a direção tomada (o oposto do gradiente no método original) é descendente, em busca de um mínimo local, a partir de uma posição inicial arbitrária. Em mais detalhes, utiliza-se  $x_{k+1} = x_k + \lambda \mathbf{g}$ , tal como explicado a seguir.

- A função que se quer minimizar ( $\min\{f(x) : x \in \mathbb{R}^n\}$ )  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  está em  $C^1$ .
- O tamanho do passo  $\lambda$  deve ser um valor pequeno e é dependente de  $f(x)$ .
- O vetor  $\mathbf{g}$  indica a direção a ser tomada e é escolhido em direção a um ponto de minimização a partir de uma posição inicial. O vetor  $\mathbf{g}$  é sujeito a  $\min\{\nabla f(x)d : \|d\| = 1\}$ , em que  $\|d\|$  é a norma do vetor  $d$ . Quando a norma euclidiana é utilizada, obtém-se o algoritmo original, que move na direção oposta do gradiente, ou seja,  $d = -\frac{\nabla f(x)}{\|\nabla f(x)\|}$ . Nenhum vetor de direção é solicitado para  $\nabla f(x) = 0$ : o algoritmo para quando alcança um ponto estacionário, ou seja,  $\nabla f(x) = 0$  [118].

Métodos de descida mais acentuada são algoritmos de primeira ordem porque utilizam apenas a derivada primeira da função. Por sua vez, ao utilizar programação linear generalizada, Amenta Bern e Eppstein [2] mostraram resolução em tempo linear para o problema de movimento de vértices em malhas irregulares triangulares, de quadriláteros e de tetraedros para otimizar formatos de elementos adjacentes.

## 2.4 Aumento na ordem dos polinômios das funções de forma

Com o aumento na ordem dos polinômios das funções de forma, o método de solução é alterado para uma aproximação de ordem superior nas regiões com erro ou gradiente relativamente grande na solução. Isso aumenta a precisão global, mas sem alterar a distribuição geométrica. Essa abordagem é utilizada no método dos elementos finitos.

As vantagens dessa abordagem são que a distribuição de pontos e os graus de liberdade não são alterados. Consequentemente, não há piora na qualidade da malha e os custos computacional e de armazenamento da aproximação na nova malha não aumentam em relação aos custos da aproximação da malha corrente. A desvantagem é que a aplicação dessa abordagem pode ser mais complicada do que com as outras abordagens comentadas.

O capítulo 22 do livro de Frey e George [56] é inteiramente dedicado à adaptação de malhas por métodos que alteram o grau na aproximação. Os autores também mostram métodos que unem essa técnica com técnicas de inserção de vértices na malha.

## 2.5 Outros tipos de adaptação de malhas

Existem várias operações que podem ser realizadas para melhorar a qualidade da malha. Por exemplo, a troca de arestas, com enfoque na triangulação de Delaunay,

<sup>1</sup>Muitas traduções para *steepest descent method* são utilizadas e a escolhida aqui é *método de descida mais acentuada*.

é abordada na subseção 4.2.2, na página 43.

Outra técnica para melhorar a qualidade da malha é o colapso de uma aresta: arestas de polítopos de má qualidade podem ser excluídas. Se há polítopos de má qualidade, por exemplo, contendo ângulo menor que um limite estipulado pelo usuário, é realizada uma tentativa de remover esses polítopos. Uma forma de se conseguir isso é por meio do colapso de um dos lados do polítopo de má qualidade, de forma que seus pontos se tornem coincidentes. Ao examinar um polítopo, a decisão de qual lado deve entrar em colapso é realizada ao se considerar cada lado do polítopo e examinar os polítopos adjacentes que existiriam se uma aresta em particular fosse removida. A configuração escolhida pode ser a que tiver o maior ângulo mínimo [143, subseção 35.3.1.3]. Note que a troca e o colapso de arestas não aumentam o número de vértices, não movem os vértices, e não aumentam a ordem dos polinômios.

## 2.6 Notas bibliográficas

Uma fonte excelente, em diversos aspectos importantes sobre técnicas de adaptação de malhas, não abrangidos neste texto, é o livro *Handbook of Grid Generation*, editado por Thompson, Soni e Weatherhill [143]. Outras fontes ótimas sobre o assunto são os livros *Mesh Generation, application to Finite Elements* de Frey e George [56] e *Delaunay Mesh Generation* de Cheng, Dey e Shewchuk [32].

## 2.7 Exercícios

1. Explique os seguintes conceitos geométricos ou operações.
  - (a) Circuncírculo:
  - (b) Circuncentro:
  - (c) Circunraio:
  - (d) Baricentro de um triângulo:
  - (e) Bisseção de um triângulo:
  - (f) Mediana de um triângulo:
  - (g) Mediatriz da aresta de um triângulo:
2. Explique o que são erros de discretização.
3. Explique o que é refinamento adaptativo de malhas.
4. Quais as vantagens do movimento de vértices em relação a inserir novos vértices na malha.
5. Explique o que é simplificação local em uma malha computacional.
6. Explique para que é utilizada a suavização laplaciana.



## Capítulo 3

# Métricas de avaliação da qualidade de triângulos

### 3.1 Introdução

Um malha gerada com pontos regularmente espaçados, frequentemente, é satisfatória em geometrias simples. No entanto, politopos de má qualidade podem aparecer em domínios com configurações complexas ou em situações em que há a necessidade de se utilizar politopos que podem ser modificados.

Uma forma de se considerar a qualidade de um politopo é analisar seus ângulos. Por exemplo, politopos podem ser considerados de má qualidade se possuírem respectivos ângulo mínimo ou máximo inferior ou superior a alguma tolerância especificada.

Politopos com ângulos próximos a  $0^\circ$  ou  $\pi$  são considerados de má qualidade. Na aplicação de um método numérico, como o método dos elementos finitos, com malhas com politopos com ângulos próximos a  $0^\circ$ , pode-se obter um sistema de equações lineares mal condicionado [32, p. 5]. Sistemas de equações lineares com matrizes com espectro de autovalores ruins afetam os resolutores desses sistemas: métodos iterativos podem demorar a convergir e erros de arredondamento podem ser introduzidos em métodos diretos [32, p. 7].

Em malhas com politopos com ângulos próximos a  $\pi$ , pode-se obter grandes erros de interpolação na aplicação do método numérico. No método dos elementos finitos, esses erros levam a *erros de discretização*, que é, como descrito, a diferença entre a aproximação computada e a solução da equação diferencial parcial [32, p. 5]. Portanto, politopos de má qualidade podem levar a resoluções instáveis e aproximações imprecisas.

Com isso, a avaliação da qualidade de malhas é um requisito importante, tanto na escolha de uma malha de elementos finitos ou de volumes finitos, quanto em malhas que foram submetidas à adaptação. Logo, é importante satisfazer restrições geométricas no formato dos politopos. Como descrito, é geralmente exigido satisfazer uma condição de ângulo mínimo ou máximo em um triangulação. Por outro lado, a restrição de ângulo mínimo pode ser reformulada, como exemplo, de forma que o círculo inscrito do triângulo não deva ser muito pequeno e essa condição pode ser estendida para dimensões maiores [111].

Várias métricas de qualidade de politopos têm sido propostas, baseadas em relações dimensionais de diversas parametrizações geométricas, em que exemplos po-

dem ser encontrados nos artigos de Baker [7], Parthasarathy, Graichen e Hathaway [109] e no livro de Frey e George [56].

Por sua vez, Pébay e Baker [111, 112] apresentaram análises de diversas métricas de qualidade de triângulos. Pébay e Baker [111, 112] também consideraram as métricas de qualidade de triângulos mais comumente utilizadas, forneceram provas de suas propriedades e examinaram seus comportamentos assintóticos. O objetivo dos autores foi provar vários resultados úteis sobre métricas de qualidade de triângulos que podem levar a uma avaliação adequada tanto de triangulações planares quanto de superfícies triangulares.

Este capítulo está organizado da seguinte maneira. Na seção 3.2, são apresentados conceitos básicos sobre triângulos. Em seguida, há uma descrição de métricas de qualidade apresentadas por Pébay e Baker [111, 112], por Frey e George [56, p. 596-598] e por Cheng, Dey e Shewchuk [32, p. 26-27]. Na seção 3.3, são apresentadas métricas de qualidade de triângulos e de tetraedros baseadas na área do polítopo. Na seção 3.4, apresenta-se uma métrica de qualidade por meio dos ângulos extremos (maior e menor) do triângulo, com uma análise sobre o menor e o maior ângulos. Na seção 3.5, é apresentada uma métrica de qualidade de triângulos baseada na razão entre os raios das circunferências circunscrita (circunraio) e inscrita (inraio) do triângulo. Na seção 3.6, são apresentadas métricas de qualidade de triângulos baseadas na razão entre as arestas extremas do triângulo. Na seção 3.7, são apresentadas métricas de qualidade de triângulos baseadas na razão entre o raio do circuncírculo (circunraio) e arestas extremas do triângulo. Na seção 3.8, são apresentadas métricas de qualidade de triângulo baseadas na razão entre as arestas extremas e o raio da circunferência inscrita (inraio) do triângulo. Pode-se considerar que as métricas de avaliação da qualidade de triângulos não abordadas neste texto são menos utilizadas que as demais. Para finalizar este capítulo, na seção 3.9, são apresentadas considerações finais sobre o assunto, com uma tabela com os resumos das propriedades de métricas de qualidade abordadas.

## 3.2 Notações e conceitos básicos

Um polítopo é considerado degenerado se seu volume é zero [111, 112]. Considere um triângulo não degenerado  $t = \triangle abc$ , como mostrado na figura 3.1, com área  $\alpha$  e os comprimentos das arestas são representados por  $l_1 = \|b - c\|$ ,  $l_2 = \|c - a\|$  e  $l_3 = \|a - b\|$ . Os ângulos nos vértices  $a$ ,  $b$  e  $c$  são  $\alpha$ ,  $\beta$  e  $\gamma$ , respectivamente. Os raios da circunferência inscrita (inraio) e do circuncírculo (circunraio) de  $t$  são  $r$  e  $R$ , respectivamente. São utilizadas as seguintes notações como normas padrões:  $|t|_0 = \min(l_1, l_2, l_3)$ ,  $|t|_\infty = \max(l_1, l_2, l_3)$ ,  $\theta_0 = \min(\alpha, \beta, \gamma)$  e  $\theta_\infty = \max(\alpha, \beta, \gamma)$ .

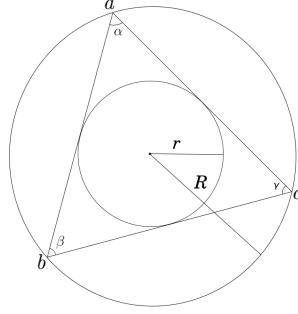
São utilizados resultados de geometria elementar sem provas. Em particular, pela lei dos senos, utiliza-se

$$2R = \frac{l_1 \cdot l_2 \cdot l_3}{2\alpha} = \frac{l_1}{\text{sen } \alpha} = \frac{l_2}{\text{sen } \beta} = \frac{l_3}{\text{sen } \gamma}, \quad (3.2.1)$$

em que

$$\alpha = rs = \sqrt{s(s - l_1)(s - l_2)(s - l_3)}, \quad (3.2.2)$$

com o lado direito dado pela fórmula de Heron. O semiperímetro  $s$  é definido a seguir.



**Figura 3.1:**  $\triangle abc$ : o raio da circunferência inscrita (inraio) é  $r$  e o raio do circuncírculo (circunraio) é  $R$ .

**Definição 3.1** (semiperímetro de triângulo). *O semiperímetro de um triângulo, cujos comprimentos das arestas são  $l_1$ ,  $l_2$  e  $l_3$ , é*

$$s = \frac{l_1 + l_2 + l_3}{2}. \quad (3.2.3)$$

### 3.3 Métricas relacionadas com a área do politopo

A seguir, são citadas algumas métricas para a avaliação da qualidade de triângulos baseadas na área ou volume do triângulo ou tetraedro, respectivamente.

- Uma forma de determinação da qualidade de um triângulo  $t$  foi proposta por Bank e colaboradores [9, 8] ao considerar uma razão entre a área e as arestas do triângulo. Essa métrica é definida como  $\mathbf{b} = \frac{4\sqrt{3}\mathbf{a}}{\mathfrak{s}}$ , em que  $\mathfrak{s} = l_1^2 + l_2^2 + l_3^2$ . Especificamente no plano euclidiano, com  $a = (x_1, y_1)$ ,  $b = (x_2, y_2)$  e  $c = (x_3, y_3)$ , tem-se  $2/\mathbf{a} = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1)$  para vértices orientados em sentido anti-horário; caso estejam orientados em sentido horário, o valor de  $\mathbf{a}$  será negativo. A constante  $4\sqrt{3}$  é utilizada como fator de normalização, de forma que  $0 < \mathbf{b} \leq 1$ . Em geral, as métricas de avaliação da qualidade de triângulos são normalizadas porque muitos usuários preferem que a qualidade de um triângulo equilátero seja 1. Dessa forma, obtém-se  $\mathbf{b} = 1$  para um triângulo equilátero e mais  $\mathbf{b}$  aproxima-se de zero quanto menor é o menor ângulo do triângulo. Há triângulos sem ocorrência de ângulos obtusos para  $\frac{\sqrt{3}}{2} \leq \mathbf{b} \leq 1$ .

Pode-se também utilizar  $\mathbf{f}_1 = \frac{\sqrt{3}\mathfrak{s}}{12\mathbf{a}}$ , em que  $\frac{\sqrt{3}}{12}$  é utilizado como normalização para que seja encontrado o valor 1 para o triângulo equilátero. Similarmente,

tem-se  $\mathbf{f}_2 = \mathbf{n}_2 \frac{(\sqrt{\sum_{i=1}^6 l_i^2})^3}{V_K}$  em três dimensões, em que  $V_K$  e  $l_i$  são o volume e o tamanho da aresta  $i$  do tetraedro  $K$ , respectivamente, e  $\mathbf{n}_2$  é um fator de normalização para que a razão resulte no valor 1 para o tetraedro regular [56, p. 597]. Em particular, um *tetraedro regular* é composto por quatro triângulos equiláteros.

- Outra métrica de qualidade é  $\mathbf{f}_3 = \frac{\sqrt{3}|t|_{\infty}\mathfrak{s}}{6\mathbf{a}}$ , em que  $\frac{\sqrt{3}}{6}$  é utilizado como normalização para que seja encontrado o valor 1 para o triângulo equilátero. Similarmente em três dimensões, tem-se  $\mathbf{f}_4 = \mathbf{n}_4 \frac{\max(l_1, \dots, l_6) \cdot S_K}{V_K}$ , em que  $V_K$  e

$S_K$  são o volume e a soma das áreas das faces do tetraedro  $K$ , respectivamente, e  $n_4$  é um fator de normalização para que a razão resulte no valor 1 para o tetraedro regular [56, p. 596].

- Frey e George [56, p. 598] também citam a métrica  $\frac{V_K^4}{(\sum_{i=1}^4 S_i^2)^3}$ , em que  $S_i$  é a área da face  $i$  do tetraedro  $K$ .
- Frey e George [56, p. 598] também citam a métrica  $\frac{|t|_{med}^3}{V_K}$ , em que  $|t|_{med}$  é uma aresta de tamanho médio.
- Por sua vez, Cheng, Dey e Shewchuk [32, p. 27] citam a métrica  $\frac{V_K}{\left(\sqrt{\frac{1}{6} \sum_{i=1}^6 l_i^2}\right)^3} = \frac{V_K}{\left(\sqrt{\frac{l_1^2 + \dots + l_6^2}{6}}\right)^3}$ , em que o denominador é a raiz do valor quadrático médio (*root mean square*), ou *valor eficaz*, dos comprimentos das seis arestas do tetraedro.

Essas métricas no espaço euclidiano tridimensional podem ser úteis, já que o custo de computar o circunraio da circunfera de um tetraedro pode ser alto.

### 3.4 Ângulos extremos

O erro de interpolação em limites de erros anteriores [20] em métodos numéricos aumenta na proporção em que o menor ângulo  $\theta_0$  do triângulo aproxima-se de zero [121]. Por outro lado, Gregory [72] indicou que o erro de interpolação aumenta na proporção em que o maior ângulo  $\theta_\infty$  do triângulo aproxima-se de  $\pi$ :  $\theta_0$  pode ser arbitrariamente pequeno se  $\theta_\infty$  não é muito próximo a  $\pi$ . Esse limite de erro leva ao critério de triangulação minmax [98]: escolhe-se a triangulação que minimiza o maior ângulo na triangulação.

Com isso, um dos meios mais comumente aceitos de métricas para avaliação da qualidade de triângulos é examinar  $\theta_0$  ou  $\theta_\infty$  [5]. Por definição desses ângulos, tem-se  $0 < \theta_0 \leq \frac{\pi}{3} \leq \theta_\infty$ , pois encontra-se o triângulo equilátero para  $\theta_0 = \theta_\infty = \frac{\pi}{3}$ . Além disso, tem-se

$$\theta_0 \leq \pi - \theta_0 - \theta_\infty \leq \theta_\infty, \quad (3.4.4)$$

que é equivalente a

$$\theta_\infty \leq \pi - 2\theta_0, \quad 2\theta_\infty \geq \pi - \theta_0. \quad (3.4.5)$$

As inequações 3.4.5 fornecem os limites de um ângulo máximo, dependendo de um ângulo mínimo, ou seja,

$$\left(\forall \theta_0 \in \left]0, \frac{\pi}{3}\right]\right) \frac{\pi - \theta_0}{2} \leq \theta_\infty \leq \pi - 2\theta_0. \quad (3.4.6)$$

*Demonstração.* Seja um triângulo com ângulos  $\theta_\infty$ ,  $\alpha$  e  $\theta_0$ . Sabe-se que a soma dos ângulos internos de um triângulo é igual a  $180^\circ$ , que em radianos, é equivalente a  $\pi$ . Dessa forma,  $\theta_\infty + \theta_0 + \alpha = \pi$ . Logo,  $\alpha = \pi - \theta_0 - \theta_\infty$ . Conseqüentemente, como se tem por hipótese que  $\theta_0$  é o menor ângulo,  $\theta_\infty$  o maior ângulo e  $\theta_0 \leq \alpha \leq \theta_\infty$ , então, tem-se a inequação  $\theta_0 \leq \pi - \theta_0 - \theta_\infty \leq \theta_\infty$ , em que a igualdade dessa inequação é satisfeita somente para triângulos equiláteros, o que prova a inequação 3.4.4. Assim, tem-se que  $\theta_0 \leq \pi - \theta_0 - \theta_\infty \Rightarrow \theta_\infty \leq \pi - 2\theta_0$ , o que prova a primeira inequação em 3.4.5, que é o lado direito da inequação 3.4.6. Para a primeira parte



da inequação 3.4.6, tem-se  $\pi - \theta_0 - \theta_\infty \leq \theta_\infty \Rightarrow \pi - \theta_0 \leq 2\theta_\infty \Rightarrow \frac{\pi - \theta_0}{2} \leq \theta_\infty$  e, ao se levar em consideração o primeiro e o terceiro termos da inequação 3.4.6, tem-se  $\frac{\pi}{2} - \frac{\theta_0}{2} \leq \pi - 2\theta_0 \Rightarrow 2\theta_0 - \frac{\theta_0}{2} \leq \pi - \frac{\pi}{2} \Rightarrow \frac{4\theta_0 - \theta_0}{2} \leq \frac{2\pi - \pi}{2} \Rightarrow 3\theta_0 \leq \pi$ , e as igualdades dessas inequações são satisfeitas somente para triângulos equiláteros.  $\square$

As inequações 3.4.4 e 3.4.5 também fornecem  $(\forall \theta_\infty \in [\frac{\pi}{3}, \frac{\pi}{2}]) \pi - 2\theta_\infty \leq \theta_0 \leq \frac{\pi - \theta_\infty}{2}$ . Portanto, tem-se  $(\forall \theta_\infty \in [\frac{\pi}{2}, \pi]) 0 < \theta_0 \leq \frac{\pi - \theta_\infty}{2}$ . Também é útil provar o resultado seguinte, antes de se examinar como as métricas de qualidade mais comumente utilizadas são relacionadas a esses ângulos extremos. Reproduz-se aqui o lema e a demonstração desse lema por Pébay e Baker [112].

**Lema 3.1.** *Os três ângulos de um triângulo não degenerado são ordenados na mesma ordem dos comprimentos de suas respectivas arestas opostas.*

*Demonstração.* Pode-se supor, sem perda de generalidade, que  $l_1 \leq l_2 \leq l_3$ . Consequentemente, pode-se deduzir, diretamente de 3.2.1, que  $\sin \alpha \leq \sin \beta \leq \sin \gamma$ . Com isso, tem-se  $\alpha < \frac{\pi}{2}$  e  $\beta < \frac{\pi}{2}$  e a função seno é monotonicamente crescente sobre  $]0, \frac{\pi}{2}]$ , assim, tem-se  $\alpha \leq \beta$ . Em relação a  $\gamma$ , dois casos podem ocorrer:

1. se o triângulo for agudo, então, o mesmo argumento pode ser utilizado;
2. se o triângulo for obtuso ou retângulo, então, claramente,  $\beta \leq \gamma$ .

Portanto, em qualquer caso, tem-se,  $\alpha \leq \beta \leq \gamma$ .  $\square$

**Corolário 3.1.** *Os tamanhos das arestas opostas aos ângulo  $\theta_0$  e  $\theta_\infty$  são  $|t|_0$  e  $|t|_\infty$ , respectivamente.*

Em particular, o ângulo diédrico pode ser considerado para avaliação da qualidade de tetraedros.

### 3.5 Razões entre raios das circunferências circunscrita e inscrita

A relação de raios é definida pela razão entre  $R$  e  $r$ , definida por

$$\rho_1 = \frac{R}{r}. \quad (3.5.7)$$

Ao combinar as inequações 3.2.1 e 3.2.2, obtém-se a expressão para  $\rho_1$  em termos dos ângulos dos triângulos  $\rho_1 = \frac{\sin \alpha + \sin \beta + \sin \gamma}{2 \cdot \sin \alpha \cdot \sin \beta \cdot \sin \gamma}$ .

*Demonstração.* Pela equação 3.2.1, sabe-se que  $2R = \frac{l_1 \cdot l_2 \cdot l_3}{2a}$ ,  $l_1 = 2R \cdot \sin \alpha$ ,  $l_2 = 2R \cdot \sin \beta$  e  $l_3 = 2R \cdot \sin \gamma$ . Logo, tem-se que

$$\begin{aligned} 2R &= \frac{2R \cdot \sin \alpha \cdot 2R \cdot \sin \beta \cdot 2R \cdot \sin \gamma}{2a} \\ &= \frac{8R^3 \cdot \sin \alpha \cdot \sin \beta \cdot \sin \gamma}{2a} \end{aligned}$$

implica em

$$\alpha = \frac{4R^3 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{2R} = 2R^2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma. \quad (3.5.8)$$

Pela equação 3.2.2, obtém-se  $r = \frac{a}{s}$  e, ao se substituir nessa equação as equações 3.5.8 e 3.2.3, obtém-se

$$r = \frac{2R^2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\frac{l_1 + l_2 + l_3}{2}}. \quad (3.5.9)$$

Pela equação 3.2.1, obtém-se

$$l_1 + l_2 + l_3 = 2R \cdot (\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma). \quad (3.5.10)$$

Substituindo-se a equação 3.5.10 na equação 3.5.9, obtém-se

$$r = \frac{4R^2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{2R \cdot (\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma)} = \frac{2R \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}. \quad (3.5.11)$$

Logo, utilizando-se a equação 3.5.7, obtém-se

$$\rho_1 = \frac{R}{\frac{2R \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}} = \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}{2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}.$$

□

Ao considerar  $\alpha \leq \beta \leq \gamma$  e como  $\alpha + \beta + \gamma = \pi$ , então, tem-se [26, p. 367]

$$\rho_1 = \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen}(\alpha + \beta)}{2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen}(\alpha + \beta)}.$$

Ainda, ao se considerar os ângulos extremos, tem-se

$$\rho_1 = \frac{\text{sen } \theta_0 + \text{sen } \theta_\infty + \text{sen}(\theta_0 + \theta_\infty)}{2 \cdot \text{sen } \theta_0 \cdot \text{sen } \theta_\infty \cdot \text{sen}(\theta_0 + \theta_\infty)}.$$

A métrica  $\rho_1$  atinge o valor mínimo somente para um triângulo equilátero [112]. A relação de raios é normalizada como  $\frac{\rho_1}{2}$  de forma que um triângulo equilátero seja 1.

A razão  $\rho_2 = \frac{r}{R}$  também pode ser utilizada. Essa métrica abrange o intervalo  $]0, \frac{1}{2}]$ , em que o valor máximo é obtido para um triângulo equilátero. Com isso, pode ser normalizada como  $2\rho_2$ .

### 3.6 Razões entre arestas

Outra abordagem bastante intuitiva é comparar as arestas extremas do triângulo por meio da *razão de arestas*, que é definida como  $\tau_1 = \frac{|t|_\infty}{|t|_0} = \frac{\text{sen } \theta_\infty}{\text{sen } \theta_0}$ , ao considerar o lema 3.1. Assim como  $\rho_1$ ,  $\tau_1$  é uma métrica adimensional. Além disso, é trivial ver que  $\tau_1 \geq 1$  e que a igualdade ocorre se e, somente se, o triângulo é equilátero. Consequentemente,  $\tau_1$  compartilha da propriedade essencial de  $\rho_1$ , de que seu valor mínimo é atingido apenas no triângulo equilátero.

Uma métrica baseada na razão  $\tau_2 = \frac{|t|_0}{|t|_\infty} = \frac{\text{sen } \theta_0}{\text{sen } \theta_\infty}$  também é possível. Similamente a  $\tau_1$ ,  $\tau_2$  também é uma métrica adimensional. Além disso, também é trivial ver que  $0 < \tau_2 \leq 1$  e que a igualdade ocorre se e, somente se, o triângulo é equilátero. Consequentemente,  $\tau_2$  atinge o valor máximo 1 apenas no triângulo equilátero. Com isso, quanto menor o valor de  $\tau_2$ , pior é a qualidade do triângulo.

### 3.7 Razões entre raio do circuncírculo e arestas

Considere as razões  $\omega_1 = \frac{R}{|t|_\infty}$  e  $\omega_2 = \frac{R}{|t|_0}$ . Então, a equação 3.2.1 e o lema 3.1 fornecem

$$\omega_1 = \frac{\theta_\infty}{2 \cdot \text{sen } \theta_\infty \cdot \theta_\infty} = \frac{1}{2 \cdot \text{sen } \theta_\infty}, \quad \omega_2 = \frac{\theta_0}{2 \cdot \text{sen } \theta_0 \cdot \theta_0} = \frac{1}{2 \cdot \text{sen } \theta_0} \quad (3.7.12)$$

As métricas  $\omega_1$  e  $\omega_2$  atingem seus únicos valores mínimos nos triângulos retângulo e equilátero, respectivamente. Com isso, torna-se claro que, exceto para aplicações particulares que requerem ângulos retos,  $\omega_1$  não é uma métrica de qualidade no sentido usual de atribuir uma relevância alta para triângulos equiláteros. Além disso, cada razão é relacionada com somente um ângulo extremo. Isso é especialmente problemático para  $\omega_1$ , que confunde triângulo que tem um e, somente um, ângulo próximo de zero com o bom triângulo retângulo. Em particular, o intervalo de  $\omega_2$  é de  $\frac{1}{\sqrt{3}} \cong 0,577$  a  $\infty$  para um triângulo equilátero. Claramente, quanto menor é  $\omega_2$ , maior é o menor ângulo do triângulo.

O principal problema com as duas razões em 3.7.12 é o fato que levam em conta somente uma aresta. Consequentemente, as boas propriedades de simetria de  $\rho_1$  e de  $\tau_1$  são perdidas. Com o objetivo de se evitar esse problema e obter-se uma quantidade adimensional, é natural utilizar o perímetro ou o semiperímetro. De fato, pode-se escrever a métrica

$$\nu = \frac{R}{s}. \quad (3.7.13)$$

Em seguida, ao substituir a equação 3.2.3 nessa equação, obtém-se  $\nu = \frac{2R}{l_1+l_2+l_3}$ . Ao substituir a equação 3.5.10 nessa equação, obtém-se

$$\nu = \frac{2R}{2R \cdot (\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma)} = \frac{1}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma},$$

que pode ser reescrita em termos de ângulos extremos como [26, p. 367]

$$\nu = \frac{1}{\text{sen } \theta_0 + \text{sen } \theta_\infty + \text{sen}(\theta_0 + \theta_\infty)},$$

em que o valor mínimo é encontrado se e, somente se, o triângulo equilátero é avaliado.

### 3.8 Razões entre arestas e raio de círculo inscrito

Uma métrica bastante utilizada para avaliação da qualidade de triângulos para estimativas de erros em elementos finitos é  $\iota = \frac{|t|_\infty}{r} 1$ . Esse valor varia de 1 a  $\infty$  e a qualidade do triângulo é maior com o valor mais próximo a 1.

---

<sup>1</sup>Frequentemente, em que Pebay e Baker [112] e Frey e George [56, p. 596] são exemplos, a métrica  $\frac{|t|_\infty}{r}$  é chamada de razão de aspecto (*aspect ratio*). Entretanto, não há nomenclatura uniforme em relação a essa expressão. Como exemplo, para Cheng, Dey e Shewchuk [32, p. 27], razão de aspecto é a métrica  $\rho_2 = \frac{r}{R}$ , mostrada na seção 3.5, na página 33. Também, é comum ver a expressão razão de aspecto associada com a razão entre os comprimentos de dois lados de um retângulo. Por isso, essa métrica é referenciada aqui simplesmente como  $\iota$  para evitar possíveis confusões.

Ao considerar a razão  $\omega_1 = \frac{R}{|t|_\infty}$ , mostrada na seção 3.7, e a equação 3.5.11, obtém-se

$$\iota = \frac{R}{\omega_1 \frac{2R \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}} = \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}{2\omega_1 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}.$$

Ao substituir a equação 3.7.12 nessa equação e, ao se levar em consideração o lema 3.1, obtém-se

$$\begin{aligned} \iota &= \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}{2 \frac{1}{2 \cdot \text{sen } \gamma} \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma} \\ &= \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}{\frac{1}{\text{sen } \gamma} \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma} \\ &= \frac{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}{\text{sen } \alpha \cdot \text{sen } \beta}. \end{aligned}$$

Em termos de ângulos extremos, tem-se [26, p. 367]

$$\iota = \frac{\text{sen } \theta_0 + \text{sen } \theta_\infty + \text{sen}(\theta_0 + \theta_\infty)}{\text{sen } \theta_0 \cdot \text{sen}(\theta_0 + \theta_\infty)}.$$

Finalmente, são comparados os comprimentos das arestas com o raio da circunferência inscrita (inraio) do triângulo. Das equações 3.5.7 e 3.7.13, pode-se obter  $\zeta = \frac{\rho_1}{\nu} = \frac{\frac{R}{K}}{\frac{s}{r}}$ . Ao substituir as equações 3.2.3 e 3.5.11 nessa equação, obtém-se

$$\zeta = \frac{l_1 + l_2 + l_3}{2 \frac{2R \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}}.$$

Em seguida, ao substituir a equação 3.5.10 nessa equação, obtém-se

$$\zeta = \frac{2R \cdot (\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma)}{2 \frac{2R \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma}{\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma}} = \frac{(\text{sen } \alpha + \text{sen } \beta + \text{sen } \gamma)^2}{2 \cdot \text{sen } \alpha \cdot \text{sen } \beta \cdot \text{sen } \gamma},$$

que em termos de ângulos extremos fica [26, p. 367]

$$\zeta = \frac{[\text{sen } \theta_0 + \text{sen } \theta_\infty + \text{sen}(\theta_0 + \theta_\infty)]^2}{2 \cdot \text{sen } \theta_0 \cdot \text{sen } \theta_\infty \cdot \text{sen}(\theta_0 + \theta_\infty)}.$$

A métrica  $\zeta$  atinge somente um valor mínimo: para triângulos equiláteros [111, 112].

### 3.9 Considerações finais

Entre as métricas estudadas, o valor mínimo de  $\omega_1$  determina que o triângulo de melhor qualidade é o triângulo retângulo. A métrica  $\omega_1$ , mostrada na seção 3.7, na página 35, é a razão entre o raio circunscrito (circunraio) e a maior aresta do triângulo.

Pela métrica  $\rho_2$ , mostrada na seção 3.5, na página 33, o valor máximo 1 determina o triângulo equilátero, que é o de melhor qualidade ao avaliar triângulos por

essa métrica. A métrica  $\rho_2$  é a razão do inraio pelo circunraio das circunferências inscrita e circunscrita do triângulo, respectivamente.

Pelas outras seis métricas mostradas na tabela 3.1, o valor mínimo determina que um triângulo equilátero é o triângulo de melhor qualidade. Mostra-se na tabela 3.1, adaptada de Pébay e Baker [111, 112], características de oito métricas para avaliação da qualidade de triângulos.

Métrica: razão entre	razão	norma- lização	valor para triângulo		triângulo de refe- rência	
			de boa qualidade	$t_{\angle \cong 0}$		$t_{\angle \cong \pi}$
arestas	$\tau_1 = \frac{ t _\infty}{ t _0}$	-	$\cong 1$	$\gg 1$	$\cong 2$	$\triangle$
circunraio e semiperímetro	$\nu = \frac{R}{s}$	$\frac{3\sqrt{3}}{2}\nu$		$\cong \frac{3\sqrt{3}}{4}$	$\gg 1$	
raios $\rho_2$	$\rho_2 = r/R$	$2\rho_2$		$\cong 0$		
raios $\rho_1$	$\rho_1 = R/r$	$\rho_1/2$				
maior aresta e inraio	$\iota = \frac{ t _\infty}{r}$	$\frac{1}{2\sqrt{3}}\iota$		$\gg 1$		
semiperímetro e inraio	$\zeta = \frac{s}{r}$	$\frac{1}{3\sqrt{3}}\zeta$				
circunraio e menor aresta	$\omega_2 = \frac{R}{ t _0}$ $= \frac{1}{2 \cdot \text{sen } \theta_0}$	$\sqrt{3}\omega_2$				
circunraio e maior aresta	$\omega_1 = \frac{R}{ t _\infty}$ $= \frac{1}{2 \cdot \text{sen } \theta_\infty}$	$2\omega_1$		$\cong 1$	$\gg 1$	$\nabla$

**Tabela 3.1:** Valores qualitativos de oito métricas de qualidade para três categorias de triângulos [111, 112]. São utilizadas as notações  $t_{\angle \cong 0}$  e  $t_{\angle \cong \pi}$  para representar, respectivamente, dois tipos de triângulos não degenerados, mas de má qualidade: *i*) triângulo que tem um e, somente um, ângulo próximo de zero; *ii*) triângulo que tem um ângulo próximo a  $\pi$ .

A métrica a ser utilizada é dependente da aplicação. Características possíveis para se definir qual métrica deve ser utilizada, em geral, são quais informações já estão disponíveis no projeto computacional, custos de processamento e de armazenamento e facilidades de implementação e de manutenção do código computacional.

### 3.10 Notas bibliográficas

Além de detalhes e demonstrações de várias das métricas de qualidade de triângulos apresentadas neste texto, Pébay e Baker [111, 112] apresentam também detalhes sobre métricas baseadas em normas de matrizes. Os livros *Mesh Generation, application to Finite Elements* de Frey e George [56, p. 596-598] e *Delaunay Mesh Generation* de Cheng, Dey e Shewchuk [32, p. 26-27] são fontes complementares sobre métricas de qualidade de triângulos e de tetraedros.

### 3.11 Exercícios

1. Explique as características de um triângulo agudo e de um triângulo obtuso.
2. Explique o que é ângulo diédrico.

3. Explique o que é função monotonicamente crescente.
4. Explique por que polítopos com ângulos próximos a  $0^\circ$  ou  $\pi$  são considerados de má qualidade.
5. Calcule a métrica  $\mathfrak{b}$ , mostrada na seção 3.3, para um triângulo com as três arestas com tamanho 1.
6. Calcule a métrica  $f_1$ , mostrada na seção 3.3, para um triângulo equilátero cujo tamanho da aresta é 2.
7. Calcule a métrica  $f_3$ , mostrada na seção 3.3, para um triângulo equilátero cujo tamanho da aresta é 2.
8. Calcule a métrica  $\omega_1 = \frac{R}{|t|_\infty} = \frac{1}{2 \cdot \text{sen } \theta_\infty}$  para:
  - (a)  $\theta_\infty = 120^\circ$ .
  - (b) o triângulo retângulo.
9. Calcule a métrica  $\omega_2 = \frac{R}{|t|_0} = \frac{1}{2 \cdot \text{sen } \theta_0}$  para:
  - (a)  $\theta_0 = 20^\circ$ .
  - (b)  $\theta_0 = 25^\circ$ .
  - (c)  $\theta_0 = 30^\circ$ .
  - (d)  $\theta_0 = 32^\circ$ .
  - (e) o triângulo equilátero e multiplique o resultado por  $\sqrt{3}$ .
  - (f) o triângulo retângulo isósceles.
10. Qual das métricas de avaliação de qualidade de triângulos mostradas neste capítulo é a mais fácil de ser compreendida? Por quê?
11. Qual das métricas de avaliação de qualidade de triângulos mostradas neste capítulo é a mais fácil de ser calculada? Por quê?

## Capítulo 4

# Triangulação de Delaunay e diagrama de Voronoi

### 4.1 Introdução

Neste capítulo, abordam-se malhas importantes: a triangulação de Delaunay [44] e o seu dual, o diagrama de Voronoi [150]. Essas duas estruturas são importantes porque possuem propriedades geométricas desejáveis, em especial, pode-se garantir a qualidade da malha se os algoritmos adequados são utilizados.

Tesselações de Delaunay são utilizadas para representar partes do espaço contínuo de uma forma que permite a algoritmos numéricos computar características do espaço [49]. A popularidade dessas malhas deve-se, principalmente, por poderem ser construídas com baixo custo computacional e por possuírem características geométricas bastante atrativas; por exemplo, o diagrama de Voronoi pode capturar proximidade. Essas duas malhas são aplicadas em diversas áreas. Algumas dessas áreas são citadas na tabela 4.1 [70].

Área	Exemplo de publicação
Computação gráfica	Alliez, Meyer e Desbrun [1]
Projetos industriais	Nordin et al. [103]
Aplicações médicas	Puentes et al. [119]
Modelagem de materiais compósitos e porosos	Dong e Atluri [47]
Modelagem de objetos deformáveis	Busaryev, Dey e Wang [25]
Modelagem molecular	Lin, Wang e Zeng [88]
Modelagem de terrenos	Tucker et al. [144]
Vídeo games	Gyves, Toledo e Rudomín [74]

**Tabela 4.1:** Algumas das áreas em que a triangulação de Delaunay é utilizada e exemplos de publicações.

Para a sequência deste capítulo, é útil relembrar alguns conceitos relacionados a triângulos. Como descrito na subseção 2.2, na página 18, *circuncentro* é a intersecção das mediatrizes das arestas de um triângulo e é equidistante aos três vértices do triângulo. Essa distância é o raio, ou *circunraio*, de uma circunferência circunscrita ao triângulo, isto é, que tange seus vértices. Isso significa que o circuncentro

é o centro do *circuncírculo* do triângulo, que é o único círculo que passa pelos três vértices do triângulo. Em termos simples, a *mediatriz* de uma aresta é a reta perpendicular a essa aresta que passa exatamente no ponto médio da aresta. Portanto, a mediatriz de uma aresta de um triângulo é a reta que passa pelo ponto médio de uma aresta do triângulo e pelo circuncentro desse triângulo. Em particular, a extensão do conceito de circuncírculo do triângulo no espaço euclidiano tridimensional é expressa como *circunfera* do tetraedro, isto é, em termos simples, a circunfera de um tetraedro é a esfera cuja superfície contém os quatro vértices do tetraedro.

Em uma triangulação de Delaunay, não há vértices internos ao circuncírculo de cada triângulo. Os vértices e os *pontos geradores* do diagrama de Voronoi são os circuncentros e os vértices dos triângulos da triangulação de Delaunay, respectivamente. Essas características são estendidas ao espaço tridimensional. A seguir, a triangulação de Delaunay e suas propriedades são apresentadas na seção 4.2 e o diagrama de Voronoi é apresentado na seção 4.3.

## 4.2 Triangulação de Delaunay

Considere a definição 1.9 (triangulação de um conjunto de pontos), na página 11 e, em seguida, a definição de triangulação de Delaunay a seguir.

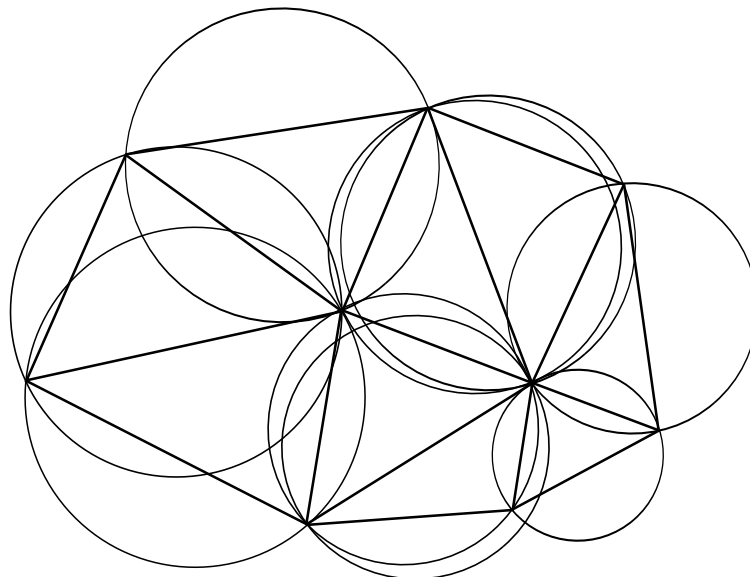
**Definição 4.1.** (*triangulação de Delaunay*) Uma triangulação  $T$  de um conjunto de pontos  $V$  é de Delaunay se qualquer ponto de  $V$  não consta no interior de qualquer circuncírculo de triângulos que compõem  $T$ .

Em três dimensões, uma *tesselação* de um conjunto de pontos  $V$  é de Delaunay se qualquer ponto de  $V$  não consta no interior de qualquer circunfera de tetraedros que compõem a tesselação. Em mais detalhes, uma triangulação (tesselação no espaço tridimensional) de Delaunay ( $TD$ ) para um conjunto de pontos  $V$  deve satisfazer à condição de Delaunay, que é:  $TD(V)$  é uma tesselação de Delaunay, tal que nenhum ponto de  $V$  permanece no interior do circuncírculo (circunfera no espaço tridimensional) de qualquer triângulo (tetraedro no espaço tridimensional) em  $TD(V)$ . Isso se aplica em dimensões arbitrárias. Note que convencionou-se chamar de triangulação de Delaunay em um domínio bidimensional e tesselação de Delaunay quando se referir tanto a um domínio bidimensional quanto tridimensional.

Nem toda triangulação é de Delaunay, mas qualquer conjunto de pontos possui uma triangulação de Delaunay. Isso é verdade porque uma forma de transformar uma triangulação em uma triangulação de Delaunay é por meio de trocas de arestas, que são abordadas na subseção 4.2.2. A triangulação de Delaunay produz triângulos que são, inerentemente, isotrópicos (veja uma explicação sobre malhas isotrópicas na subseção 1.5.2, na página 8). Como exemplo, na figura 4.1, observa-se uma triangulação de Delaunay para um conjunto de nove pontos. Descrições e propriedades de triangulações de Delaunay apresentadas nesta seção também são encontradas em Nogueira e Gonzaga de Oliveira [102, 101, 100].

A seguir, o conceito de circuncírculos vazios da triangulação de Delaunay é abordado na subseção 4.2.1. O conceito de arestas locais da triangulação de Delaunay é abordado na subseção 4.2.2. Propriedades e características da triangulação de Delaunay são descritas na subseção 4.2.3. A triangulação de Delaunay restrita é abordada na subseção 4.2.4. Algoritmos para a triangulação de Delaunay são citados na subseção 4.2.5. Refinamento de Delaunay é abordado na subseção 4.2.6.





**Figura 4.1:** Triangulação de Delaunay para um conjunto de nove pontos. Não devem haver pontos interiores aos circuncírculos dos triângulos para que seja satisfeita a condição de Delaunay. (Figura adaptada de [102].)

Tesselações de Delaunay tridimensionais com tetraedros com vértices quase coplanares são abordadas na subseção 4.2.7. A triangulação de Pitteway é abordada na subseção 4.2.8.

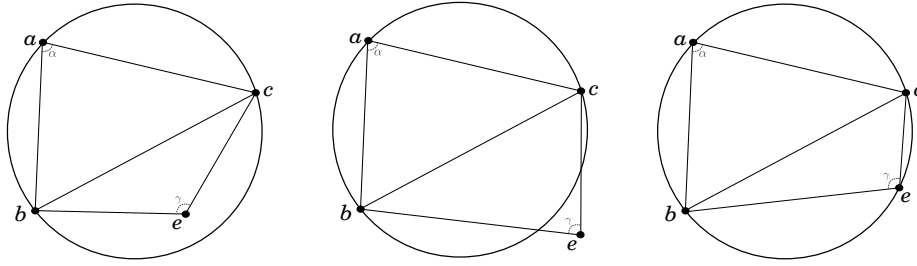
### 4.2.1 Circuncírculos vazios

Não é óbvio que o conjunto de arestas de Delaunay de um conjunto de vértices formará uma triangulação. Por isso, é utilizada a expressão *triângulo de Delaunay*. Um triângulo é dito ser de Delaunay se e, somente se, seu circuncírculo é vazio. Essa propriedade é denominada *propriedade do circuncírculo vazio*. Um círculo é dito ser *vazio*, em relação a um conjunto  $V$  de vértices (ou pontos), se nenhum vértice de  $V$  é *interior* a esse círculo (aberto).

Como descrito, pela *condição do circuncírculo vazio*, para que ocorra uma triangulação de Delaunay, é necessário que cada circuncírculo de cada triângulo não contenha pontos em seu interior. Na figura 4.2(a), observa-se um ponto interior ao circuncírculo do triângulo  $\triangle abc$ . Na figura 4.2(b), observa-se a condição do circuncírculo vazio aplicada no triângulo  $\triangle abc$  sem pontos no interior do triângulo.

A aplicação prática da condição do circuncírculo vazio é escolher as duas combinações que determinem dois triângulos cuja soma dos ângulos mínimos seja a máxima. A seguir, são mostradas duas maneiras de implementação da condição do circuncírculo vazio.

1. Soma dos ângulos opostos a uma aresta. Considere uma aresta  $\overline{bc}$  compartilhada pelos triângulos  $\triangle abc$  e  $\triangle bce$ . Quando realizada a soma dos ângulos opostos a  $\overline{bc}$ , digamos  $\alpha$  e  $\gamma$ , são três as classificações possíveis dessa soma, mostradas a seguir.



(a) Não há uma triangulação de Delaunay porque o ponto  $e$  é interior ao circuncírculo do triângulo  $\triangle abc$ : a soma dos ângulos  $\alpha$  e  $\gamma$  é maior que  $180^\circ$ .

(b) Condição do circuncírculo vazio aplicada sem pontos no interior no triângulo  $\triangle abc$ ; logo, ocorre a triangulação de Delaunay: a soma dos ângulos  $\alpha$  e  $\gamma$  é menor que  $180^\circ$ .

(c) Ocorre a triangulação de Delaunay: a soma dos ângulos  $\alpha$  e  $\gamma$  é igual a  $180^\circ$ .

**Figura 4.2:** Três casos de testes de circuncírculo vazio da triangulação de Delaunay. Note que deve-se testar se o circuncírculo do triângulo  $\triangle bce$  é vazio para se determinar se ocorre uma triangulação de Delaunay nas imagens do meio e à direita.

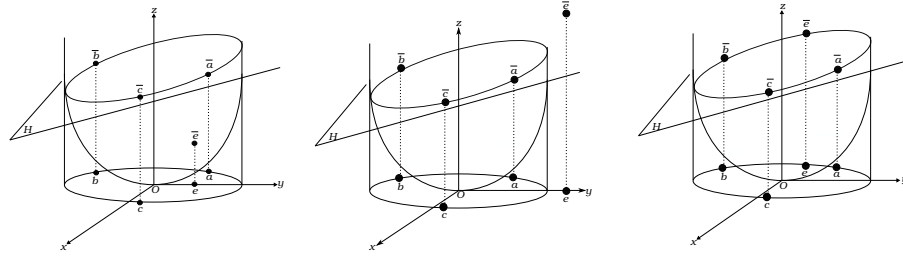
- Quando a soma dos ângulos  $\alpha$  e  $\gamma$  for maior que  $180^\circ$ , não haverá uma triangulação de Delaunay. Essa propriedade é exemplificada na figura 4.2(a): a soma dos ângulos  $\alpha$  e  $\gamma$  resulta em um ângulo maior que  $180^\circ$ .
- Quando a soma dos ângulos  $\alpha$  e  $\gamma$  for menor que  $180^\circ$ , haverá uma triangulação de Delaunay. Essa propriedade é exemplificada na figura 4.2(b): a soma dos ângulos  $\alpha$  e  $\gamma$  é menor que  $180^\circ$ .
- Quando a soma dos ângulos  $\alpha$  e  $\gamma$  for igual a  $180^\circ$ , os pontos são cocirculares e haverá uma triangulação de Delaunay. Esse caso é exemplificado na figura 4.2(c).

2. Resolução por determinante. Esse método foi apresentado por Guibas e Stolfi [73] e baseia-se nas coordenadas dos pontos  $a$ ,  $b$ ,  $c$  e  $e$ , ao se calcular

$$\text{Determinante}(a,b,c,e) = \begin{vmatrix} x_a & y_a & x_a^2 + y_a^2 & 1 \\ x_b & y_b & x_b^2 + y_b^2 & 1 \\ x_c & y_c & x_c^2 + y_c^2 & 1 \\ x_e & y_e & x_e^2 + y_e^2 & 1 \end{vmatrix}.$$

São três os resultados possíveis do determinante, mostrados a seguir.

- $\text{Determinante}(a,b,c,e) > 0$ : o quarto ponto  $e$  é interior ao circuncírculo do triângulo  $\triangle abc$ . Nesse caso, não ocorre a triangulação de Delaunay. Esse caso é exemplificado na figura 4.3(a).
- $\text{Determinante}(a,b,c,e) < 0$ : o quarto ponto  $e$  é exterior ao circuncírculo do triângulo  $\triangle abc$ . Nesse caso, ocorre a triangulação de Delaunay. Na figura 4.3(b), observa-se o ponto  $e$  exterior ao circuncírculo do triângulo  $\triangle abc$ .
- $\text{Determinante}(a,b,c,e) = 0$ : o quarto ponto  $e$  pertence ao circuncírculo do triângulo  $\triangle abc$ . Nesse caso, ocorre a triangulação de Delaunay. Esse caso é observado na figura 4.3(c).



(a)  $Determinante(a,b,c,e) > 0$ : o ponto  $e$  está interno ao circuncírculo do triângulo  $\triangle abc$  e o ponto  $\bar{e}$  está abaixo do plano  $H$  que passa pelo triângulo  $\triangle \bar{a}\bar{b}\bar{c}$ .  
 (b)  $Determinante(a,b,c,e) < 0$ : o ponto  $e$  está externo ao circuncírculo do triângulo  $\triangle abc$  e o ponto  $\bar{e}$  está acima do plano  $H$  que passa pelo triângulo  $\triangle \bar{a}\bar{b}\bar{c}$ .  
 (c)  $Determinante(a,b,c,e) = 0$ : os pontos  $a, b, c$  e  $e$  são cocirculares e o plano  $H$  passa pelos pontos  $\bar{a}, \bar{b}, \bar{c}$  e  $\bar{e}$ .

**Figura 4.3:** Exemplos de testes de circuncírculo vazio da triangulação de Delaunay por meio do cálculo de  $Determinante(a,b,c,e)$ . Considere  $\bar{a} = (a_x, a_y, a_x^2 + a_y^2)$ ,  $\bar{b} = (b_x, b_y, b_x^2 + b_y^2)$ ,  $\bar{c} = (c_x, c_y, c_x^2 + c_y^2)$ ,  $\bar{e} = (e_x, e_y, e_x^2 + e_y^2)$  e  $H$  é um plano que passa por  $\bar{a}, \bar{b}, \bar{c}$  e, no terceiro caso,  $H$  também passa pelo ponto  $\bar{e}$ . Essas figuras são baseadas em Vigneron [149] [102].

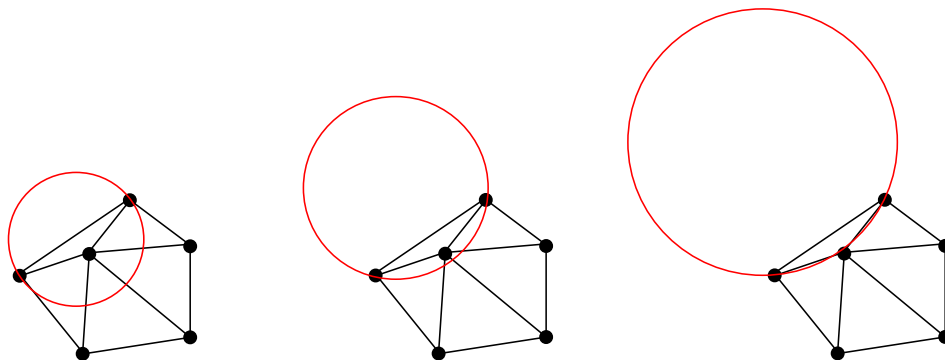
Mais detalhes sobre esse assunto e outras formas de implementação da condição do circuncírculo vazio são apresentados por Shewchuk [132].

## 4.2.2 Arestas locais de Delaunay

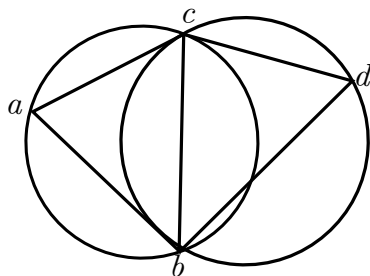
Uma triangulação é estritamente de Delaunay se todas as arestas que a compõem são *arestas locais* de Delaunay. Ao se referir a uma aresta ser localmente de Delaunay, significa que apenas os vértices dos triângulos que compartilham a aresta em questão são considerados. Uma aresta de Delaunay é local se ao menos uma das duas situações seguintes ocorre [48].

- A aresta pertence somente a um triângulo e à fronteira da triangulação. Em particular, um triângulo na fronteira da triangulação será sempre um triângulo de Delaunay. Isso porque o circuncírculo que passa sobre qualquer triângulo da fronteira da triangulação nunca conterà um ponto da triangulação em seu interior [100]. Isso faz com que triângulos de má qualidade possam existir na fronteira da triangulação de Delaunay. Qualquer aresta da fronteira de uma triangulação é localmente de Delaunay, pois o circuncírculo do triângulo que a aresta pertence é vazio, como exemplificado na figura 4.4.
- Uma aresta  $\bar{bc}$  é localmente de Delaunay se é compartilhada por dois triângulos, digamos  $\triangle abc$  e  $\triangle bcd$ , e os pontos  $d$  e  $a$  não estão no interior dos circuncírculos dos triângulos  $\triangle abc$  e  $\triangle bcd$ , respectivamente. A aresta  $\bar{bc}$  na figura 4.5 é uma aresta local de Delaunay.

Outra possibilidade de se verificar se uma aresta é localmente de Delaunay, é verificar se existe um círculo vazio que passa pelos vértices da aresta. Isso significa que a aresta  $\bar{ab}$  é dita ser de *Delaunay* e estará presente na triangulação de Delaunay se e, somente se, existe um círculo que passa por  $a$  e  $b$  que é vazio. Esse é o círculo diametral da aresta. Veja definição de círculo diametral na definição 4.2, na página 52.



**Figura 4.4:** Exemplo de busca por um círculo vazio de uma aresta na fronteira da triangulação. (Figuras extraídas de [106].)

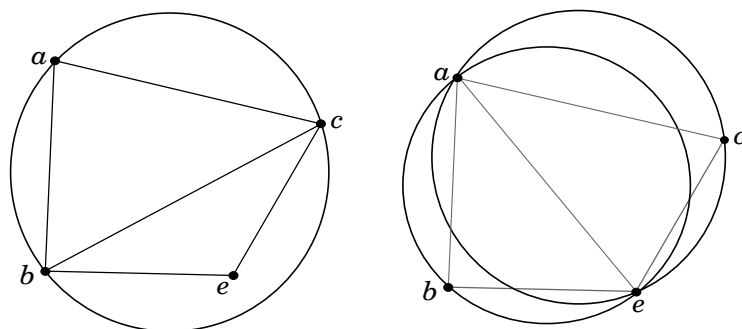


**Figura 4.5:** A aresta  $\overline{bc}$  é uma aresta local de Delaunay.

Se uma aresta não é localmente de Delaunay, uma forma de transformá-la em uma aresta válida para a triangulação de Delaunay é a troca dessa aresta no quadrilátero em que essa aresta faz parte. A *troca de arestas* é a transformação topológica mais simples em uma triangulação: substitui-se dois triângulos adjacentes por outros dois triângulos adjacentes.

A troca de arestas é realizada para melhorar a regularidade dos dois triângulos: ao trocar uma aresta que não é localmente de Delaunay, o ângulo mínimo é aumentado, bem como o maior circuncírculo entre os dois triângulos alterados é reduzido. A conectividade dos dois triângulos é alterada se o ângulo mínimo que ocorre nos dois novos triângulos é maior que o ângulo mínimo que ocorre nos dois triângulos originais.

Como exemplo, considere os triângulos  $\triangle abc$  e  $\triangle bce$  na figura 4.6(a). Para a realização da *troca de arestas*, verifica-se o quadrilátero formado pelos pontos  $a$ ,  $b$ ,  $c$  e  $e$ . Se a aresta  $\overline{bc}$  não é uma aresta local de Delaunay, a aresta  $\overline{bc}$  é trocada pela aresta  $\overline{ae}$ . Caso a troca nas arestas melhore localmente a triangulação, a aresta original é considerada ilegal na triangulação de Delaunay. Na figura 4.6(b), é observado um exemplo de troca de aresta inválida para a triangulação de Delaunay. Ao realizar a troca de aresta inválida no quadrilátero, obtém-se uma aresta local de Delaunay. Embora mais difícil, em três dimensões, também é possível melhorar a qualidade da malha por meio da implementação de um procedimento de troca de arestas.

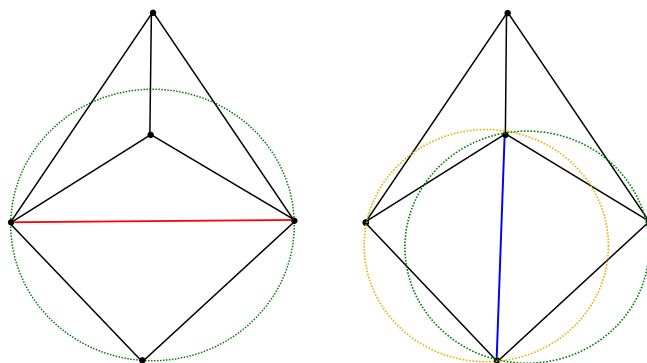


(a) Não há uma triangulação de Delaunay porque o ponto  $e$  é interior ao circuncírculo do triângulo  $\triangle abc$ .

(b) Condição do circuncírculo vazio aplicada sem pontos nos interiores dos triângulos  $\triangle abe$  e  $\triangle ace$ .

**Figura 4.6:** Exemplo de troca de arestas na triangulação de Delaunay: a aresta ilegal  $\overline{bc}$  da figura à esquerda é trocada pela aresta  $\overline{ae}$ . Conseqüentemente, na figura à direita, ocorre a maximização dos ângulos mínimos, diminuição dos circuncírculos e a triangulação de Delaunay.

Mostra-se outro exemplo de troca de arestas na figura 4.7. A aresta em vermelho na imagem à esquerda não é localmente de Delaunay e é trocada pela aresta em azul na imagem à direita, obtendo-se uma triangulação de Delaunay.



**Figura 4.7:** Exemplo de troca de arestas na triangulação de Delaunay. Há um triângulo que não satisfaz a condição do circuncírculo vazio na imagem à esquerda e a aresta em vermelho não é localmente de Delaunay. Na imagem à direita, a aresta em vermelho da imagem à esquerda foi trocada pela aresta em azul. Com isso, a aresta em azul é uma aresta local de Delaunay. (Figuras extraídas de [104].)

### 4.2.3 Propriedades

Nesta subseção, são apresentadas propriedades [102, 101, 100] e características da triangulação de Delaunay. As propriedades são abordadas sem provas. Algumas dessas propriedades são baseadas em Barth [10, pág. 20]. Provas de propriedades são apresentadas por Cheng, Dey e Shewchuk [32]. Veja as referências citadas para detalhes sobre cada propriedade.

**Propriedade 4.1.** *Todo circuncírculo na triangulação de Delaunay é vazio.*

Essa propriedade é descrita em detalhes na subseção 4.2.1.

**Propriedade 4.2.** *Na triangulação de Delaunay, maximiza-se o ângulo mínimo de todo triângulo da triangulação [94, 84, 133].*

Essa pode ser considerada a principal vantagem da triangulação de Delaunay. Essa propriedade pode ser obtida em toda a triangulação por meio de troca das arestas inválidas. Essa propriedade é considerada adequada para interpolação [86, 152, 38] principalmente porque o erro de interpolação em limites de erros anteriores em métodos numéricos [20] aumenta na proporção em que o menor ângulo do triângulo aproxima-se de zero [121]. Entretanto, essa propriedade não é estendida para um domínio tridimensional [32, p. 85].

**Propriedade 4.3.** *Seja  $T$  uma triangulação. Se todos os triângulos de  $T$  são de Delaunay, então, todas as arestas de  $T$  são de Delaunay, e vice-versa.*

**Propriedade 4.4.** *Se todas as arestas de uma triangulação são localmente de Delaunay, então, todas as arestas são de Delaunay (globalmente).*

**Propriedade 4.5.** *Seja  $\overline{ab}$  uma aresta de uma triangulação  $T$ : ou  $\overline{ab}$  é localmente de Delaunay, ou  $\overline{ab}$  pode ser trocada no quadrilátero formado pelos dois triângulos que compartilham a aresta  $\overline{ab}$  e a aresta criada pela troca é localmente de Delaunay.*

Essa propriedade foi descrita em detalhes na subseção 4.2.2.

**Propriedade 4.6.** *Se não há aresta invadida em uma triangulação de Delaunay, então, não há circuncentro que caia fora do domínio.*

**Propriedade 4.7.** *A triangulação de Delaunay possui unicidade, exceto em casos em que quatro ou mais pontos são cocirculares.*

Isso significa que, quando houver exatamente quatro pontos no mesmo circuncírculo, a triangulação de Delaunay não é única. Para um caso como esse, as duas triangulações possíveis satisfazem à condição de Delaunay.

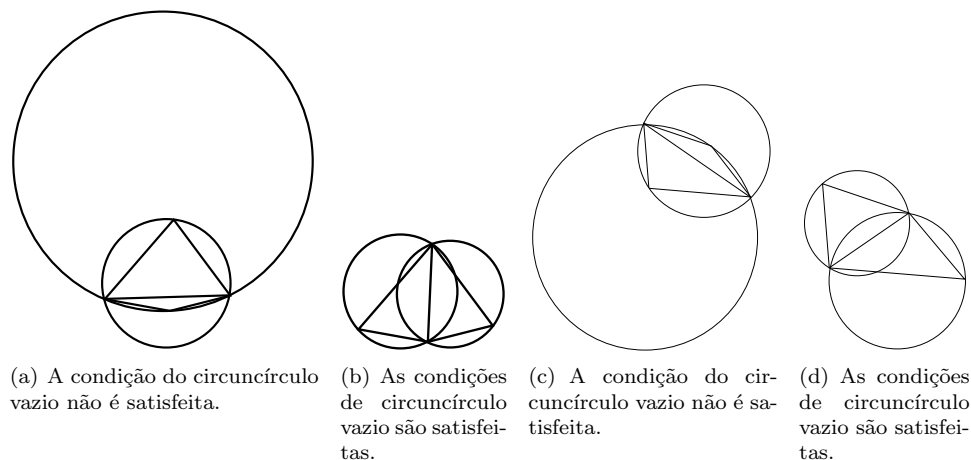
**Propriedade 4.8.** *A triangulação de Delaunay é uma malha irregular. Entretanto, apresenta uma triangulação com irregularidade mínima.*

Veja o artigo de Rippa [121], para detalhes sobre essa propriedade.

**Propriedade 4.9.** *Com a obtenção de triângulos com ângulos mínimos maximizados, também ocorre a minimização dos circuncírculos dos triângulos da triangulação.*

Essa propriedade foi chamada de *propriedade do confinamento do círculo mínimo* por Barth [10, pág. 21]. Essa propriedade é exemplificada na figura 4.8. Não ocorrem triangulações de Delaunay nas figuras 4.8(a) e 4.8(c), pois há triângulos que não satisfazem à condição de circuncírculo vazio. Note nas figuras 4.8(b) e 4.8(d) que, após as trocas de arestas, triangulações de Delaunay são obtidas e os circuncírculos dos triângulos de Delaunay são minimizados.

Como nas triangulações de Delaunay há a minimização dos circuncírculos dos triângulos, conseqüentemente, os raios máximos dos circuncírculos contidos na triangulação também são minimizados [36]. Essa caracterização foi generalizada para dimensões maiores que dois por Rajan [120].



**Figura 4.8:** Exemplos da propriedade do confinamento do círculo mínimo. Não ocorrem triangulações de Delaunay em (a) e (c) porque há triângulos que não satisfazem à condição do circuncírculo vazio. Ocorrem triangulações de Delaunay em (b) e (d) e os circuncírculos dos triângulos são minimizados após as trocas das arestas inválidas em ambos os casos. (Figuras extraídas de [102, 100].)

**Propriedade 4.10.** *Na propriedade do vizinho mais próximo, tem-se que uma aresta formada ao se ligar um vértice ao seu vizinho mais próximo é uma aresta da triangulação de Delaunay.*

Essa propriedade faz com que a triangulação de Delaunay possa ser utilizada para resolver o problema do vizinho mais próximo. Por outro lado, as arestas do vizinho mais próximo não descrevem todas as arestas da triangulação de Delaunay [10, p. 22].

**Propriedade 4.11.** *Triangulações de Delaunay maximizam a média aritmética dos raios inscritos dos triângulos [82].*

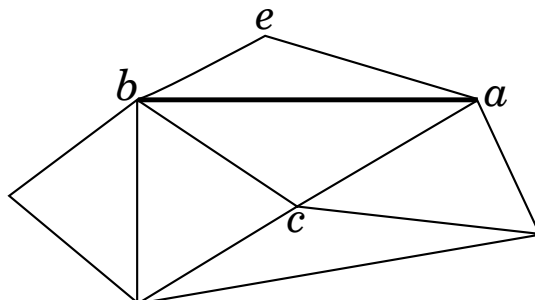
Ainda, em uma triangulação de Delaunay ótima, há uma minimização do erro de interpolação em relação a todas as triangulações possíveis de serem geradas com o mesmo número de vértices. Veja o artigo de Chen e Holst [31], para detalhes sobre esquemas eficientes para otimização de malhas baseadas em triangulações de Delaunay ótimas.

Para finalizar esta subseção, é válido mencionar que a triangulação de Delaunay ponderada é uma generalização da triangulação de Delaunay. Cheng, Dey e Shewchuk [32, p. 43-44] apresentam detalhes sobre a triangulação de Delaunay ponderada.

#### 4.2.4 Triangulação de Delaunay restrita

Em uma triangulação de Delaunay *restrita*, certas arestas são descritas *a priori*, chamadas de *arestas restritas*. Essas arestas restritas podem ser os segmentos de um PSLG, explicado na seção 1.6, na página 10. Uma triangulação de Delaunay de um conjunto de pontos  $V$  é restrita se e, somente se, para qualquer vértice  $e$  que é interno ao circuncírculo de um triângulo formado por vértices de  $V$ , então, o vértice  $e$  não é *visível* por, pelo menos, um dos vértices desse triângulo. Um vértice  $e$  não

é visível por um vértice  $c$  se, ao traçar um segmento de reta entre os vértices  $e$  e  $c$ , então, esse segmento de reta intercepta ao menos uma aresta restrita. Na figura 4.9, o ponto  $e$  não é visível pelo vértice  $c$  por causa da aresta restrita  $\overline{ba}$ .



**Figura 4.9:** O vértice  $e$  não é visível pelo vértice  $c$  por causa da aresta restrita  $\overline{ba}$ .

A triangulação de Delaunay restrita foi formalizada matematicamente por Lee e Lin [85]. Esses autores também estenderam, para a triangulação de Delaunay restrita, a prova de Lawson [84] de que a triangulação de Delaunay maximiza o ângulo mínimo.

#### 4.2.5 Algoritmos para a triangulação de Delaunay

Há uma quantidade enorme de algoritmos eficientes para a geração da triangulação de Delaunay. Diversos desses algoritmos são abordados ou citados nos capítulos 1, 16 e 20 do livro editado por Thompson, Soni e Weatherhill [143] e também são citados por Nogueira [100].

De acordo com Cheng, Dey e Shewchuk [32, p. 10], provavelmente, o primeiro algoritmo para a geração da triangulação de Delaunay, bem como o primeiro *método de avanço de fronteira* (*advancing front method*), foi proposto no artigo de Frederick, Wong e Edge [52]. Segundo Cheng, Dey e Shewchuk [32, p. 10], aparentemente, os autores não tinham conhecimento que seu algoritmo gerava a triangulação de Delaunay restrita. A seguir, como exemplo de algoritmo para geração da triangulação de Delaunay, o algoritmo de Lawson é descrito, e algoritmos com comportamento linear são citados em seguida.

##### Algoritmo de Lawson

Um dos primeiros algoritmos propostos para triangulação de Delaunay foi o algoritmo de Lawson [84]. Esse algoritmo é  $O(n^2)$  para  $n$  pontos no plano.

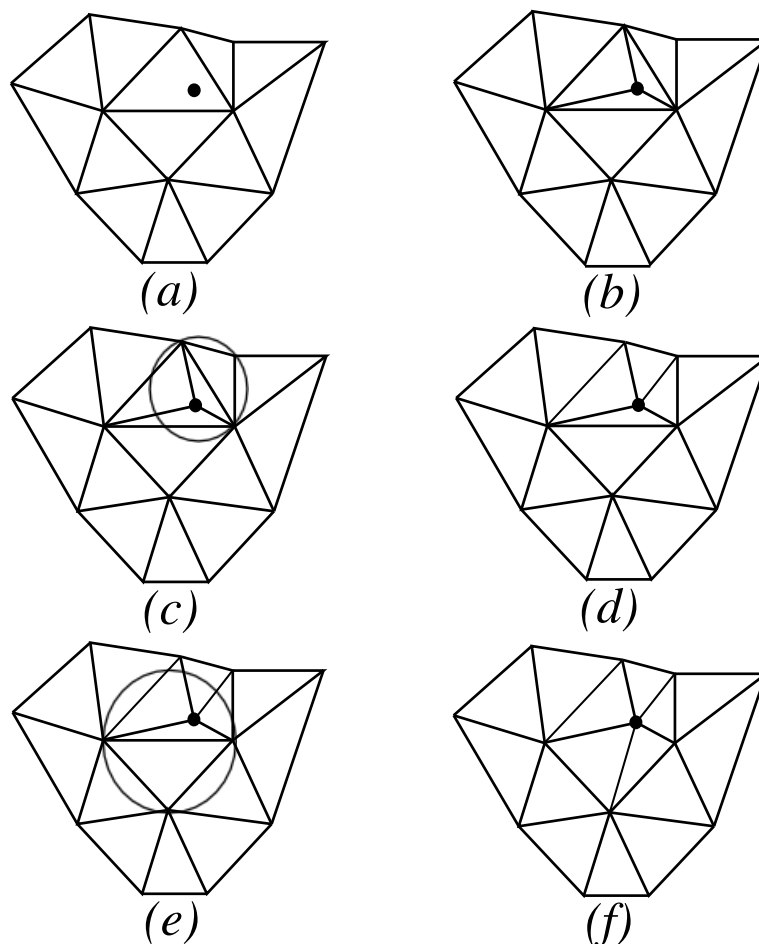
O algoritmo de Lawson é incremental. Algoritmos por inserção incremental são bem simples. Algoritmos projetados por essa técnica inserem um vértice por vez e a cada inserção são realizadas operações para manter a triangulação de Delaunay. O algoritmo proposto por Lawson [84] provavelmente foi o primeiro algoritmo incremental para a geração da triangulação de Delaunay.

O algoritmo de Lawson é baseado na troca de arestas. A troca de aresta foi abordada na subseção 4.2.2, na página 43, e exemplificada nas figuras 4.6 e 4.7, na página 45. A partir de uma triangulação qualquer, realiza-se a troca de arestas para se obter uma triangulação de Delaunay no algoritmo de Lawson. Essa triangulação qualquer deve abranger todo o domínio, de forma que novos vértices a serem inseridos na triangulação sempre pertencerão a algum triângulo da triangulação.



Três arestas são inseridas na triangulação a cada novo vértice  $v$  inserido em um triângulo  $\triangle abc$ :  $\overline{va}$ ,  $\overline{vb}$  e  $\overline{vc}$ . Se  $v$  não pertence ao circuncírculo de nenhum triângulo da nova triangulação, as três novas arestas inseridas são de Delaunay e obtém-se uma nova triangulação de Delaunay. Realiza-se a troca de arestas para cada uma das três arestas que não atende à condição de Delaunay.

Na figura 4.10, mostra-se um exemplo da aplicação do algoritmo de Lawson. Nesse algoritmo, quando um novo vértice é inserido, o triângulo que o contém é encontrado, como exemplificado na figura 4.10(a). Então, três novas arestas são inseridas na triangulação ao se ligar o novo vértice aos três vértices desse triângulo, como exemplificado na figura 4.10(b). Desse modo, o triângulo original é substituído por três novos triângulos.



**Figura 4.10:** Um exemplo de aplicação do algoritmo de Lawson. (a) Triangulação de Delaunay inicial e vértice que será inserido. (b) O novo vértice é ligado aos três vértices do triângulo que o contém. (c) Uma aresta que não é de Delaunay é detectada. (d) A troca de arestas é aplicada para remover a aresta que não é de Delaunay e inserir uma aresta válida na triangulação de Delaunay. (e) Outra aresta que não satisfaz à condição de Delaunay é encontrada. (f) A troca de arestas é aplicada novamente. A triangulação final é de Delaunay. Os circuncírculos mostrados nas figuras (c) e (e) são utilizados para mostrar que a condição de Delaunay não foi atendida. (Figuras extraídas de [106].)

Em seguida, um procedimento, que pode ser recursivo, verifica se as novas arestas (ou os novos triângulos) são localmente de Delaunay. Se alguma aresta não for localmente de Delaunay, como exemplificado na figura 4.10(c), então, a troca de arestas é aplicada para remover a aresta que não é de Delaunay e inserir uma aresta válida na triangulação de Delaunay, como exemplificado na figura 4.10(d). Duas novas arestas (ou triângulos) devem ser testadas a cada troca de arestas (veja a figura 4.10(e)). O procedimento termina quando não houver mais aresta oposta ao novo vértice que não seja de Delaunay (veja a figura 4.10(f)).

Em geral, o algoritmo de Lawson é utilizado para manutenção da triangulação de Delaunay, após a malha ter sofrido adaptações. Uma das vantagens do algoritmo de Lawson é a facilidade de ser implementado. Um pseudocódigo da versão do algoritmo de Lawson para ser utilizado para a verificação ou conversão de uma triangulação para uma triangulação de Delaunay é mostrado no algoritmo 1.

---

**Algoritmo 1:** Lawson.
 

---

**Entrada:** triangulação  $T$ .

**Saída:** triangulação de Delaunay  $T$ .

```

1 início
2   para (cada triângulo  $\Delta abc \in T$ ) faça
3     // condição do circuncírculo vazio
4     // ao considerar  $\Delta abd$  como o
5     // triângulo que compartilha com o
6     // triângulo  $\Delta abc$  a aresta inválida  $\overline{ab}$ 
7     se (vértice  $d$  está dentro do circuncírculo do triângulo  $\Delta abc \in T$ )
8       então
9         Troque  $\overline{ab}$  por  $\overline{cd}$ ; // realiza a troca da aresta
10      fim-se;
11    fim-para;
12  retorna  $T$ ;
13 fim.
```

---

Agora, considere que o vértice  $p$  é inserido no triângulo  $\Delta abd$ . Com isso, as arestas  $\overline{ap}$ ,  $\overline{bp}$  e  $\overline{dp}$  são inseridas na triangulação  $T$ . Precisa-se verificar se o vértice  $p$  invade os circuncírculos de três triângulos, formados pelas arestas  $\overline{ab}$ ,  $\overline{ad}$  e  $\overline{bd}$ , respectivamente. Com isso, três verificações são realizadas, ao executar um procedimento como o mostrado no algoritmo 2:  $\text{Verifica}(p, \overline{ab}, T)$ ,  $\text{Verifica}(p, \overline{ad}, T)$  e  $\text{Verifica}(p, \overline{bd}, T)$ .

O algoritmo de Bowyer-Watson [19, 151] é, em geral, mais utilizado que o algoritmo de Lawson. Pelo algoritmo de Bowyer-Watson, ao inserir um vértice  $v$  na triangulação que invade circuncírculo(s) de triângulo(s), os triângulos com circuncírculos invadidos são excluídos e obtém-se um polígono em que  $v$  é interno. Com isso, liga-se  $v$  ao vértices desse polígono.

### Algoritmos com comportamento linear

Uma revisão em que são citados 34 algoritmos com comportamento linear para a geração da tesselação de Delaunay foi apresentado por Gonzaga de Oliveira, Nogueira e Tavares [70]. Os algoritmos mais rápidos atualmente são algoritmos incrementais que utilizam eficientemente a hierarquia de memória da arquitetura atual dos sistemas de computação [70]. Mais especificamente, a utilização eficiente da

---

**Algoritmo 2:** Verifica.

---

**Entrada:** vértice  $p$ , aresta  $\overline{ab}$ , triangulação  $T$ , em que  $\{\triangle abp, \triangle abd, \triangle ade, \triangle bdf\} \subset T$ .

**Saída:** triangulação de Delaunay  $T$ .

```

1 início
2    $c \leftarrow \text{Circuncentro}(\triangle abd)$ ;
3   se ( $\|p, c\| < \text{TamanhoCircunraio}(\triangle abd)$ ) então
4     // o vértice  $p$  está dentro do
5     // circuncírculo do triângulo  $\triangle abd$ 
6     Troque a aresta  $\overline{ab}$  por  $\overline{pd}$  em  $T$ ;
7     Verifica( $p, \overline{ad}, T$ ); Verifica( $p, \overline{bd}, T$ );
8   fim-se;
9   retorna  $T$ ;
10 fim.
```

---

memória *cache* é fundamental. Esses algoritmos inserem os pontos em uma determinada sequência ao utilizar eficientemente a localidade de memória. Isso pode ser obtido ao se utilizar a sequência dada por uma curva de preenchimento de espaço. Três curvas de preenchimento de espaço bem conhecidas são as curvas de Peano [110], de Hilbert [76] e de Sierpiński [134]. Como exemplos, a curva de Hilbert foi utilizada por Liu e Snoeyink [90] e as curvas de Peano, de Hilbert e de Sierpiński foram utilizadas por Schrijvers, van Bommel e Buchin [127, 128], no projeto de algoritmos com comportamento linear para a geração da tesselação de Delaunay.

Outra sequência de inserção adequada pode ser obtida ao se utilizar uma estrutura de dados, como a *kd-tree* [12]. Especificamente, Liu, Yan e Lo [89] propuseram um algoritmo incremental para a geração da tesselação de Delaunay tridimensional, em que a inserção é realizada na sequência dada por uma variação da *kd-tree*. A *kd-tree* é uma árvore  $k$ -dimensional, ou seja, é uma generalização da árvore binária de busca. A cada nível de uma *kd-tree*, é utilizada uma das coordenadas como índice. A alteração na *kd-tree* proposta por Liu, Yan e Lo [89] é que, a cada ramificação da árvore, a coordenada utilizada é aquela que tiver a maior diferença em uma coordenada entre os pontos em consideração.

Por sua vez, Lo [91] propôs um algoritmo incremental para a geração da triangulação de Delaunay em que a inserção é realizada na sequência dada por um esquema multi-*grid*. Em particular, as simulações apresentadas por Lo [91] foram com até 100 milhões de pontos bidimensionais.

É importante ser mencionado que o procedimento para encontrar o polítopo que contém o novo vértice inserido na tesselação deve apresentar baixo custo computacional. O *walking procedure* é uma implementação padrão em algoritmos para a geração da tesselação de Delaunay. Para detalhes sobre esse assunto, veja a seção 5.5 de Cheng, Dey e Shewchuk [32]. Em particular, muitas aplicações utilizam a biblioteca CGAL [140] para implementar algoritmos geométricos.

Como descrito, os algoritmos incrementais têm alcançado os melhores resultados atualmente entre os algoritmos com comportamento linear para a geração da triangulação de Delaunay. Por outro lado, outra técnica muito utilizada no projeto de algoritmos para a geração da triangulação de Delaunay é a divisão e conquista. Provavelmente, há centenas de algoritmos por divisão e conquista para a geração da triangulação de Delaunay. Um exemplo de algoritmo recente por divisão

e conquista, com comportamento linear, para a geração da tesselação de Delaunay tridimensional, é o algoritmo proposto por Yang, Choi e Jung [154].

#### 4.2.6 Refinamento de Delaunay

Uma triangulação de Delaunay pode ser refinada e a qualidade da malha deve ser mantida. Refinamento de Delaunay é a noção de se manter uma triangulação de Delaunay ao se inserir vértices em posições determinadas pela própria triangulação.

As vantagens de algoritmos para refinamento de Delaunay são que as propriedades da triangulação de Delaunay são mantidas e que são projetados para terem garantias matemáticas, isto é, sempre geram uma malha válida [32, p. 11]. Nesta subseção, dois algoritmos para o refinamento de Delaunay são abordados: algoritmo de Ruppert [124, 125, 126] e algoritmo de Ünğör [145, 146], que podem ser considerados dos melhores algoritmos para refinamento de Delaunay [65].

##### Algoritmo de Ruppert

O algoritmo de Ruppert [126] garante que todos os triângulos pertencentes à triangulação terão ângulos entre  $\alpha$  e  $\pi - 2\alpha$ , de forma que  $\alpha$  pode ser um ângulo máximo de, aproximadamente,  $20,7^\circ$ . Ao se inserir um vértice na triangulação, duas posições são possíveis: inserir o ponto médio de uma aresta ou inserir o circuncentro do circuncírculo de um triângulo.

Um algoritmo para a geração da triangulação de Delaunay pode utilizar uma métrica de qualidade. Uma dessas métricas de qualidade é  $\omega_2 = \frac{R}{|t|_0}$ , mostrada na seção 3.7, na página 35, que é a razão entre o circunraio  $R$  e a menor aresta do triângulo  $|t|_0$ . É especificado um limite superior para a métrica  $\omega_2$  para todos os triângulos da triangulação. Como descrito na seção 3.7, a razão  $\omega_2$  de um triângulo é relacionada com seu menor ângulo  $\alpha$  pela fórmula  $\omega_2 = \frac{1}{2 \cdot \text{sen } \alpha}$ . Quanto menor for a razão  $\omega_2$ , maior será o menor ângulo  $\alpha$  do triângulo. Um limite superior para a métrica  $\omega_2$  garante que não há triângulo na malha com ângulo menor que  $\arcsen \frac{1}{2\omega_2}$  [113].

Uma aresta possui infinitos círculos que tangem os vértices dessa aresta, como exemplo, veja a figura 4.4. Por outro lado, considere a definição de *círculo diametral* a seguir.

**Definição 4.2** (círculo diametral). *O círculo diametral de uma aresta  $\overline{ac}$  é o círculo que passa pelos vértices  $a$  e  $c$  e o diâmetro do círculo é o comprimento da aresta  $\overline{ac}$ .*

Isso significa que o círculo diametral de uma aresta  $\overline{ac}$  é o círculo com menor diâmetro que passa pelos vértices  $a$  e  $c$ . O algoritmo de Ruppert pode receber um grafo planar de linhas não curvas (*Planar Straight Line Graph* - PSLG). Um PSLG é um conjunto de vértices e seus segmentos. Aqui, segmentos são arestas que não podem ser removidas. Claramente, os segmentos não se interceptam. Um segmento é considerado invadido quando um vértice incide dentro de seu círculo diametral. Um pseudocódigo para o algoritmo de Ruppert [126] é apresentado no algoritmo 3.

Nas situações em que o PSLG possui ângulos menores que  $90^\circ$ , o algoritmo tende a formar triângulos com ângulos muito agudos. Shewchuk [131] provou que o algoritmo parará para qualquer entrada com ângulos de, no mínimo,  $60^\circ$ .

Ruppert [126] provou que o seu algoritmo sempre parará para qualquer  $\alpha < 20,7^\circ$ . Todavia, o algoritmo se mostra eficiente para ângulos de até, aproximadamente,  $30^\circ$ .

**Algoritmo 3:** Algoritmo de Ruppert.

---

**Entrada:** PSLG  $G = (L_V, L_S)$ , em que  $L_V$  é um conjunto de vértices e  $L_S$  é um conjunto de segmentos; ângulo mínimo  $\alpha$ ;  $n \in \mathbb{N}^{n+}$ , em que  $n = 3$  é utilizado por padrão.

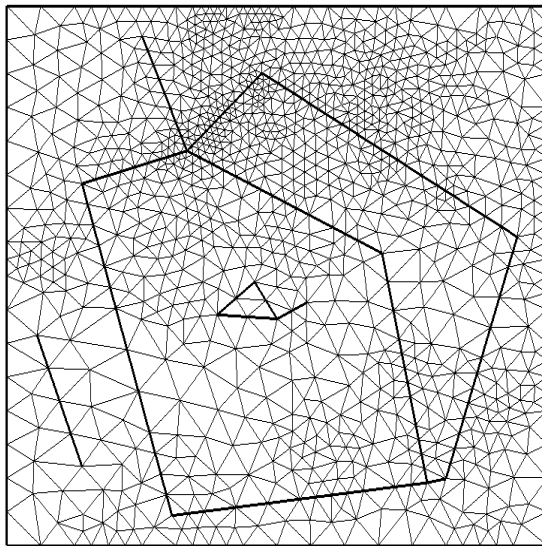
**Saída:** triangulação de Delaunay de  $TD = (T_V, T_S)$ , com todos os  $\angle \geq \alpha$ .

```

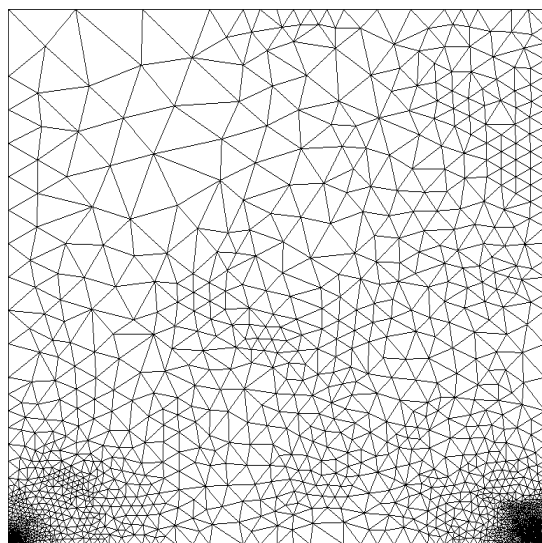
1 início
  // as coordenadas extremas de  $G = (L_V, L_S)$  são
  // identificadas como:  $x_{min}, y_{min}, x_{max}, y_{max}$ 
2  $span(G) = \max(x_{max} - x_{min}, y_{max} - y_{min})$ ;
  // adiciona-se um delimitador (bounding box)
  //  $D$  no PSLG  $G = (L_V, L_S)$ 
3 Seja  $D$  o delimitador de lado  $n \times span(G)$ , e considere que o PSLG
   $G = (L_V, L_S)$  está centralizado em  $D$ ;
4 Adicione os quatro segmentos de contorno de  $D$  no PSLG  $G = (L_V, L_S)$ ;
5 Construa a triangulação de Delaunay inicial  $TD = (T_V, T_S)$ ;
6 repita
  // os círculos diametraais dos segmentos
  // são verificados e são inseridos os
  // pontos médios dos segmentos invadidos
7 enquanto (existe segmento  $\overline{ab} \in L_S$  invadido) faça
  // insere ponto médio  $m$  do segmento
  // invadido  $\overline{ab}$  em  $T_V$ , e insere as
  // arestas  $\overline{ma}, \overline{mb}, \overline{md}, \overline{me}$  em  $T_S$ ,
  // considerando-se que  $\triangle abd, \triangle abe \subset TD$ 
8   InserePontoMédio( $\overline{ab}$ );
9 fim-enquanto;
  // verifica triângulos com  $\angle < \alpha$ 
10 enquanto (existe triângulo  $\triangle abd \subset TD$  com  $\angle < \alpha$ ) faça
11   se (o circuncentro  $c$  do triângulo  $\triangle abd \subset TD$  invade o segmento
    $\overline{ab} \in T_S$ ) então
   // insere ponto médio  $m$  do
   // segmento invadido  $\overline{ab}$  em  $T_V$ ,
   // e insere as arestas  $\overline{ma}, \overline{mb}, \overline{md}, \overline{me}$  em  $T_S$ ,
   // considerando-se que  $\triangle abe \subset TD$ 
12   InserePontoMédio( $\overline{ab}$ );
13   senão
   // insere circuncentro  $c$  em  $T_V$  e insere
   // em  $T_S$  três arestas, com pontos
   // terminais em  $c$  e com os três pontos
   // do triângulo em que  $c$  é interior
14   InsereCircuncentro( $\triangle abd$ );
15   fim-se;
16 fim-enquanto;
17 até (não há segmento invadido e não ocorre  $\angle < \alpha$ );
18 retorna  $TD$ ;
19 fim.
```

---

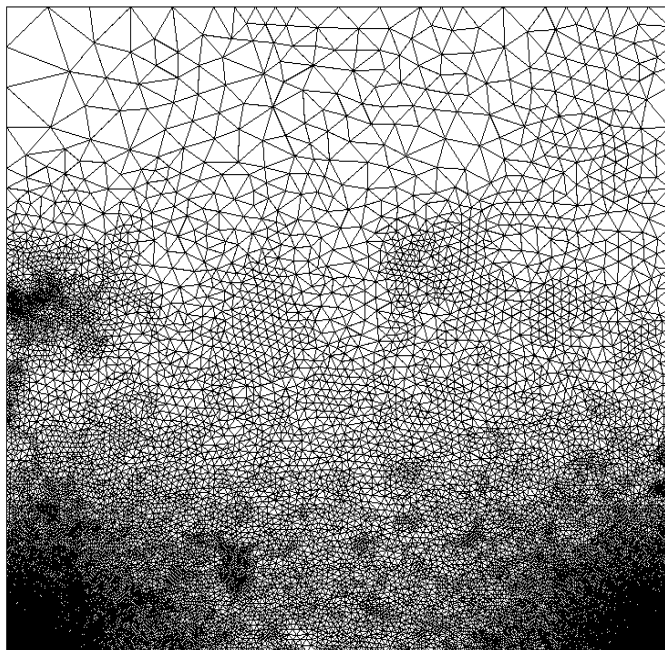
Mostra-se um exemplo de triangulação de Delaunay gerada pelo algoritmo de Ruppert na figura 4.11, em que a entrada foram os segmentos em negrito. Nas figuras 4.12 e 4.13, são mostradas triangulações de Delaunay geradas pelo algoritmo de Ruppert. Os ângulos mínimos são de  $33^\circ$  nas triangulações de Delaunay mostradas nas figuras 4.11, 4.12 e 4.13.



**Figura 4.11:** Exemplo de triangulação de Delaunay gerada pelo algoritmo de Ruppert, com ângulo mínimo de  $33^\circ$ . Arestas em negrito representam segmentos do PSLG de entrada para o algoritmo. (Figura extraída de [106].)



**Figura 4.12:** Triangulação de Delaunay com 6514 vértices e ângulo mínimo de  $33^\circ$ , gerada pelo algoritmo de Ruppert. (Figura extraída de [106].)



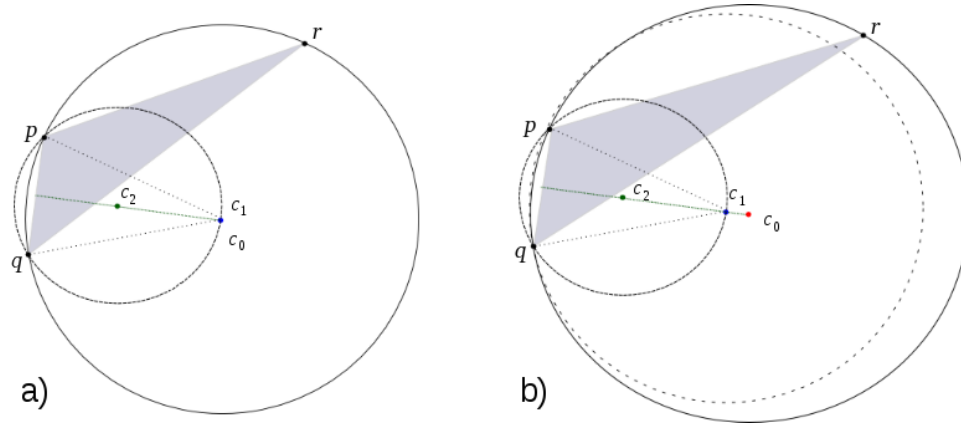
**Figura 4.13:** Triangulação de Delaunay com 65859 vértices e ângulo mínimo de  $33^\circ$ , gerada pelo algoritmo de Ruppert. (Figura extraída de [107].)

### Algoritmo de Üngör

Üngör [145, 146] propôs um refinamento de Delaunay similar ao algoritmo de Ruppert [126]. No algoritmo de Üngör [145, 146], são inseridos pontos chamados de *off-centers* em vez dos circuncentros, como é realizado no algoritmo de Ruppert. Considere a métrica  $\omega_2 = \frac{R}{|t|_0}$ , que é a razão entre o circunraio  $R$  e a menor aresta  $|t|_0$  do triângulo, mostrada na seção 3.7, na página 35. Se um triângulo é considerado de má qualidade ao ser avaliado pela métrica  $\omega_2$ , então, o algoritmo tenta inserir o ponto *off-center*. Caso o *off-center* invada alguma aresta, então, o novo vértice é inserido no ponto médio dessa aresta em vez do *off-center*.

Para calcular o *off-center*, considera-se um triângulo  $\triangle pqr$  de má qualidade. Considere que a menor aresta e o circuncentro de um triângulo  $\triangle pqr$  são  $\overline{pq}$  e  $c_0$ , respectivamente. Se o triângulo  $\triangle pqr$  tem duas arestas menores de mesmo tamanho, escolhe-se uma delas, arbitrariamente. O *off-center*  $c_1$  do triângulo  $\triangle pqr$  é o seu circuncentro se o valor da métrica  $\omega_2$  calculado no triângulo  $\triangle pqc_0$  for menor ou igual a um dado limite  $\omega_\alpha$ , em que  $\omega_\alpha$  é a métrica  $\omega_2$  de um ângulo  $\alpha$ , conforme mostrado na figura 4.14(a). Caso contrário, o *off-center*  $c_1$  será o ponto na mediatriz da aresta  $\overline{pq}$ , dentro do circuncírculo, que faz com que a métrica  $\omega_2$  do triângulo  $\triangle pqc_1$  seja exatamente igual a  $\omega_\alpha$ . O círculo que passa pelos vértices da menor aresta  $\overline{pq}$ , centrado no *off-center*  $c_1$ , é chamado de *off-circle*. O *off-circle* com *off-center*  $c_1$  é mostrado em linha tracejada na figura 4.14(b).

Como exemplo, considere que um usuário definiu que todos os ângulos da triangulação tenham ângulo mínimo  $\alpha$  de  $30^\circ$  em sua triangulação de Delaunay. Portanto,  $w_\alpha = \frac{1}{2 \cdot \text{sen } \alpha} = 1$ . (Claramente, foi escolhido  $\alpha = 30^\circ$  convenientemente



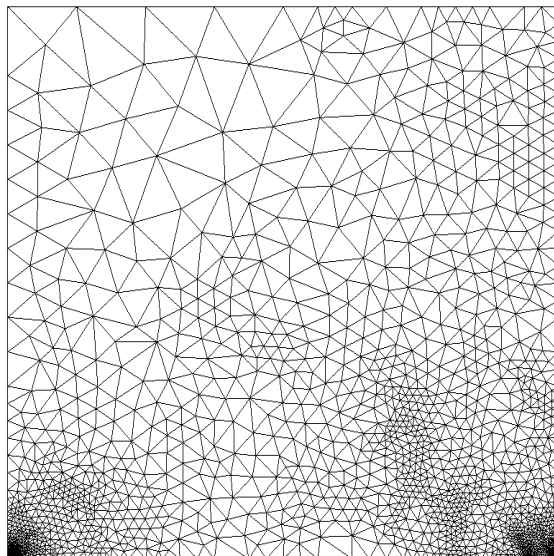
**Figura 4.14:** O circuncentro do triângulo  $\Delta pqr$  é  $c_0$ . O *off-center* do triângulo  $\Delta pqr$  é  $c_1$ , porque a menor aresta do triângulo  $\Delta pqr$  é  $\overline{pq}$ . O circuncentro do triângulo  $\Delta pqc_1$  é  $c_2$ . Se  $|\overline{c_1c_2}| \leq \omega_\alpha |\overline{pq}|$ , então,  $c_1 = c_0$ , como mostrado em (a); caso contrário,  $c_1 \neq c_0$ . Se  $|\overline{c_1c_2}| > \omega_\alpha |\overline{pq}|$ , então, calcula-se  $c_2$  de forma que  $|\overline{c_1c_2}| = \omega_\alpha |\overline{pq}|$ , como mostrado em (b). O *off-circle* do triângulo  $\Delta pqr$  é o circuncírculo do triângulo  $\Delta pqr$  (porque  $c_0 = c_1$ ) em (a), e é mostrado em linha tracejada em (b): *off-circle* é o círculo que tem como circuncentro o *off-center*  $c_1$  e que passa pela menor aresta  $\overline{pq}$ . As figuras são adaptadas de Üngör [145, 146] [104].

para se obter  $\omega_2 = 1$ .) Considere, novamente, um triângulo  $\Delta pqr$  de má qualidade, com menor aresta  $\overline{pq}$  e circuncentro  $c_0$ . Considere que, ao se inserir um vértice no circuncentro  $c_0$  do triângulo  $\Delta pqr$ , como mostrado na figura 4.14(a), haveria ângulo do triângulo menor que  $30^\circ$ . Por isso, em vez de inserir o circuncentro de  $\Delta pqr$ , Üngör definiu que o ponto a ser inserido deveria estar um pouco mais próximo da aresta  $\overline{pq}$  do que é a distância do circuncentro  $c_0$  para a aresta  $\overline{pq}$ , de forma que a métrica  $\omega_2$  do novo triângulo resulte exatamente no valor dado pelo ângulo mínimo definido pelo usuário. Um lugar geométrico adequado para se inserir esse novo ponto é na mediatriz de  $\overline{pq}$ , claramente, interior ao circuncírculo de  $\Delta pqr$ . Suponha que o usuário escolha inserir o ponto  $s_1$  mais próximo da aresta  $\overline{pq}$  do que o circuncentro  $c_0$ . Suponha que o menor ângulo de  $\Delta pqs_1$  seja, por exemplo,  $25^\circ$ . Isso faz com que o valor da métrica  $\omega_2$  do triângulo  $\Delta pqs_1$  seja  $\cong 1,18$ . Como o usuário deseja que o valor da métrica  $\omega_2$  do novo triângulo seja igual a 1, o usuário desiste de inserir  $s_1$  e decide inserir um novo ponto  $s_2$  (na mediatriz de  $\overline{pq}$ , dentro do circuncírculo de  $\Delta pqr$ ) um pouco mais próximo da aresta  $\overline{pq}$  do que é  $s_1$  em relação à aresta  $\overline{pq}$  (para que o valor da métrica  $\omega_2$  seja menor que 1,18). Suponha que o menor ângulo de  $\Delta pqs_2$  seja, por exemplo,  $27^\circ$ . Isso faz com que o valor da métrica  $\omega_2$  do triângulo  $\Delta pqs_2$  seja  $\cong 1,1$ . Com isso, o usuário também desiste de inserir  $s_2$  e decide inserir um novo ponto  $s_3$  um pouco mais próximo da aresta  $\overline{pq}$  do que é  $s_2$  em relação à aresta  $\overline{pq}$ . Suponha que o menor ângulo de  $\Delta pqs_3$  seja, por exemplo,  $29^\circ$ . Isso faz com que o valor da métrica  $\omega_2$  do triângulo  $\Delta pqs_3$  seja  $\cong 1,03$ . Com isso, o usuário percebe que precisa inserir o novo ponto ainda um pouco mais próximo da aresta  $\overline{pq}$  para obter o valor  $\omega_2 = 1$  do novo triângulo. O lugar geométrico em que deve inserir o novo ponto para que a métrica  $\omega_2$  seja igual a 1 deve ser exatamente em  $c_1$ , o ponto chamado de *off-center* por Üngör.

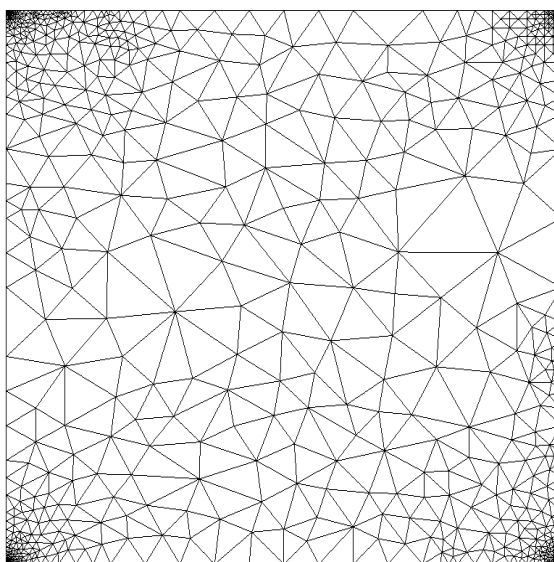
Üngör [145, 146] mostrou que o seu algoritmo possui as mesmas garantias que o algoritmo de Ruppert. Seus experimentos mostraram que o seu algoritmo in-



sere aproximadamente 40% menos vértices e as malhas tornam-se 30% menores em relação ao número de polítopos em comparação com algoritmos baseados na inserção de circuncentros, como o algoritmo de Ruppert. Nas figuras 4.15 e 4.16, são mostradas triangulações de Delaunay com ângulo mínimo de  $33^\circ$ , geradas pelo algoritmo de inserção de *off-centers* de Üngör.

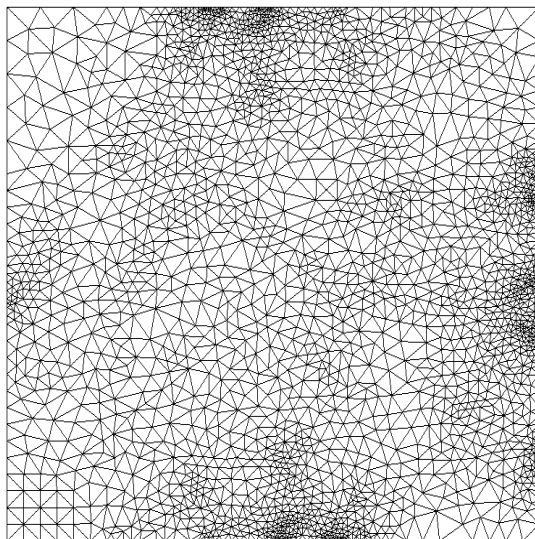


**Figura 4.15:** Triangulação de Delaunay com 5356 vértices, ângulo mínimo de  $33^\circ$ , gerada pelo algoritmo de inserção de *off-centers* de Üngör. (Figura extraída de [106].)

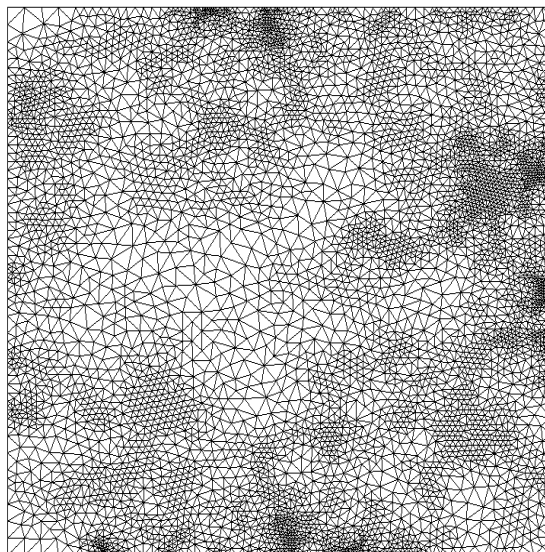


**Figura 4.16:** Triangulação de Delaunay e ângulo mínimo de  $33^\circ$ , gerada pelo algoritmo de inserção de *off-centers* de Üngör.

Nas figuras 4.17 e 4.18, são mostradas triangulações de Delaunay com 2316 e 5555 vértices e com ângulos mínimos de  $30^\circ$  e  $32^\circ$ , respectivamente. Essas triangulações de Delaunay foram geradas utilizando-se tanto o refinamento por inserção de circuncentros de Ruppert quanto o algoritmo de inserção de *off-centers* de Ünör.



**Figura 4.17:** Triangulação de Delaunay com 2316 vértices e ângulo mínimo de  $30^\circ$ , gerada pelos algoritmos de Ruppert e de Ünör.



**Figura 4.18:** Triangulação de Delaunay com 5555 vértices e ângulo mínimo de  $32^\circ$ , gerada pelos algoritmos de Ruppert e de Ünör.

### 4.2.7 Tetraedros com vértices quase coplanares

Um *sliver* é um tetraedro cujos vértices são quase coplanares e com circunraio (de sua circunfera) que não é muito maior que sua menor aresta. Ao considerar  $|t|_0$  como a menor aresta,  $r$  o inraio e  $R$  o circunraio da circunfera de um *sliver*, esse tetraedro pode ser considerado de boa qualidade ao ser avaliado pela métrica  $\omega_2 = \frac{R}{|t|_0}$ , mostrada na seção 3.7, na página 35; mas é considerado um tetraedro de má qualidade ao ser avaliado pela métrica  $\rho_2 = \frac{r}{R}$ , mostrada na seção 3.5, na página 33. Cheng, Dey e Shewchuk [32, p. 26] explicam que um *sliver* pode ser avaliado como  $\omega_2 = \frac{R}{|t|_0} = \frac{1}{\sqrt{2}} \cong 0,71$ , que é um valor próximo do valor  $\sim 0,61$  do tetraedro regular. Entretanto, *sliver* pode ser considerado problemático em outras métricas por causa do seu volume e altitude pequenas, e com ângulos diédricos que podem ser pequenos (próximos a  $0^\circ$ ) ou grandes (próximos a  $180^\circ$ ).

*Slivers* podem ser considerados como um dos tipos de tetraedros mais problemáticos. Se ocorrem *slivers* em uma tesselação, a precisão da simulação numérica pode ser prejudicada ou a solução numérica pode não convergir para a solução do problema.

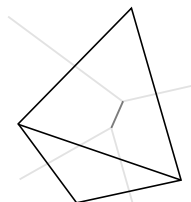
Cavendish, Field e Frey [30] perceberam que *slivers* são frequentes em tesselações de Delaunay (tridimensionais). Talmor [136] percebeu que tesselações de Delaunay possuem *slivers* mesmo com pontos com espaçamentos adequados. Com isso, uma das principais dificuldades na geração de malhas tridimensionais é evitar a presença de *slivers* [70]. Cheng, Dey e Shewchuk [32] apresentam detalhes sobre esse assunto e sobre algoritmos para a geração de malhas tridimensionais que evitam a presença de *slivers*.

### 4.2.8 Triangulação de Pitteway e grafo de Gabriel

As arestas de uma triangulação de Pitteway [114] de um conjunto de pontos  $V$  são formadas ao ligar cada vértice  $q \in V$  mais próximo de cada ponto  $p \in V$ , com  $p \neq q$ . Esse é um caso especial da triangulação de Delaunay em que cada aresta interna cruza uma aresta do diagrama de Voronoi, explicado na seção 4.3, na página 60.

Nem todo conjunto de pontos suporta uma triangulação de Pitteway. Quando existe, a triangulação de Pitteway consiste na união do fecho convexo com o grafo de Gabriel [57]. Logo, o grafo de Gabriel é um subgrafo da triangulação de Delaunay. O grafo de Gabriel expressa a noção de proximidade entre pontos. Forma-se o grafo de Gabriel ao formar arestas entre pontos  $p, q \in V$ , em que o círculo diametral da aresta  $\overline{pq}$  é vazio (veja a definição 4.2, na página 52). Grafos de Gabriel são estendidos naturalmente para três dimensões. Nesse caso, a condição do círculo diametral vazio é substituída pela condição da circunfera vazia do triângulo.

Na figura 4.21, observa-se um exemplo da triangulação de Pitteway. Na figura 4.19, observa-se um exemplo em que não ocorre a triangulação de Pitteway.



**Figura 4.19:** Exemplo de triangulação de Delaunay que não é uma triangulação de Pitteway.

### 4.3 Diagrama de Voronoi

O diagrama de Voronoi é um tipo específico de decomposição (em regiões) de um espaço. Na sua forma mais típica, essa decomposição do espaço é determinada pelas distâncias para um conjunto de pontos geradores no espaço. Em mais detalhes, o diagrama de Voronoi no plano euclidiano é o conjunto não sobreposto de regiões  $\mathfrak{R}_i$ , considerando-se que o restante do espaço é delimitado por arestas infinitas. Considere a seguinte definição de região de diagrama de Voronoi no espaço euclidiano, que é o caso mais simples e comum.

**Definição 4.3.** (*região de diagrama de Voronoi*) Seja  $V \subset \mathbb{R}^d$  um conjunto de  $n$  pontos geradores  $p_i$  no espaço euclidiano, para  $1 \leq i \leq n$ . Uma região de Voronoi  $\mathfrak{R}_i$ , associada ao ponto gerador  $p_i$ , é o conjunto de todos os pontos em  $\mathbb{R}^d$  que estão pelo menos tão próximos a  $p_i$  do que a qualquer outro ponto gerador em  $V$ , ou seja, para dimensões  $d$  iguais a 2 ou 3, cada região de Voronoi é um conjunto de pontos  $\mathfrak{R}_i = \{p \in \mathbb{R}^d : (\exists p_i \in V)(\forall p_j \in V) \|p - p_i\| \leq \|p - p_j\|, \text{ com } i \neq j \wedge 1 \leq i, j \leq n\}$ .

Para um dado conjunto de *pontos geradores*  $V$ , um diagrama de Voronoi é gerado de forma que cada ponto gerador possui uma região mais próxima associada a esse ponto gerador do que a qualquer outro ponto gerador. Em mais detalhes, ao gerar um diagrama de Voronoi, o espaço é subdividido em regiões  $\mathfrak{R}_i$ , denominadas *regiões de Voronoi*, de tal forma que a região  $\mathfrak{R}_i$  é o espaço mais próximo do ponto gerador  $p_i \in V$  do que a qualquer outro ponto em  $V$ .

Em um domínio finito (fechado), as regiões dessa tesselação decompõem o domínio em um conjunto de polítopos convexos não sobrepostos em torno de pontos geradores [150], que compreendem todo o domínio. Em um domínio finito, a interface entre duas regiões de Voronoi, isto é, o conjunto de pontos que delimitam duas regiões de Voronoi adjacentes, é uma aresta e um polígono no espaço euclidiano bidimensional e tridimensional, respectivamente.

Conceitos semelhantes aos apresentados por Voronoi também foram abordados por Descartes [45] ao descrever que o sistema solar consiste de vórtices e ao mostrar ilustrações em que decompõe o espaço em regiões convexas em torno de estrelas. É possível que o primeiro a descrever formalmente essas ideias tenha sido Dirichlet [46], na investigação de formas quadráticas positivas. Dirichlet [46] propôs um método em que um determinado domínio poderia ser decomposto, sistematicamente, em um conjunto de regiões convexas. Por isso, essa estrutura geométrica também é conhecida como *tesselação de Dirichlet*. No contexto de geografia e meteorologia, o diagrama de Voronoi é conhecido como polígonos de Thiessen [141].

O diagrama de Voronoi pode ser lembrado em várias situações na natureza. Por exemplo, na figura 4.20, mostra-se parte da casca de um marolo (*Annona crassiflora*).



**Figura 4.20:** Parte da casca de marolo (*Annona crassiflora*) e sua similaridade com o diagrama de Voronoi.

Um dos primeiros algoritmos propostos para a geração do diagrama de Voronoi foi o algoritmo por divisão e conquista de Shamos e Hoey [130]. Esse algoritmo é  $O(n \lg n)$  no pior caso, para  $n$  pontos no plano. Outro algoritmo por divisão e conquista para a geração do diagrama de Voronoi foi proposto por Bentley, Weide e Yao [13, 14]. Pode ser que esse tenha sido o primeiro algoritmo com comportamento linear para a geração do diagrama de Voronoi.

Brown [22, 23] mostrou como computar um diagrama de Voronoi  $d$ -dimensional ao transformar pontos para o espaço  $d + 1$ , computar o fecho donvexo dos pontos transformados e, então, transformar o fecho convexo do espaço  $d$ -dimensional. Em termos simples, um *fecho convexo* de um conjunto de pontos no espaço euclidiano bidimensional é o menor conjunto convexo que contém esse conjunto de pontos (veja a definição 1.10, na página 13). A versão do método de Brown para dimensões maiores foi apresentada por Seidel [129].

Também há correspondência entre o diagrama de Voronoi e o eixo medial. Veja detalhes sobre esse assunto em Nogueira [100], por exemplo.

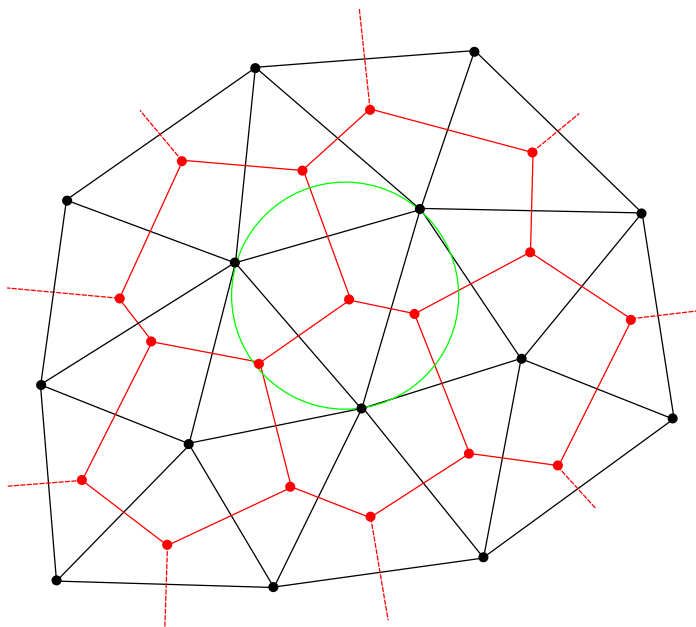
A seguir, são abordadas a dualidade entre o diagrama de Voronoi e a tesselação de Delaunay na subseção 4.3.1. O diagrama de potência, uma generalização do diagrama de Voronoi, é abordado na subseção 4.3.2

### 4.3.1 Dualidade entre diagrama de Voronoi e tesselação de Delaunay

Os pontos geradores do diagrama de Voronoi são os vértices da tesselação de Delaunay. Em um domínio finito, se todos os pares de pontos geradores que têm alguma interface (aresta no domínio bidimensional e polígono no domínio tridimensional) de regiões em comum são unidos por arestas, um conjunto de triângulos ou de tetraedros é formado no domínio bidimensional ou tridimensional, respectivamente. Com isso, o resultado é uma tesselação de Delaunay do fecho convexo do conjunto de pontos geradores. Em outras palavras, é obtida uma tesselação de Delaunay dos pontos geradores do diagrama de Voronoi se esses pontos geradores, tendo uma interface de região comum, forem ligados por arestas. Ainda, os vértices do diagrama de Voronoi são circuncentros de triângulos adjacentes (ou centros de esferas que circundam tetraedros em um domínio tridimensional) da tesselação de Delaunay que contêm intersecção com as regiões de Voronoi. Isso significa que, em particular em um domínio bidimensional, ao se considerar que o triângulo  $\triangle abc$  pertence a uma triangulação de Delaunay, o centro do circuncírculo do triângulo  $\triangle abc$  corresponde a um vértice do diagrama de Voronoi. Veja um exemplo de triangulação de Delaunay na figura 4.21.

Por causa dessa dualidade entre o diagrama de Voronoi e a tesselação de Delaunay, o diagrama de Voronoi é definido como uma estrutura dual da tesselação de Delaunay. Dessa maneira, é possível a transformação da tesselação de Delaunay em diagrama de Voronoi e vice-versa. Dado o diagrama de Voronoi de um conjunto de  $n$  pontos, a triangulação de Delaunay pode ser obtida em tempo  $O(n)$ . O mesmo ocorre para a transformação da triangulação de Delaunay em diagrama de Voronoi.

Como descrito, em um domínio bidimensional, entre polígonos de Voronoi e triângulos correspondentes da triangulação de Delaunay que têm intersecção, arestas da triangulação de Delaunay (arestas entre pontos geradores) e arestas do diagrama de Voronoi formam ângulos retos (veja a figura 4.21). Logo, uma triangulação de Delaunay só existe se dois pontos a serem conectados forem pontos terminais de um segmento de reta perpendicular a uma aresta no diagrama de Voronoi. Uma consequência disso é que a tesselação resultante é adequada para ser utilizada na



**Figura 4.21:** Partição do diagrama de Voronoi em vermelho e a triangulação de Delaunay corresponde em preto. Os pontos vermelhos são os circuncentros dos circuncírculos dos triângulos de Delaunay, e também são vértices das regiões de Voronoi. Os pontos pretos, que são os vértices dos triângulos de Delaunay, também são os pontos geradores do diagrama de Voronoi. (Figura extraída de [107]).

discretização de equações diferenciais parciais por volumes finitos. Note que utiliza-se o termo tesselação como sinônimo de malha computacional. Esses conceitos da tesselação de Delaunay e do diagrama de Voronoi são particularmente importantes para a aplicação do método dos volumes finitos com o diagrama de Voronoi por causa do teorema da divergência. Considere uma aresta  $\overline{ab}$  da triangulação de Delaunay, incidente a pontos de avaliação  $a$  e  $b$  de dois volumes de controle (polígonos de Voronoi em um domínio bidimensional)  $\mathfrak{R}_a$  e  $\mathfrak{R}_b$ , respectivamente. O ângulo entre a aresta  $\overline{ab}$  e a interface (aresta do diagrama de Voronoi em um domínio bidimensional) entre  $\mathfrak{R}_a$  e  $\mathfrak{R}_b$  é reto. Com isso, cálculos de cossenos não necessitam ser realizados, e esses cálculos ocasionariam acréscimo inerente no erro numérico. Na prática, gera-se a triangulação de Delaunay e os pontos de avaliação da aplicação do método dos volumes finitos são os próprios vértices da tesselação de Delaunay, ou seja, os pontos geradores do diagrama de Voronoi correspondente.

O grau dos vértices geradores da triangulação de Delaunay pode ser variável. Já o grau do vértice gerador do diagrama de Voronoi é fixo, exceto em casos em que há uma triangulação de Delaunay com quatro ou mais pontos cocirculares.

### 4.3.2 Diagrama de potência

O diagrama de Voronoi ponderado é uma generalização do diagrama de Voronoi e é dual à triangulação de Delaunay ponderada. Como é calculada a distância no diagrama de Voronoi ponderado é o que o difere do diagrama de Voronoi, em que é utilizada a distância euclidiana. Um ponto gerador  $c$  é substituído por um

valor  $w(c)$ . Isso significa que um ponto gerador  $c$  possui um peso no diagrama de Voronoi ponderado. Com isso, em vez de se avaliar a distância de um ponto  $p \in \mathbb{R}^d$  para o ponto gerador  $c$ , como é realizado no diagrama de Voronoi, a distância do ponto  $p$  é avaliada em função da distância entre  $p$  e  $c$  e também de  $w(c)$ , no diagrama de Voronoi ponderado [4]. Aurenhammer [4] também explica que a diferença da distância entre  $p$  e  $c$  por  $w(c)$  leva a diagramas cujas instâncias planares podem ser interpretadas como diagramas de Voronoi para circunferências, em que  $w(c) \geq 0$  é o circunraio da circunferência com circuncentro  $c$  e a distância de  $p$  para a circunferência é a *potência de um ponto*, definida a seguir.

**Definição 4.4.** (*potência de ponto em relação à circunferência*) A *potência de um ponto  $p$ , no plano euclidiano, em relação à circunferência, com circuncentro  $c$  e circunraio  $R$ , é  $\|p - c\|^2 - R^2$ , em que  $\|p - c\|$  é a distância euclidiana do ponto  $p$  para a circunferência.*

Dessa forma, o diagrama de Voronoi ponderado é chamado de diagrama de potência. Como exemplo de aplicação do diagrama de potência, já em 1979, Boots [17] aplicou esse diagrama em Geografia.

Uma forma simples de se entender o diagrama de potência é o seguinte. Considere um conjunto de circunferências. Um polígono do diagrama de potência é formado por todos os pontos que estão mais próximos de uma circunferência do que de todas as outras circunferências desse conjunto. Para saber a distância de um ponto  $p$  para uma circunferência, utiliza-se a *potência do ponto* em relação à circunferência, definida anteriormente.

Se a potência do ponto em relação a uma circunferência é positiva, negativa ou zero significa que o ponto é externo, interno ou pertence à circunferência, respectivamente. Considere a seguinte definição de região do diagrama de potência.

**Definição 4.5.** (*região do diagrama de potência no plano euclidiano*) Seja um conjunto de circunferências  $C = \{C_1, C_2, \dots, C_n\}$ , em que uma circunferência  $C_i$  tem circuncentro  $c_i$  e circunraio  $R_i$ . Cada região do diagrama de potência é um conjunto de pontos  $P_i = \{p \in \mathbb{R}^2 : (\exists C_i \in C)(\forall C_j \in C) \|p - c_i\|^2 - R_i^2 \leq \|p - c_j\|^2 - R_j^2, \text{ com } i \neq j \wedge 1 \leq i, j \leq n\}$ .

O diagrama de potência (ou de Laguerre) é o conjunto não sobreposto das regiões  $P_i$ , com  $1 \leq i \leq n$ ; considerando-se que o restante do espaço é delimitado por arestas infinitas, no plano euclidiano. A potência de um vértice do diagrama de potência no plano euclidiano é a mesma para três circunferências. Em um domínio bidimensional finito, uma aresta (um polígono em um espaço tridimensional finito) do diagrama de potência é o conjunto de pontos que delimitam duas regiões adjacentes do diagrama de potência.

Como descrito, o diagrama de Voronoi é um caso especial do diagrama de potência: no plano euclidiano, se todas as circunferências são congruentes, o diagrama de potência é o diagrama de Voronoi, em que os circuncentros são os pontos geradores. Essas características do diagrama de potência podem ser estendidas a dimensões maiores. Esta subseção fornece apenas uma noção do diagrama de potência. Uma ótima fonte para o diagrama de potência é o artigo de Aurenhammer [4], em que o autor descreve as relações do diagrama com fecho convexo e apresenta algoritmos eficientes para computar o diagrama, entre outros resultados.

## 4.4 Notas bibliográficas

Uma fonte excelente, em diversos aspectos importantes de geração de tesselações de Delaunay, não abrangidos neste texto, é o livro *Delaunay Mesh Generation* de Cheng, Dey e Shewchuk [32]. Os autores descrevem a tesselação de Delaunay em detalhes, bem como fornecem demonstrações dos conceitos abordados, detalhes de algoritmos, custos computacionais, bem como detalhes da tesselação de Delaunay em três dimensões.

## 4.5 Exercícios

1. Explique a condição de circuncírculos vazios da triangulação de Delaunay.
2. Toda triangulação é uma triangulação de Delaunay? Explique.
3. O que é uma aresta local de Delaunay?
4. Se uma aresta não é localmente de Delaunay, o que pode ser realizado para torná-la uma aresta local de Delaunay e qual o algoritmo que pode ser utilizado nesse caso?
5. Sempre existe uma triangulação de Delaunay para  $n$  pontos? Explique.
6. O que caracteriza uma triangulação de Delaunay restrita?
7. Necessariamente, uma triangulação de Delaunay restrita é uma triangulação de Delaunay? Por quê?
8. Altere o algoritmo de Lawson mostrado no algoritmo 1, na página 50, de forma que seja utilizado para a geração da triangulação de Delaunay e não somente para a manutenção da malha.
9. Explique quais as aplicações de uma curva de preenchimento de espaço no contexto de malhas computacionais.
10. Ao identificar um triângulo de má qualidade, quais os lugares geométricos possíveis em que um novo vértice será inserido pelo algoritmo de Ruppert?
11. Explique por que o diagrama de Voronoi é uma malha dual da triangulação de Delaunay.
12. Forneça uma noção intuitiva sobre o teorema da divergência. Em seguida, explique as implicações de aplicar o teorema da divergência no domínio de Voronoi em um domínio fechado.
13. Explique o que são circunferências congruentes.
14. Por que o diagrama de Voronoi é um caso especial do diagrama de potência?



## Capítulo 5

# Representações de malhas triangulares

### 5.1 Introdução

O desenvolvimento de métodos para controlar e ajustar níveis de detalhes de um conjunto de dados é uma área de pesquisa ativa. Isso ocorre, principalmente, por causa da crescente necessidade de estudo e visualização de detalhes de conjuntos de dados tridimensionais complexos, tais como em formas geométricas, modelos geográficos ou campos escalares volumétricos [41].

De Floriani, Kobbelt e Puppo [41] também explicam que muitas aplicações exigem operações de extração de dados em tempo real ou, pelo menos, *on-line*, para estruturas que, geralmente, possuem geometrias bastante complicadas. Exigências de desempenho e escalabilidade impõem vários desafios no projeto de sistemas dessa natureza, e o projeto de estruturas de dados adequadas tem um papel importante nesse contexto. Geralmente, uma melhor qualidade da solução do problema é diretamente proporcional ao custo computacional e, levando-se em conta a escalabilidade, necessita-se encontrar um equilíbrio para que o modelo seja genérico, flexível, com boa qualidade de aproximação à solução do problema e com baixo custo computacional e de armazenamento. De forma geral, a escolha da estrutura de dados depende da aplicação, do desempenho esperado, tamanho dos dados e das operações a serem realizadas. Em particular, no plano euclidiano, é mais fácil lidar com o armazenamento de triângulos do que com polígonos mais complexos. Neste capítulo, são abordadas estruturas de dados para a representação de malhas triangulares.

Formas básicas de armazenamento de grafos são mostradas na seção 5.2. A estrutura de dados *winged-edge* [11] é descrita na seção 5.3. A estrutura de dados DCEL [96] é abordada na seção 5.4. Também, é mostrado um grafo especialmente projetado para a representação de refinamento adaptativo de malhas triangulares [69] na seção 5.5. Finalmente, custos de armazenamento de 10 estruturas de dados para representação de malhas triangulares são mostrados na seção 5.6.

### 5.2 Formas básicas de armazenamento

Como descrito na seção 1.2, uma malha pode ser compreendida como um grafo não direcionado. Ainda, grafos e matrizes são estruturas equivalentes. Com isso,

formas básicas de armazenamento de matrizes são abordadas nesta seção. Matrizes de adjacências e listas de adjacências são comentadas na subseção 5.2.1. Na subseção 5.2.2, são abordadas formas de armazenamento baseadas no formato CSR (*Compressed Sparse Row*). Essas estruturas de dados podem ser utilizadas para representar quaisquer tipos de malhas.

### 5.2.1 Matriz de adjacências e listas de adjacências

Uma forma básica de se representar um grafo  $G = (V, E)$  é por uma matriz de adjacências. A vantagem dessa forma de armazenamento é o acesso à estrutura de dados em  $\Theta(1)$ . Todavia, o custo de armazenamento dessa estrutura de dados é  $O(|V|^2)$  e isso pode ser um limitador em matrizes de grande porte.

Uma alternativa natural para matrizes de adjacências são as listas de adjacências, já que, tipicamente, as matrizes são esparsas em simulações numéricas. Note que um vértice de um grafo em memória pode representar um polítopo da malha e, com isso, adjacências entre vértices do grafo seriam as adjacências entre polítopos. Essa é uma forma natural de armazenamento em discretizações por volumes finitos, por exemplo. Com listas de adjacências [77], pode-se armazenar uma lista encadeada simples dos vértices do grafo. Na representação por listas de adjacências, para cada vértice do grafo, armazena-se a lista de adjacências do vértice. Dessa forma, em um grafo não direcionado, cada aresta é armazenada duas vezes, ou seja, uma vez em cada vértice da aresta. Deve-se armazenar o endereço do vértice inicial e, para se determinar o final da lista encadeada, há um endereço nulo no último vértice da lista encadeada. Logo, a lista de adjacências primária tem  $|V| + 1$  endereços armazenados. Portanto, essa forma de armazenamento tem custo  $2|A| + |V| + 1 = O(|V| + |A|)$ . Vantagens dessa estrutura de dados são a sua simplicidade de implementação e alterações topológicas na malha são simples de serem efetuadas: apenas endereços de memória são alterados para se atualizar a estrutura de dados. Em geral, o número de adjacências de um vértice é pequeno em uma malha; logo, as adjacências são rapidamente determinadas.

O armazenamento de uma malha por listas de adjacências é uma forma adequada se determinados métodos numéricos são utilizados. Por exemplo, pode-se resolver um sistema de equações lineares com matriz de grande porte por um método iterativo, como o método dos gradientes conjugados [75, 83]. O método dos gradientes conjugados é raramente utilizado sem um condicionador. O condicionador de Jacobi é adequado se listas de adjacências são utilizadas. No entanto, os condicionadores baseados em fatoração incompleta de Cholesky e fatoração incompleta LU são inviáveis para serem utilizados com listas de adjacências. Por isso, formas de armazenamento baseadas em um esquema matricial são geralmente utilizadas com esses condicionadores.

### 5.2.2 Formato CSR e variações

A seguir, são descritos formatos básicos de se armazenar matrizes, os formatos por coordenadas, CSR, *skyline* e CSR-SSS.

#### Formato por coordenadas

Uma forma básica de armazenamento que alia vantagens da matriz de adjacências e das listas de adjacências é a seguinte. Considere a matriz

$$A = \begin{pmatrix} a_{11} & 0 & a_{13} & 0 \\ 0 & a_{22} & 0 & a_{24} \\ a_{31} & 0 & a_{33} & 0 \\ 0 & a_{42} & 0 & a_{44} \end{pmatrix}.$$

Os coeficientes não nulos da matriz  $A$  podem ser armazenados em

$$A_n = [a_{11}, a_{31}, a_{22}, a_{42}, a_{13}, a_{33}, a_{24}, a_{44}]$$

e os índices das linhas e das colunas em

$$A_i = [1, 3, 2, 4, 1, 3, 2, 4]$$

e

$$A_j = [1, 1, 2, 2, 3, 3, 4, 4],$$

respectivamente. O custo dessa forma de armazenamento é de  $3N_z$ , em que  $N_z$  é o número de coeficientes não nulos da matriz  $A$ . Note que pode-se ter qualquer disposição dos coeficientes em  $A_n$ , alterando-se  $A_i$  e  $A_j$  de acordo com a disposição dos coeficientes não nulos em  $A_n$ .

### Formato CSR

No formato CSR ou CSC (*Compressed Sparse Row/Column*, também chamado de *Compressed Row/Column Storage*), os coeficientes não nulos da matriz  $A$  são dispostos linha por linha ou coluna por coluna, respectivamente. Dessa forma, no formato CSR, a matriz  $A$  é representada por  $A_n = [a_{11}, a_{13}, a_{22}, a_{24}, a_{31}, a_{33}, a_{42}, a_{44}]$ ,  $A_c = [1, 3, 2, 4, 1, 3, 2, 4]$  e  $A_l = [1, 3, 5, 7, 9]$ , em que  $A_c$  contém os índices das colunas dos coeficientes não nulos da matriz  $A$  e  $A_l$  contém os índices de início de cada linha em  $A_c$  e a entrada  $A_l[n + 1]$  contém  $A_l[1] + N_z$ . O tamanho de  $A_n$  é  $N_z$ , o tamanho de  $A_c$  é  $N_z$  e o tamanho de  $A_l$  é  $n + 1$ , em que  $n$  é o número de linhas da matriz  $A$ . Note que os coeficientes de  $A_n$  podem ser substituídos por endereços de memória, por exemplo, dos polítopos da malha armazenados como objetos em uma linguagem orientada a objetos.

### Formato skyline

Há diversas variações dos formatos CSR e CSC. Uma formato bastante popular na comunidade de elementos finitos é o formato SSS, chamado de esquema de armazenamento compacto (*Compact Storage Scheme*) por Jennings [79] e de *Skyline Storage Scheme* por Felippa [51] ou *Symmetric Sparse Skyline* [61], também conhecido como armazenamento por envelope, *profile* ou largura de banda variável. Para descrições dessas características de matrizes, veja o livro de Gonzaga de Oliveira e Chagas [66].

Considere que a matriz  $A$  é simétrica. No formato SSS [51], armazenase a diagonal principal e o envelope da matriz  $A$ . O vetor *skyline* é  $A_n = [a_{11}, a_{22}, a_{13}, 0, a_{33}, a_{24}, 0, a_{44}]$  e a  $i$ -ésima entrada do vetor de índices indica o coeficiente da diagonal principal da  $i$ -ésima linha:  $A_i = [1, 2, 5, 8]$ . Portanto, o tamanho de  $A_n$  é o *profile* da matriz  $A$  adicionado do número  $n$  de linhas da matriz  $A$ . O tamanho de  $A_i$  é o número de linhas da matriz  $A$ . Na linguagem Fortran, é conveniente incluir a primeira entrada de  $A_i$  como zero, obtendo-se  $A_i$  com  $n + 1$  entradas.

A grande vantagem do formato *skyline* é que *fill-in* recai somente no envelope da matriz  $A$  durante uma decomposição como a de Cholesky. Para armazenar uma matriz assimétrica no formato *skyline*, são armazenados os envelopes das matrizes triangular inferior e superior da matriz  $A$ .

Como o custo desse armazenamento é dependente do *profile* da matriz  $A$ , heurísticas para reduções de *profile* e da largura de banda da matriz  $A$  são largamente empregadas nesse contexto. O livro de Gonzaga de Oliveira e Chagas [66] é inteiramente dedicado a esses assuntos.

### Formato CSR-SSS

Para matrizes simétricas, em vez de utilizar o formato CSR, uma escolha natural é armazenar somente as linhas da matriz triangular inferior. Ainda, um vetor grande pode ser impeditivo, porque há a necessidade de alocação de espaço contínuo da memória, em vez de espaços distribuídos, como ocorre nas listas de adjacências. Portanto, é conveniente diminuir o vetor de coeficientes do formato CSR ao armazenar os coeficientes da diagonal principal em um vetor separado. Com isso, o vetor de índices das colunas também é diminuído porque não é mais necessário armazenar as colunas dos coeficientes da diagonal principal. Esse formato é conhecido como CSR-SSS: os coeficientes da diagonal principal são armazenados separadamente e os coeficientes não nulos da matriz triangular inferior são armazenados linha a linha, como no formato CSR. Se a matriz  $A$  é simétrica, então, tem-se  $A_d = [a_{11}, a_{22}, a_{33}, a_{44}]$ ,  $A_n = [a_{31}, a_{42}]$ ,  $A_c = [1, 2]$  e  $A_l = [0, 0, 1, 2, 2]$ . O tamanho de  $A_d$  é o número  $n$  de linhas da matriz  $A$ , o tamanho de  $A_l$  é  $n + 1$  e ambos,  $A_n$  e  $A_c$  têm o número de coeficientes não nulos da matriz triangular inferior da matriz  $A$ , que é o número de arestas do grafo. Como comparação, listas de adjacências têm custo  $2|A| + |V| + 1$  e o formato CSR-SSS tem custo  $2|A| + 2|V| + 1$ .

## 5.3 Estrutura de dados *winged-edge*

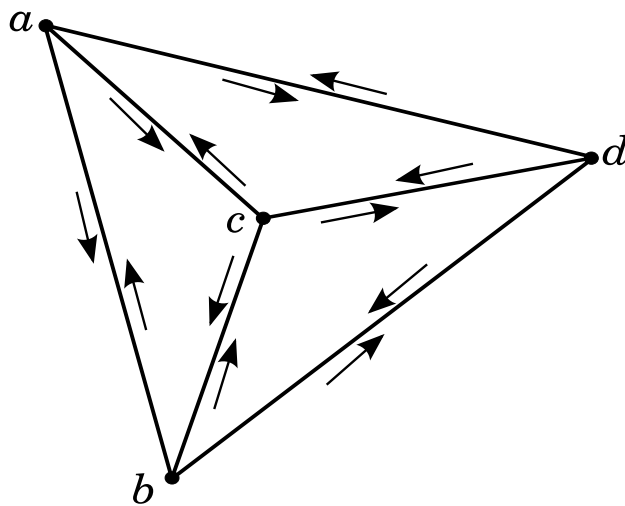
A estrutura de dados *winged-edge* foi proposta originalmente para representar poliedros com faces poligonais [11]. A aresta é a entidade central na estrutura de dados *winged-edge*. As quatro arestas adjacentes a uma aresta  $\overline{ab}$  são armazenadas aos pares, isto é, uma predecessora e uma sucessora, em duas listas circulares, e as duas faces incidentes à aresta  $\overline{ab}$ , formam as “asas” da aresta  $\overline{ab}$ , então, cada aresta é “alada”.

Na estrutura de dados *winged-edge*, as adjacências imediatas para entidades da malha são codificadas. A estrutura de dados codifica explicitamente todas as entidades que formam a malha e codifica a maioria da topologia do politopo. Para cada aresta  $\overline{ab}$  da malha, a estrutura de dados mantém os índices dos dois vértices  $a$  e  $b$ , dos dois polígonos a que a aresta  $\overline{ab}$  pertence, das quatro arestas que são adjacentes à aresta  $\overline{ab}$  e também pertencem aos dois polígonos a que a aresta  $\overline{ab}$  pertence.

Vértices e polígonos também são codificados. Em uma forma simplificada, pode-se armazenar apenas a aresta inicial de cada vértice, além de suas coordenadas. Em uma forma mais completa, pode-se armazenar, para cada vértice, um ponteiro para cada aresta incidente ao vértice. Para cada polígono, as arestas do polígono são armazenadas em uma lista circular, ao se estabelecer uma aresta inicial. Isso significa que cada aresta direcionada tem uma aresta direcionada predecessora e uma aresta direcionada sucessora.

Ligações de arestas mantêm uma forma de percorrer a estrutura de dados de forma eficiente. A estrutura de dados é percorrida pelas arestas de um polígono ou pelas arestas a que um vértice tem ligação, em sentido horário ou anti-horário. Ao supor que o tamanho de um índice é uma unidade e sem levar em consideração informações geométricas e atributos dependentes da aplicação, o custo total de armazenamento dessa estrutura de dados é, aproximadamente,  $27n$ , em que  $n$  é o número de vértices na malha [41].

Uma representação do tetraedro mostrado na figura 5.1 pela estrutura de dados *winged-edge* é mostrada a seguir: triângulos, vértices e arestas são representados nas Tabelas 5.1, 5.2 e 5.3, respectivamente. A representação está em sentido horário, a partir da parte externa do tetraedro.



**Figura 5.1:** Vista superior de um tetraedro para ser representado pela estrutura de dados *winged-edge*. Cada um dos quatro triângulos do tetraedro possui uma lista circular de suas arestas. A lista circular do  $\triangle abd$  é mostrada na parte externa da imagem.

$\triangle$	$acb$	$adc$	$bcd$	$bda$
aresta	$\overline{ac}$	$\overline{ad}$	$\overline{bc}$	$\overline{bd}$

**Tabela 5.1:** Uma representação dos triângulos do tetraedro mostrado na figura 5.1: armazena-se uma aresta que o triângulo incide na entrada de cada triângulo.

Vértice	aresta
$a$	$\overline{ac}$
$b$	$\overline{bd}$
$c$	$\overline{cb}$
$d$	$\overline{dc}$

**Tabela 5.2:** Uma representação dos vértices do tetraedro mostrado na figura 5.1: armazena-se uma aresta incidente na entrada de cada vértice. As coordenadas dos vértices também são armazenadas.

### 5.4 Estrutura de dados DCEL

A estrutura de dados *Doubly-Connected Edge List* (DCEL) [96] é uma versão simplificada da estrutura de dados *winged-edge*, descrita na seção 5.3. Apesar do nome, a estrutura de dados DCEL é mais do que uma simples lista duplamente encadeada de arestas. Na estrutura de dados DCEL, para cada aresta, índices de duas arestas adjacentes são mantidos em sentido anti-horário em relação ao polígono incidente à esquerda da aresta. Também são armazenados os vértices de origem (de saída) e destino (de chegada) da aresta (direcionada). Além disso, com a estrutura de dados DCEL, os elementos a que um vértice tem ligações podem ser percorridos somente em sentido anti-horário e a fronteira de polígonos em sentido horário. A estrutura de dados DCEL tem custo de armazenamento de  $21n$ , em que  $n$  é o número de vértices na malha [41].

Aresta <i>winged</i>	Orientação à					
	direita			esquerda		
	$\triangle$ incidente	aresta		$\triangle$ incidente	aresta	
		pred.	sucessora		pred.	sucessora
$\overline{ba}$	$bac$	$\overline{cb}$	$\overline{ac}$	$abd$	$\overline{da}$	$\overline{bd}$
$\overline{ac}$	$acb$	$\overline{ba}$	$\overline{cb}$	$cad$	$\overline{dc}$	$\overline{ad}$
$\overline{cb}$	$cba$	$\overline{ac}$	$\overline{ba}$	$bcd$	$\overline{db}$	$\overline{cd}$
$\overline{bd}$	$bda$	$\overline{ab}$	$\overline{da}$	$dbc$	$\overline{cd}$	$\overline{bc}$
$\overline{dc}$	$dca$	$\overline{ad}$	$\overline{ca}$	$cdb$	$\overline{bc}$	$\overline{db}$
$\overline{da}$	$dab$	$\overline{bd}$	$\overline{ab}$	$adc$	$\overline{ca}$	$\overline{dc}$

**Tabela 5.3:** Representação das arestas do tetraedro mostrado na figura 5.1, em sentido horário, a partir da parte externa do tetraedro. Claramente, a aresta  $\overline{ba}$  tem vértices de saída  $b$  e de chegada  $a$ . Ponteiros para os vértices de cada aresta também são armazenados.

Como na estrutura de dados *winged-edge*, na estrutura de dados DCEL, cada aresta é representada por duas arestas direcionadas. Cada aresta tem um ponteiro para a sua aresta direcionada “gêmea”, isto é, com direção oposta. Cada aresta direcionada é associada com dois polígonos, um a sua esquerda e outro a sua direita, que são incidentes a essa aresta.

Informações armazenadas em relação a vértices são as mesmas nas estruturas de dados *winged-edge* e DCEL. Isso significa que, em relação ao vértice  $a$ , é armazenada somente índice de aresta em que o vértice  $a$  é o vértice de origem da aresta. Em relação a polígonos, informações armazenadas em relação a vértices podem ser as mesmas nas estruturas de dados *winged-edge* e DCEL, isto é, na estrutura de dados DCEL, armazena-se um ponteiro para uma só aresta em que esse polígono é incidente.

## 5.5 Grafo para representar refinamento adaptativo

Um grafo especialmente projetado para representar o refinamento adaptativo de malhas triangulares (bidimensionais) não conformes para discretizações de equações diferenciais parciais por volumes finitos foi mostrado por Gonzaga de Oliveira, Kischinhevsky e Tavares [69]. Esse grafo é uma extensão para malhas triangulares do grafo proposto por Burgarelli, Kischinhevsky e Biezuner [24] para representação de refinamento adaptativo de malhas quadrangulares.

São fornecidas as relações entre triângulos da malha nesse grafo para representação do refinamento adaptativo de malhas triangulares. Para evitar confusão entre vértice do grafo e vértice da triangulação, utiliza-se o termo *nodo* do grafo nesta seção. Os triângulos da malha são representados por nodos do grafo. Dois nodos são diretamente conectados se representam dois triângulos com interface em comum. Ao se refinar um triângulo, o triângulo original é substituído por um novo subgrafo composto de:

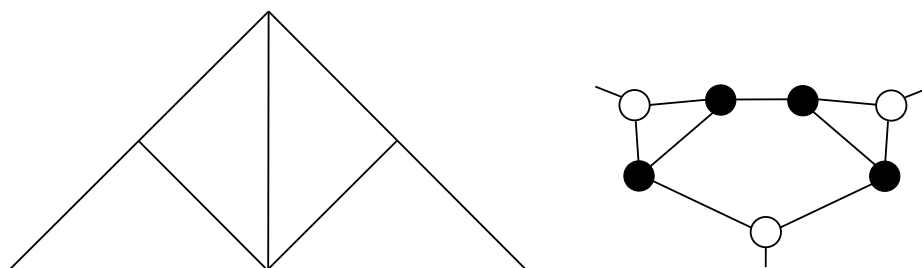
- quatro novos *nodos de triângulo* que representam os quatro novos triângulos;
- três *nodos de transição* entre nodos de triângulos do grafo que, por sua vez, representam triângulos com níveis de refinamentos diferentes;

- as ligações entre esses nodos do grafo.

Em um refinamento local de um triângulo, o nodo que representa o triângulo original é excluído (em memória), bem como possíveis nodos de transição desse nodo de triângulo também são excluídos. Logo, somente os nodos criados na operação de refinamento são armazenados. Os nodos que representam a triangulação inicial são estabelecidos com nível zero de refinamento e, com o refinamento local, os níveis de refinamento dos nodos de triângulos gerados recebem o valor do nível do nodo de triângulo refinado incrementado de uma unidade.

Nodos de transição conectam triângulos de níveis diferentes no refinamento adaptativo da triangulação. Os nodos de transição indicam o nível de refinamento do triângulo em relação a seus triângulos adjacentes. Portanto, nodos de transição existem se há triângulos adjacentes com níveis de refinamento diferentes no refinamento adaptativo da triangulação. Com isso, é possível ter um número menor de nodos no grafo do que com a propagação do refinamento adaptativo da triangulação. Consequentemente, o custo de armazenamento da triangulação é pequeno. Claramente, a malha resultante após o refinamento adaptativo será não conforme. Por isso, como descrito, essa estrutura de dados foi projetada para discretizações de equações diferenciais parciais por volumes finitos. Uma adaptação seria necessária para essa estrutura de dados ser utilizada em discretizações de equações diferenciais parciais por elementos finitos, de forma similar à realizada por Brandão, Gonzaga de Oliveira e Kischinhevsky [21].

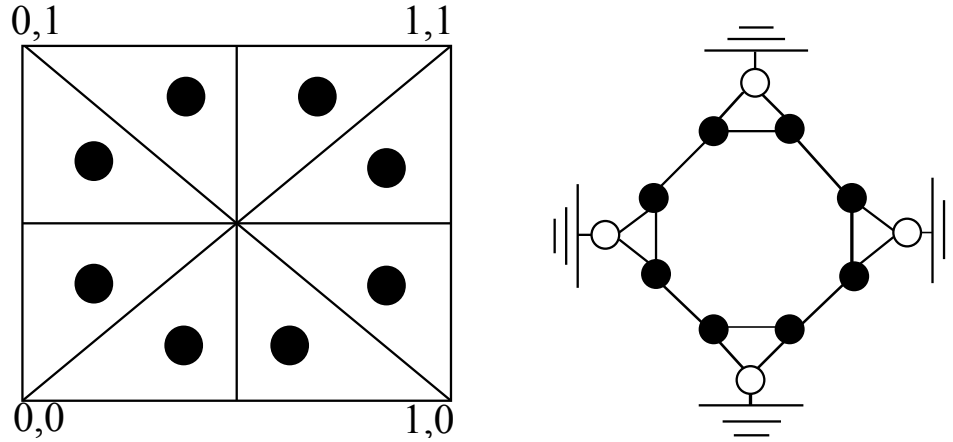
O processo de refinamento é exemplificado na figura 5.2. Como descrito, um nodo de triângulo do grafo representa um triângulo da malha. Se o triângulo da malha é refinado, então, o nodo de triângulo que representa o triângulo é substituído por um subgrafo como exemplificado na figura 5.2. No subgrafo à direita na figura 5.2, ligações entre nodos do grafo são representadas por linhas e um grafo direcionado é representado como um grafo não direcionado em busca de clareza na apresentação. Isso significa que uma linha entre nodos do grafo na figura 5.2 representa que os nodos apontam-se aos pares. Os nodos de triângulos apontam para nodos de transição aos pares. Cada nodo de transição também aponta para dois nodos de triângulos no subgrafo. Finalmente, cada nodo do grafo tem três ponteiros. Um ponteiro não utilizado na fronteira do domínio recebe o valor nulo.



**Figura 5.2:** Com o refinamento adaptativo de um triângulo, um nodo de triângulo do grafo que representa o triângulo da malha é substituído por um subgrafo (à direita). O oposto é realizado no processo de simplificação local. (Figuras extraídas de [69].)

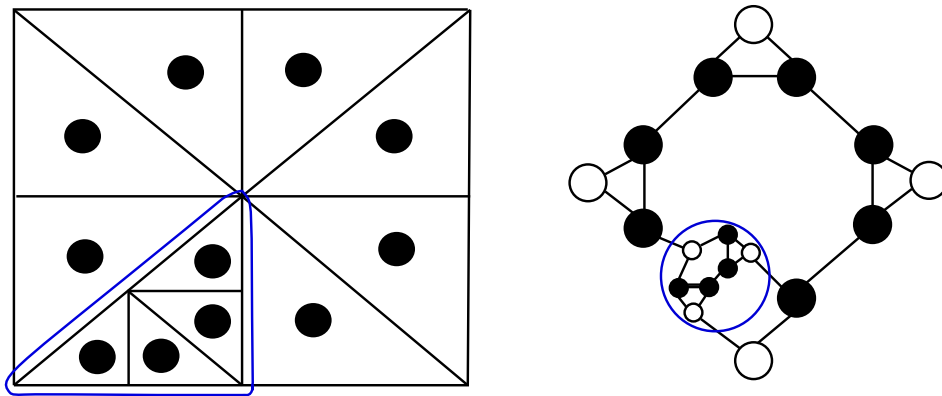
Um exemplo de triangulação inicial e o grafo correspondente são mostrados na figura 5.3. Nessa figura, os *baricentros* (veja a definição 2.1, na página 23) dos

triângulos são representados por círculos pretos e círculos brancos representam as ligações para a fronteira do domínio fechado.



**Figura 5.3:** O quadrado unitário como um exemplo de domínio, e as ligações do grafo. Os nodos do grafo representam o nível de refinamento inicial da triangulação, isto é, o refinamento de nível zero. (Figuras extraídas de [69].)

Aplica-se a subdivisão 4T-LE de Rivara [122], mostrada na subseção 2.2, na página 18, para a subdivisão de um triângulo. Uma malha com um triângulo subdividido e o grafo resultantes são mostrados na figura 5.4. As fronteiras do domínio fechado são omitidas nessa figura em busca de clareza na apresentação.

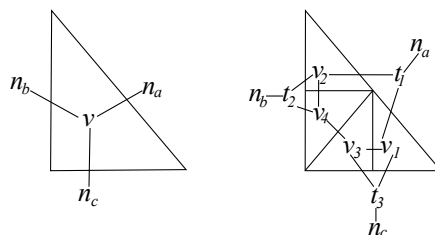


**Figura 5.4:** Grafo e triangulação correspondentes. Constam nodos de triângulos e de transição no subgrafo identificados com um círculo azul por causa do refinamento adaptativo em um triângulo da triangulação, identificado também em azul. (Figuras adaptadas de [69].)

Ao considerar um nodo de triângulo  $v$  com suas ligações para os nodos de triângulos  $n_a$ ,  $n_b$  e  $n_c$  como mostrado na imagem à esquerda da figura 5.5, ao refinar o nodo de triângulo  $v$ ,  $v$  é substituído pelos nodos de triângulos  $v_1$ ,  $v_2$ ,  $v_3$ ,  $v_4$  e pelos nodos de transição  $t_1$ ,  $t_2$ , and  $t_3$ . As ligações entre nodos do subgrafo são estabelecidas de acordo, incluindo em relação a  $n_a$ ,  $n_b$  e  $n_c$ . Isso é exemplificado na imagem à direita da figura 5.5.

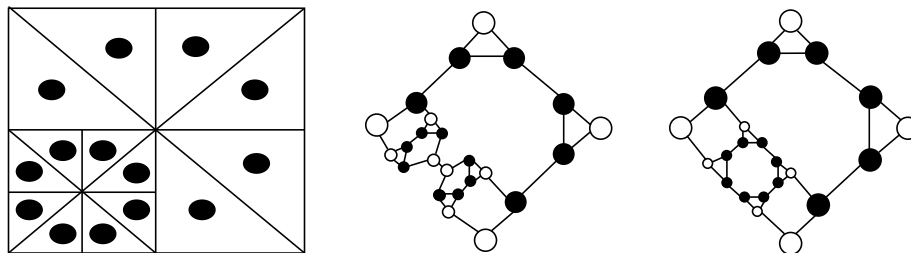
Outro refinamento realizado na triangulação mostrada na figura 5.4 é mostrado





**Figura 5.5:** O nodo de triângulo  $v$ , com ligações para os nodos de triângulos  $n_a$ ,  $n_b$  e  $n_c$  na imagem à esquerda é substituído pelos nodos de triângulos  $v_1$ ,  $v_2$ ,  $v_3$ ,  $v_4$  e pelos nodos de transição  $t_1$ ,  $t_2$ , and  $t_3$ , mostrados na imagem à direita. (Figuras extraídas de [69].)

na figura 5.6. Como exemplificado na figura 5.6, nodos de transição não são armazenados se há nodos de triângulos com os mesmos níveis de refinamento, isto é, quando nodos de triângulos representam triângulos na malha com interface em comum.



**Figura 5.6:** Refinamento de outro triângulo da triangulação mostrada na figura 5.4, uma representação intermediária e o grafo final que representa essa nova triangulação. (Figuras adaptadas de [69].)

Nas figuras 5.7 e 5.8, são mostrados exemplos de triangulações compostas por triângulos retângulos isósceles e escalenos, respectivamente. Ambas as triangulações no quadrado unitário foram representadas pelo grafo descrito.

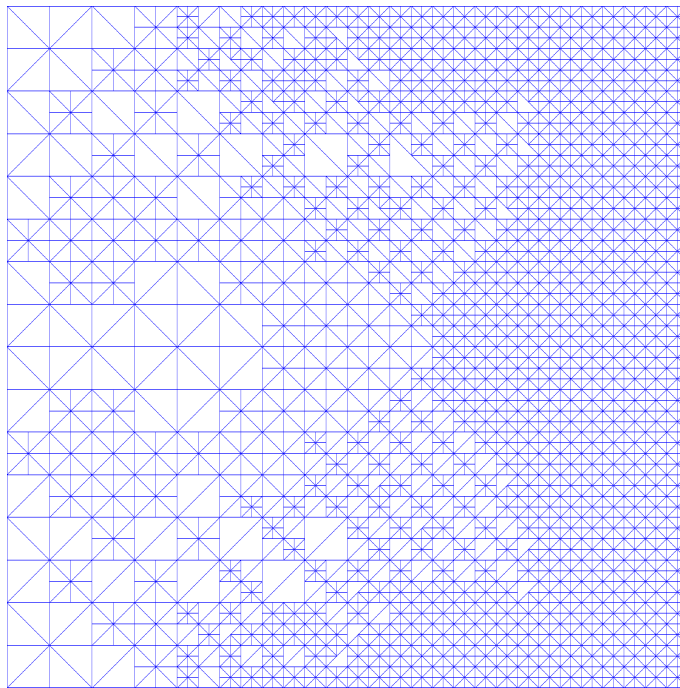
A seguir, o processo de simplificação local é mostrado na subseção 5.5.1. A ordenação total dos triângulos da malha é descrita na subseção 5.5.2.

### 5.5.1 Simplificação local

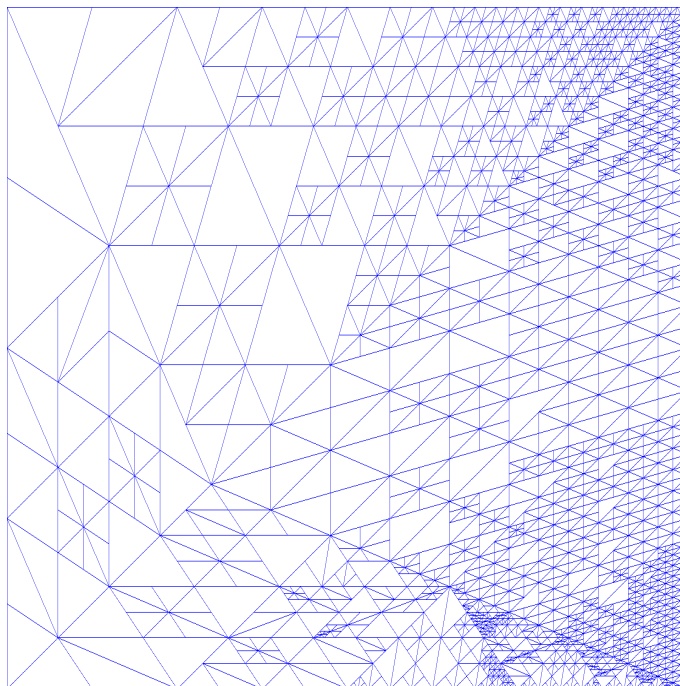
O processo de refinamento é reversível, isto é, a operação de simplificação local na triangulação é permitida. Ainda, são excluídos os nodos de triângulos em que a operação de simplificação local foi realizada e seus nodos de transição correspondentes.

Há um atributo, em cada nodo do subgrafo composto pelos quatro nodos de triângulos e seus nodos de transição correspondentes, criados a partir do refinamento de um triângulo, que indica que pertencem ao mesmo subgrafo. No processo de simplificação local, esse subgrafo é substituído por um nodo de triângulo e o nível de refinamento desse nodo é decrementado de uma unidade, em relação ao nível dos nodos de triângulos em que a operação de simplificação local foi realizada.

Se um nodo  $u$  de triângulo adjacente a esse subgrafo em que a operação de simplificação local foi realizada é do mesmo nível de refinamento do nodo de triângulo resultante  $v$ , ligações são criadas diretamente entre  $v$  e  $u$ . Caso contrário, nodos



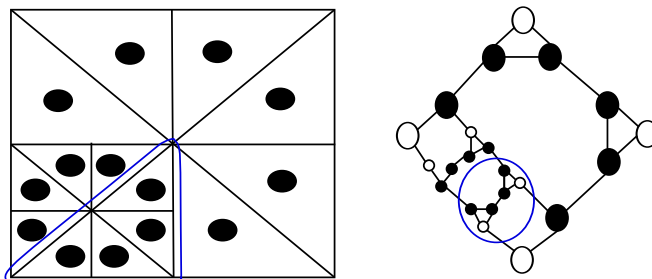
**Figura 5.7:** Exemplo de triangulação com 4739 triângulos retângulos isósceles representada pelo grafo descrito nesta seção.



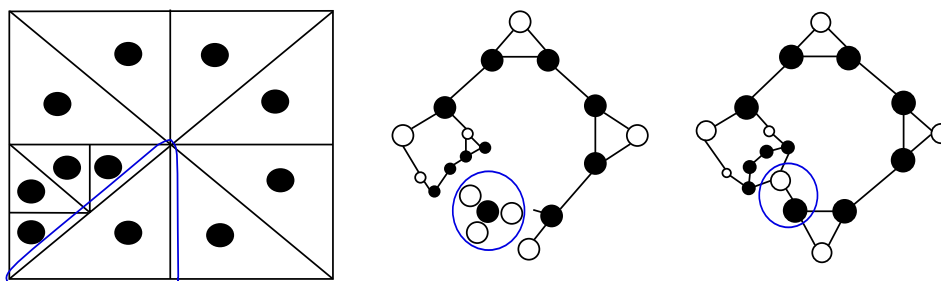
**Figura 5.8:** Exemplo de triangulação com 2948 triângulos escalenos representada pelo grafo descrito nesta seção.

de transição são criados para conectar nodos de triângulos com diferentes níveis de refinamento.

Como exemplo, é mostrada uma operação de simplificação local aplicada no subgrafo identificado por um círculo azul na figura 5.9. Esse subgrafo é composto por quatro nodos de triângulos e seus nodos de transição correspondentes. Esse subgrafo é substituído por um nodo de triângulo, como exemplificado na figura 5.10. A configuração final da triangulação é mostrada no grafo à direita da figura 5.10, seguida da representação intermediária desse grafo.



**Figura 5.9:** Grafo e triangulação correspondentes. Uma operação de simplificação local é aplicada no subgrafo que representa os triângulos identificados em azul na imagem à direita. Esse subgrafo do grafo na imagem à esquerda é identificado por um círculo azul. (Figuras adaptadas de [69].)



**Figura 5.10:** Triangulação à esquerda e grafo à direita correspondentes. O triângulo identificado em azul é representado pelos nodos do subgrafo identificado no círculo em azul do grafo à direita. É realizada a operação de simplificação local no subgrafo identificado na figura 5.9, o grafo intermediário é mostrado ao meio e a configuração final do grafo após a simplificação local é mostrada à direita. (Figuras adaptadas de [69].)

### 5.5.2 Ordenação dos triângulos da malha

Considere a seguinte definição de caminho em um grafo.

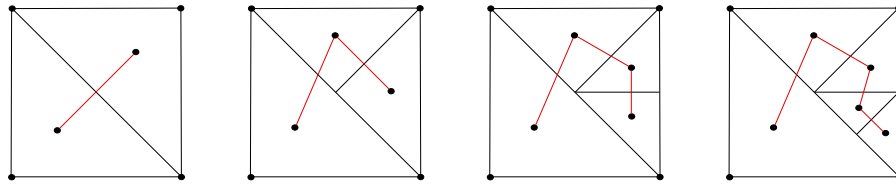
**Definição 5.1.** (*caminho em um grafo*) Um caminho é uma sequência de vértices adjacentes e arestas de um grafo  $G = (V, E)$ , tal que formam um subgrafo conexo  $G' = (V', E')$ , em que  $V' \subseteq V$  e  $E' \subseteq E$ .

É comum a tarefa de percorrer todos os polítopos de uma malha. Esse problema pode ser descrito como o problema do caminho hamiltoniano, que é o problema de encontrar um caminho simples em vértices de um grafo. Em termos simples, um

caminho é simples se não há repetição de vértices no caminho e um caminho em um grafo é dito ser hamiltoniano se todos os nodos são visitados uma só uma vez.

A partição de triângulos retângulos pela subdivisão 4T-LE de Rivara [122], mostrada na seção 2.2, na página 18, permite a utilização de uma curva de Sierpiński modificada para ordenar os nodos de triângulos do grafo. A curva de Sierpiński modificada tem um baixo custo computacional para ser gerada [147, 68]. Em particular, a curva de preenchimento de espaço utilizada é uma curva de Pólya [116] se somente triângulos retângulos isósceles são utilizados, como exemplificado na figura 5.7. Em termos simples, uma curva de preenchimento de espaço é uma curva cujo intervalo contém todo o quadrado unitário, ou de uma forma generalizada, o hipercubo  $n$ -dimensional.

A curva de Sierpiński modificada para a ordenação total dos triângulos da malha pode ser implementada como uma lista duplamente encadeada. Com isso, somente atualizações locais são realizadas durante operações de refinamento e simplificação locais. Exemplos de refinamentos adaptativos sucessivos em que o triângulo é subdividido por bisseção em um domínio quadrangular e ordenação pela curva de Sierpiński modificada são mostrados na figura 5.11.



**Figura 5.11:** Subdivisões sucessivas de triângulos por bisseção. Os triângulos são ordenados pela curva de Sierpiński modificada, em vermelho.

Esse esquema permite atualização trivial na lista duplamente encadeada para a ordenação total dos triângulos da malha. Como descrito, é aplicada a subdivisão 4T-LE de Rivara [122] nesse processo. No esquema de refinamento original de Rivara, o refinamento é propagado em triângulos adjacentes para se manter a conformidade da malha. Entretanto, aqui, não se realiza a propagação do refinamento já que a conformidade da triangulação não é exigida.

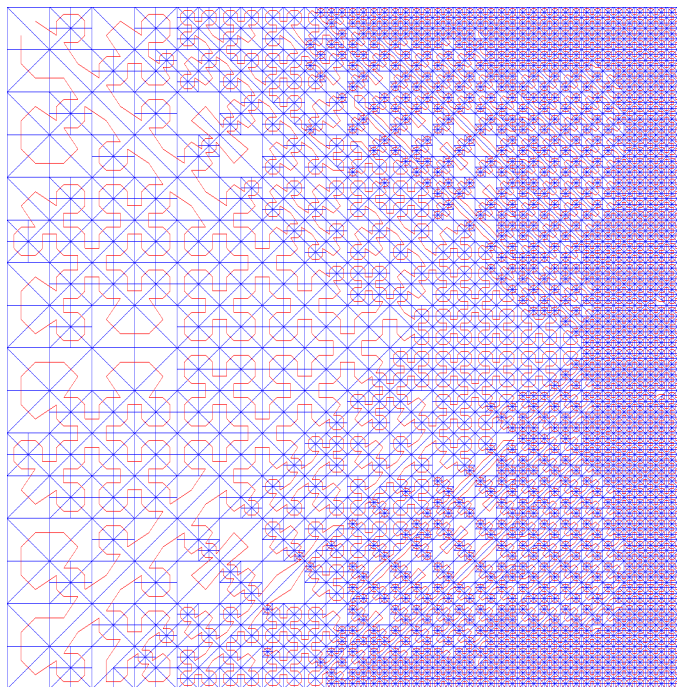
Uma triangulação sequencial generalizada, ou uma *triangulação hamiltoniana*, é uma triangulação em que há um conjunto ordenado  $t_1, \dots, t_n$  de todos os seus  $n$  triângulos de forma que triângulos consecutivos  $t_i$  e  $t_{i+1}$  compartilham uma aresta. Cada triângulo em uma triangulação hamiltoniana tem uma aresta de entrada e uma aresta de saída em relação à ordem hamiltoniana e o conhecimento de todas essas arestas caracteriza completamente a sequência [147]. Geometricamente, uma sequência triangular generalizada pode ser representada ao se construir um caminho orientado na discretização. Nesse caminho, todos os triângulos são percorridos ao percorrer suas arestas de entrada e de saída.

O problema de encontrar a melhor sequência de triângulos de uma triangulação é NP-Difícil [50]. Todavia, ao se adicionar novos pontos, é sempre possível refinar um triângulo em uma triangulação hamiltoniana [3]. Como tal, o esquema mostrado gera uma triangulação sequencial generalizada desde que o refinamento seja estritamente local.

Como a subdivisão 4T-LE é aplicada em triângulos retângulos, uma triangulação hamiltoniana é obtida e cada atualização na lista duplamente encadeada é estritamente local.

tamente local. Isso segue a atualização estritamente local do grafo, gerando ou eliminando um subgrafo gerado na mesma operação de refinamento ou de simplificação local, respectivamente. Isso significa que somente ligações locais da curva de Sierpiński modificada são atualizadas ao criar ou eliminar um subgrafo em uma operação de refinamento ou de simplificação local, respectivamente. Esse processo de atualização das estruturas de dados envolvidas é executado em  $\Theta(1)$  porque somente realiza atualizações locais no grafo e na curva de Sierpiński modificada (já construídos). Com isso, claramente, os triângulos da triangulação são percorridos em tempo linear no número de triângulos.

Na figura 5.12, mostra-se um exemplo de uma triangulação representada pelo grafo apresentado nesta seção. A triangulação é mostrada em azul e a curva de Sierpiński modificada é mostrada em vermelho. Como descrito, com a triangulação inicial composta por triângulos retângulos isósceles, foi utilizada a subdivisão 4T-LE [122], que divide o triângulo pelo ponto médio da hipotenusa (claramente, a maior aresta do triângulo retângulo). Isso permite ordenar a triangulação pela curva de Sierpiński modificada.



**Figura 5.12:** Uma malha com 12251 triângulos no quadrado unitário. A triangulação, mostrada em azul, é representada pelo grafo descrito nesta seção e a curva de Sierpiński modificada é mostrada em vermelho.

Por sua vez, não é possível utilizar a curva de Sierpiński modificada no caso da triangulação mostrada na figura 5.8, que é composta por triângulos escalenos. Isso porque, para a curva de Sierpiński modificada ser utilizada, os triângulos deveriam ter um padrão de refinamento: sempre serem divididos por um ponto de uma aresta padrão. Sem subdividir os triângulos pelo ponto médio da maior aresta, a qualidade dos triângulos pioraria rapidamente. Com isso, a subdivisão 4T-LE foi aplicada e, para serem percorridos, novos triângulos foram inseridos ao final de uma lista encadeada simples. Por isso, foi utilizado o método Cuthill-McKee reverso [60],

em que a sequência é iniciada pelo algoritmo de George-Liu [58, 59], para se obter localidade de memória na sequência dos triângulos armazenados no grafo. Detalhes sobre esses e outros algoritmos para fornecer sequências de vértices de grafos são descritos por Gonzaga de Oliveira e Chagas [66].

## 5.6 Custos de armazenamento

De Floriani, Kobbelt e Puppo [41] explicam que a estrutura de dados para malhas triangulares mais comum é conhecida como *estrutura de dados indexada*: são mantidos três índices para os vértices em cada triângulo. O custo dessa estrutura de dados, sem levar em consideração informações sobre geometria e atributos dependentes da aplicação, é de, aproximadamente,  $6n$ , em que  $n$  é o número de vértices da triangulação. O custo de armazenamento dessa estrutura de dados é mostrada na última linha da tabela 5.4.

**Tabela 5.4:** Custos de armazenamento de estruturas de dados para geração de malhas.

Estrutura de dados	Índices por vértice ( $\cong$ )	Entidade primária
<i>winged-edge</i> [11]	27	aresta
<i>half-edge</i> [93]		
<i>Doubly-Connected Edge List</i> [96]	21	
de Botsch et al. [18]		
<i>directed edges</i> [27]	13	triângulo
indexada com adjacências de Lawson [84]		
IA [99, 108]		
<i>Triangle-Segment</i> [42]		
$\sphericalangle$ [69]	8	
indexada	6	

A estrutura de dados indexada foi estendida por Lawson [84] para que fosse possível ser percorrida pelas arestas ao armazenar as arestas e os três triângulos adjacentes de cada triângulo. Essa estrutura de dados de Lawson [84] é chamada de *estrutura de dados indexada com adjacências*. Para ser possível percorrer a estrutura de dados de forma eficiente, ou seja, para que seja possível ser percorrida também pelos vértices, De Floriani, Kobbelt e Puppo [41] explicam que é conveniente armazenar, em cada vértice da malha, uma ligação a um dos seus triângulos incidentes. O custo de armazenamento dessa estrutura de dados estendida, sem levar em consideração informações sobre geometria e atributos dependentes da aplicação, é de  $13n$ , como mostrado na tabela 5.4.

Na estrutura de dados *half-edge* [93], são codificadas duas cópias da mesma aresta, orientadas em direções opostas. Para cada aresta da malha, são armazenadas duas “meia” arestas, e as informações sobre a aresta é dividida entre essas duas metades. Cada meia aresta armazena o índice para seu vértice de saída (ou de origem), o índice de sua face  $F$  à esquerda, dois índices de suas arestas predecessora e sucessora em relação à face  $F$  e os índices de sua aresta “gêmea”, que tem orientação na direção oposta. Para saber o vértice de chegada (ou de destino) da aresta, deve-se verificar o vértice de saída da aresta “gêmea”. As informações armazenadas nos vértices e faces são as mesmas que as descritas nas estruturas de dados *winged-edge* e DCEL, descritas nas seções 5.3 e 5.4, respectivamente. Segundo DeFloriani,

Kobbelt e Puppo [41], o custo de armazenamento da estrutura de dados *half-edge* é a mesma da estrutura de dados *winged-edge*. Campagna, Kobbelt e Seidel [27] propuseram a estrutura de dados *directed edges*, que é uma implementação eficiente da estrutura de dados *half-edge* e tem custo de armazenamento de  $13n$ . Botsch e colaboradores [18] também publicaram uma implementação da estrutura de dados *half-edge*, com custo de armazenamento de  $21n$  [27].

A estrutura de dados *Triangle-Segment* [42] foi proposta para representar triangulações gerais. Como descrito por De Floriani e Hui [39], essa estrutura de dados é uma extensão da estrutura de dados *indexada com adjacências* (IA) [99, 108] e de três estruturas de dados baseadas em arestas: *partial edge* [87]; *loop edge-use* [95], que é uma especialização da estrutura de dados *radial edge* [153]; *directed edges* [27]. As especializações das estruturas de dados *partial edge* e *loop edge-use* são descritas em detalhes por De Floriani e Hui [40]. O custo de armazenamento de ambas as estruturas de dados indexada com adjacências (IA) [99, 108] e *Triangle-Segment* [42] é de  $13n$ .

Custos de armazenamento de diversas outras estruturas de dados para geração de malhas foram apresentadas por De Floriani e colaboradores [41, 42, 39]. Custos de armazenamento de estruturas de dados que pode ser utilizadas para representar malhas triangulares (bidimensionais) são apresentados na tabela 5.4, supondo-se que o tamanho de um índice é uma unidade e sem levar em consideração informações geométricas e atributos dependentes da aplicação. O custo de armazenamento da estrutura de dados para representar malhas triangulares, mostrada na seção 5.5, na página 70, e representada como  $\triangleleft$  na tabela 5.4, é de  $8n$  [69].

Na terceira coluna da tabela 5.4, mostra-se a entidade primária que a estrutura de dados codifica: aresta ou triângulo. Em geral, as estruturas de dados que codificam triângulos como entidade primária têm custo de armazenamento menor que as estruturas de dados que codificam arestas como entidade primária. Em particular, as estruturas de dados *Triangle-Segment* [42] e  $\triangleleft$  [69] necessitam de 7 e de 8 nodos da estrutura de dados para cada polígono da malha, respectivamente.

## 5.7 Notas bibliográficas

Publicada em 2004, a revisão de De Floriani, Kobbelt e Puppo [41] sobre estruturas de dados para malhas bidimensionais e tridimensionais ainda é bastante atual. As estruturas de dados *quadtrees* e *octrees* não foram abordadas neste texto por serem mais utilizadas com malhas quadrilaterais. Exemplo de texto em que *quadtrees* são explicadas em detalhes é no capítulo 14 do livro *Computational Geometry*, de Berg e colaboradores [15]. Também, um exemplo de texto em que métodos baseados em *quadtrees* ou em *octrees* são explicados em detalhes é no capítulo 5 do livro *Mesh Generation, application to Finite Elements*, de Frey e George [56].

## 5.8 Exercícios

1. Considere uma orientação em sentido anti-horário do tetraedro mostrado na figura 5.1 e mostre a sua representação pela estrutura de dados *winged-edge*.
2. Considere o grafo e a triangulação mostrados na figura 5.4 e faça o que se pede.
  - (a) Realize a subdivisão do triângulo do lado direito do triângulo identificado em azul e mostre o grafo correspondente.

- (b) Em seguida, realize a subdivisão do triângulo imediatamente acima do triângulo particionado e mostre o grafo correspondente.
  - (c) Em seguida, realize a simplificação do triângulo refinado em (a) e mostre o grafo correspondente.
3. Explique o que é uma curva de preenchimento de espaço.
  4. Explique a estrutura de dados indexada para representação de triangulações.
  5. Explique as extensões realizadas por Lawson [84] na estrutura de dados indexada para representação de triangulações.



# Bibliografia

- [1] ALLIEZ, P.; MEYER, M.; DESBRUN, M. Interactive geometry remeshing. *ACM Transactions on Graphics*, ACM, New York, NY, v. 21, n. 3, p. 347–354, 2002.
- [2] AMENTA, N.; BERN, M.; EPPSTEIN, D. Optimal point placement for mesh smoothing. In: *Proceedings of the Eighth Annual Symposium on Discrete Algorithms*. New Orleans, Louisiana: Association for Computing Machinery, 1997. p. 528–537.
- [3] ARKIN, E. et al. Hamiltonian triangulations for fast rendering. In: *The 2nd Annual European Symposium on Algorithms*. Utrecht, The Netherlands: Springer, 1994. (Lecture Notes in Computer Science, LNCS, v. 855), p. 36–47.
- [4] AURENHAMMER, F. Power diagrams: Properties, algorithms, and applications. *SIAM Journal on Computing*, v. 16, n. 1, p. 78–96, February 1987.
- [5] BABUSKA, I.; AZIZ, A. K. On the angle condition in the finite element method. *SIAM Journal on Numerical Analysis*, New Providence, New Jersey, v. 13, n. 2, p. 214–226, 1976.
- [6] BAKER, C. M. J. et al. Goal based mesh adaptivity for fixed source radiation transport calculations. *Annals of Nuclear Energy*, v. 55, p. 169–183, 2013.
- [7] BAKER, T. J. Element quality in tetrahedral meshes. In: *Proceedings of the 7th International Conference Finite Element Methods in Flow Problems*. Huntsville, AL: AL: 35899, 1989. p. 1018–1024.
- [8] BANK, R. E.; SMITH, R. K. Mesh smoothing using a posteriori error estimates. *SIAM Journal On Numerical Analysis*, Philadelphia, v. 34, n. 3, p. 979–997, 1997.
- [9] BANK, R. E.; XU, J. An algorithm for coarsening unstructured meshes. *Numerische Mathematik*, v. 73, p. 1–36, 1996.
- [10] BARTH, T. B. Aspects of unstructured grids and finite - volume solvers for the Euler and Navier-Stokes equations. *Von Karman Institute for Fluid Dynamics Lecture Series, NASA Ames Research Center*, 1995.
- [11] BAUMGART, B. G. A polyhedron representation for computer vision. In: *Proceedings of American Federation of Information Processing Societies (AFIPS) National Computer Conference and Exposition*. Anaheim, CA: ACM, 1975. v. 44, p. 589–596.
- [12] BENTLEY, J. L. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, v. 18, n. 9, p. 509, 1975.

- [13] BENTLEY, J. L.; WEIDE, B. W.; YAO, A. C. Optimal expected-time algorithms for closest point problems. In: *16th Proceedings - Annual Allerton Conference on Communication, Control, and Computing*. Monticello, IL: [s.n.], 1978. p. 843–851.
- [14] BENTLEY, J. L.; WEIDE, B. W.; YAO, A. C. Optimal expected-time algorithms for closest point problems. *ACM Transactions on Mathematical Software*, ACM, v. 6, p. 563–580, 1980.
- [15] BERG, M. de et al. *Computational Geometry: Algorithms and Applications*. 3rd. ed. Berlin: Springer, 2008.
- [16] BERN, M. W.; EPPSTEIN, D. Mesh generation and optimal triangulation. In: DU, D.-Z.; HWANG, F. K.-M. (Ed.). *Computing in Euclidean Geometry*. second. California: World Scientific, 1995, (Lecture Notes Series on Computing, 4). p. 47–123.
- [17] BOOTS, B. N. Weighting thiesen polygons. *Economic Geography*, v. 56, n. 3, p. 248–259, July 1979.
- [18] BOTSCH, M. et al. Openmesh - a generic and efficient polygon mesh data structure. In: REINERS, D. (Ed.). *Proceedings of the OpenSG Symposium*. Darmstadt, Germany: [s.n.], 2002.
- [19] BOWYER, A. Computing dirichlet tessellations. *The Computer Journal*, Oxford University Press, v. 24, n. 2, p. 162–166, 1981.
- [20] BRAMBLE, J. H.; ZLAMAL, M. Triangular elements in the fiite element method. *Mathematics of Computation*, American Mathematical Society, v. 24, n. 112, p. 809–820, October 1970.
- [21] BRANDÃO, D. N.; GONZAGA DE OLIVEIRA, S. L.; KISCHINHEVSKY, M. Finite-element non-conforming h-adaptive strategy based on autonomous leaves graph. In: ALLEN, G. et al. (Ed.). *9th International Conference on Computational Science (ICCS)*. Baton Rouge, LA: Springer, 2009. (Lecture Notes in Computer Science, LNCS, v. 5544 - Volume Part I), p. 570–579.
- [22] BROWN, K. Q. Voronoi diagrams from convex hulls. *Information Processing Letters*, v. 9, n. 5, p. 223–228, December 1979.
- [23] BROWN, K. Q. *Geometric transforms for fast geometric algorithms*. Tese (Doutorado) — Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1980. Available as Technical Report CMU-CS-80-101.
- [24] BURGARELLI, D. D.; KISCHINHEVSKY, M.; BIEZUNER, R. J. A new adaptive mesh refinement strategy for numerically solving evolutionary PDE's. *Journal of Computational and Applied Mathematics*, v. 196, p. 115–131, 2006.
- [25] BUSARYEV, O.; DEY, T. K.; WANG, H. Adaptive fracture simulation of multi-layered thin plates. *ACM Transactions on Graphics*, ACM, New York, NY, v. 32, n. 4, p. 52:1–52:6, 2013.
- [26] CALVO, J. A. *A Mathematical Tour of Functions*. 3th ed.. ed. Raleigh, N.C.: Lulu.com, 2014.

- [27] CAMPAGNA, S.; KOBELT, L.; SEIDEL, H.-P. Directed edges – A scalable representation for triangle meshes. *ACM Journal of Graphics Tools*, v. 3, n. 4, p. 1–12, 1998.
- [28] CANANN, S. A.; STEPHENSON, M.; BLACKER, T. Optismoothing: An optimization-driven approach to mesh smoothing. *Finite Elements in Analysis and Design*, v. 13, p. 185–190, 1993.
- [29] CAO, W.; HUANG, W.; RUSSELL, R. D. Comparison of two-dimensional r-adaptive finite element methods using various error indicators. *Mathematics and Computers in Simulation*, v. 56, p. 127–143, 2001.
- [30] CAVENDISH, J. C.; FIELD, D. A.; FREY, W. H. An approach to automatic three-dimensional nite element mesh generation. *International Journal of Numerical Methods in Engineering*, v. 21, p. 329–347, 1985.
- [31] CHEN, L.; HOLST, M. Efficient Mesh Optimization Schemes based on Optimal Delaunay Triangulations. *Computer Methods in Applied Mechanics and Engineering*, v. 200, p. 967–984, 2011.
- [32] CHENG, S.-W.; DEY, T. K.; SHEWCHUK, J. R. *Delaunay Mesh Generation*. Boca Raton, FL: CRC Press, Taylor & Francis Group, 2013.
- [33] COURANT, R.; FRIEDRICHS, K.; LEWY, H. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, v. 100, n. 1, p. 32–74, 1928.
- [34] COURANT, R.; FRIEDRICHS, K.; LEWY, H. *On the partial difference equations of mathematical physics*. AEC Research and Development Report, AEC Computing and Applied Mathematics Centre - Courant Institute of Mathematical Sciences. Technical report NYO-7689. New York, September 1956. V + 76 p.
- [35] COURANT, R.; FRIEDRICHS, K.; LEWY, H. On the partial difference equations of mathematical physics. *IBM Journal of Research and Development*, v. 11, n. 2), p. 215–234, March 1967.
- [36] D’AZEVEDO, E. F.; SIMPSON, R. B. On optimal interpolation triangle incidences. *SIAM Journal on Scientific and Statistical Computing*, v. 6, p. 1063–1075, 1989.
- [37] DE BOOR, C. *Good approximation by splines with variable knots. Spline Functions and Approximation Theory*. Basel: Birkhauser Verlag, 1973. 57–73 p.
- [38] DE FLORIANI, L.; FALCIDIENO, B.; PIENOVI, C. Delaunay-based representation of surfaces defined over arbitrarily shaped domains. *Computer Vision, Graphics and Image Processing*, v. 32, p. 127–140, 1985.
- [39] DE FLORIANI, L.; HUI, A. Data structures for simplicial complexes: an analysis and a comparison. In: DESBRUN, M.; POTTMANN, H. (Ed.). *Eurographics Symposium on Geometry Processing*. Vienna, Austria: The Eurographics Association and Blackwell Publishing, 2005.
- [40] DE FLORIANI, L.; HUI, A. *A survey on data structures for simplicial complexes*. Computer and Information Science Department (DISI), University of Genova. Technical report. Genova, Italy, May 2005.

- [41] DE FLORIANI, L.; KOBELT, L.; PUPPO, E. A survey on data structures for level-of-detail models. In: DODGSON, N.; FLOATER, M.; SABIN, M. (Ed.). *Advances in Multiresolution for Geometric Modelling, Series in Mathematics and Visualization*. New York, NY: Springer Verlag, 2004. (Series in Mathematics and Visualization), p. 49–74.
- [42] DE FLORIANI, L. et al. A multi-resolution topological representation for non-manifold meshes. *Computer-Aided Design Journal*, v. 36, n. 2, p. 141–159, 2004.
- [43] DE SANTIS, G. *Novel Mesh Generation Method for Accurate Image-Based Computational Modelling of Blood Vessels*. XVI, 184 p. Tese (Doutorado) — Ghent University. Faculty of Engineering and Architecture, Gent, Belgium, October 2011.
- [44] DELAUNAY, B. N. Sur la sphere vide. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk (Bulletin of Academic Science URSS, Class. Science National)*, v. 7, p. 793–800, 1934.
- [45] DESCARTES, R. *Principia Philosophiae*. Amsterdam: Ludovicus Elzevirius, 1644.
- [46] DIRICHLET, J. P. G. L. Über die reduction der positiven quadratischen formen mit drei unterestimmten ganzen zahlen. *Journal für die Reine und Angewandte Mathematik*, v. 40, p. 209–227, 1850.
- [47] DONG, L.; ATLURI, S. N. T-Trefftz Voronoi cell finite elements with elastic/rigid inclusions or voids for micromechanical analysis of composite and porous materials. *CMES: Computer Modeling in Engineering & Sciences*, v. 83, n. 2, p. 183–219, 2012.
- [48] EDELSBRUNNER, H. Triangulations and meshes in computational geometry. *Acta Numerica*, v. 9, p. 133–213, 2000.
- [49] EDELSBRUNNER, H. *Geometry and topology for mesh generation*. New York, NY: Cambridge University Press, 2001. (Cambridge monographs on applied and computational mathematics).
- [50] EVANS, F.; SKIENA, S.; VARSHNEY, A. *Completing sequential triangulations is hard*. State University of New York, Department of Computer Science. Technical report. Stony Brook, NY, 1996.
- [51] FELIPPA, C. A. Solution of linear equations with skyline-stored symmetric matrix. *Computers & Structures*, v. 5, p. 13–29, April 1975.
- [52] FREDERICK, C. O.; WONG, Y. C.; EDGE, F. W. Two-dimensional automatic mesh generation for structural analysis. *International Journal for Numerical Methods in Engineering*, v. 2, p. 133–144, 1970.
- [53] FREITAG, L. A.; JONES, M.; PLASSMANN, P. E. An efficient parallel algorithm for mesh smoothing. In: *Proceedings of the 4th International Meshing Roundtable*. Albuquerque, New Mexico: Sandia National Laboratories, 1995. p. 47–58.
- [54] FREITAG, L. A.; JONES, M.; PLASSMANN, P. E. A parallel algorithm for mesh smoothing. In: *Proceedings of the Eighth SIAM Conference on Parallel Processing for Scientific Computing*. Minneapolis: SIAM, 1997.

- [55] FREITAG, L. A.; JONES, M.; PLASSMANN, P. E. An efficient parallel algorithm for mesh smoothing. *SIAM Journal on Scientific Computing*, v. 20, n. 6, p. 2023–2040, 1999.
- [56] FREY, P. J.; GEORGE, P.-L. Mesh generation, application to finite elements. *ISTE Ltd.*, Londres, 2008.
- [57] GABRIEL, K. R.; SOKAL, R. R. A new statistical approach to geographic variation analysis. *Systematic Zoology (Society of Systematic Biologists)*, v. 18, n. 3, p. 259–270, 1969.
- [58] GEORGE, A.; LIU, J. W. H. An implementation of a pseudoperipheral node finder. *ACM Transactions on Mathematical Software*, n. 5, p. 284–295, 1979.
- [59] GEORGE, A.; LIU, J. W. H. *Computer solution of large sparse positive definite systems*. Englewood Cliffs: Prentice-Hall, 1981.
- [60] GEORGE, J. A. *Computer implementation of the finite element method*. 228 p. Tese (Doutorado) — Computer Science Department, Stanford University, Stanford, CA, 1971.
- [61] GKOUNTOUVAS, T. et al. Improving the performance of the symmetric sparse matrix-vector multiplication in multicore. In: *27th International Symposium on Parallel & Distributed Processing (IPDPS)*. Boston, MA: IEEE, 2013. p. 273–283.
- [62] GOFFIN, M. A. et al. Minimising the error in eigenvalue calculations involving the boltzmann transport equation using goal-based adaptivity on unstructured meshes. *Journal of Computational Physics*, v. 242, p. 726–752, 2013.
- [63] GONZAGA DE OLIVEIRA, S. L. *Algoritmos e seus fundamentos*. Lavras, MG: Editora UFLA, 2011.
- [64] GONZAGA DE OLIVEIRA, S. L. An overview of procedures for refining triangulations. In: MURGANTE, B. et al. (Ed.). *Proceedings of the 12th international conference on Computational Science and Its Applications (ICCSA)*. Salvador, BA: Springer-Verlag, 2012. (Lecture Notes in Computer Science, LNCS, v. 7333 - Volume Part I), p. 198–213.
- [65] GONZAGA DE OLIVEIRA, S. L. A review of Delaunay refinement techniques. In: MURGANTE, B. et al. (Ed.). *Proceedings of the 12th international conference on Computational Science and Its Applications (ICCSA)*. Salvador, BA: Springer-Verlag, 2012. (Lecture Notes in Computer Science, LNCS, v. 7333 - Volume Part I), p. 172–187.
- [66] GONZAGA DE OLIVEIRA, S. L.; CHAGAS, G. O. *Introdução a heurísticas para redução de largura de banda de matrizes*. São Carlos, Brasil: Sociedade Brasileira de Matemática Aplicada e Computacional (SBMAC), 2014.
- [67] GONZAGA DE OLIVEIRA, S. L.; CHAGAS, G. O. Adaptive mesh refinement for finite-volume discretizations with scalene triangles. *Procedia Computer Science (ICCS 2015 International Conference On Computational Science)*, v. 51, p. 239–245, 2015.

- [68] GONZAGA DE OLIVEIRA, S. L.; KISCHINHEVSKY, M. Sierpiński curve for total ordering of a graph-based adaptive simplicial-mesh refinement for finite volume discretizations. In: *31o. Congresso Nacional de Matemática Aplicada e Computacional (CNMAC)*. Belém, PA: Sociedade Brasileira de Matemática Aplicada e Computacional (SBMAC), 2008. p. 581–585.
- [69] GONZAGA DE OLIVEIRA, S. L.; KISCHINHEVSKY, M.; TAVARES, J. M. R. S. Novel graph-based adaptive triangular mesh refinement for finite-volume discretizations. *CMES: Computer Modeling in Engineering & Sciences*, v. 95, n. 2, p. 119–141, 2013.
- [70] GONZAGA DE OLIVEIRA, S. L.; NOGUEIRA, J. R.; TAVARES, J. M. R. S. A systematic review of algorithms with linear-time behaviour to generate delaunay and voronoi tessellations. *CMES: Computer Modeling in Engineering & Sciences*, v. 100, n. 1, p. 31–57, 2014.
- [71] GONZAGA DE OLIVEIRA, S. L.; OLIVEIRA, F. S. de; CHAGAS, G. O. A novel approach to the weighted laplacian formulation applied to 2D delaunay triangulations. *Lecture Notes on Computer Science (LNCS - 15th International Conference on Computational Science and Its Applications - ICCSA)*, Springer International Publishing Switzerland, v. 9155, n. Part I, p. 502–515, 2015.
- [72] GREGORY, J. A. Error bounds for linear interpolation in triangles. In: WHITEMAN, J. R. (Ed.). *The Mathematics of Finite Elements and Application II*. London: Academic Press, 1975. p. 163–170.
- [73] GUIBAS, L. J.; STOLFI, J. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *ACM Transactions on Graphics*, v. 4, n. 2, p. 74–123, 1985.
- [74] GYVES, O. D.; TOLEDO, L.; RUDOMÍN, I. Proximity queries for crowd simulation using truncated Voronoi diagrams. In: *Proceedings of Motion on Games*. New York, NY: ACM, 2013. (MIG '13), p. 65:87–65:92.
- [75] HESTENES, M. R.; STIEFEL, E. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, v. 49, n. 36, p. 409–436, 1952.
- [76] HILBERT, D. Über die stetige abbildung einer linie auf ein flächenstück. *Mathematische Annalen*, v. 38, p. 459–460, 1891.
- [77] HOPCROFT, J. E.; TARJAN, R. E. Efficient algorithms for graph manipulations. *Communications of the ACM*, v. 16, p. 372–378, 1973.
- [78] HUANG, W.; RUSSELL, R. D. *Adaptive moving mesh methods*. New York, NY: Springer, 2011.
- [79] JENNINGS, A. A compact storage scheme for the solution of symmetric linear simultaneous equations. *The Computer Journal*, v. 9, n. 3, p. 281–285, 1966.
- [80] KAMEL, H. A.; EISENSTEIN, K. Automatic mesh generation in two- and three-dimensional inter-connected domains. In: *Symposium on High Speed Computing of Elastic Structures*. Liège, Belgium: Les Congrès et Colloques de l'Université de Liège, 1970.

- [81] KOBBELT, L. et al. Interactive multi-resolution modeling on arbitrary meshes. In: *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY: ACM, 1998. (ACM SIGGRAPH '98), p. 105–114.
- [82] LAMBERT, T. The Delaunay triangulation maximizes the mean inradius. In: *Proceedings of Sixth Canadian Conference on Computational Geometry*. Saskatoon, Saskatchewan, Canada: [s.n.], 1994. p. 201–206.
- [83] LANCZOS, C. Solutions of systems of linear equations by minimized iterations. *Journal of Research of the National Bureau of Standards*, v. 49, n. 3, p. 33–53, 1952.
- [84] LAWSON, C. L. Software for  $C^1$  surface interpolation. In: RICE, J. R. (Ed.). *Mathematical Software III*. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION, Washington, D.C: Academic Press, 1977. p. 161–194.
- [85] LEE, D.-T.; LIN, A. K. Generalized delaunay triangulations for planar graphs. *Discrete & Computational Geometry*, v. 1, p. 201–217, 1986.
- [86] LEE, D. T.; SCHACHTER, B. J. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer and Information Sciences*, Plenum Publishing Corporation, v. 9, n. 3, p. 219–242, June 1980.
- [87] LEE, S. H.; LEE, K. Partial-entity structure: a fast and compact non-manifold boundary representation based on partial topological entities. In: *Proceedings of the sixth ACM symposium on Solid modeling and applications*. Ann Arbor, Michigan: ACM Press, 2001. p. 159–170.
- [88] LIN, L.; WANG, X.; ZENG, X. Geometrical modeling of cell division and cell remodeling based on Voronoi tessellation method. *CMES: Computer Modeling in Engineering & Sciences*, v. 98, n. 2, p. 203–220, 2014.
- [89] LIU, J.-F.; YAN, J.-H.; LO, S. H. A new insertion sequence for incremental Delaunay triangulation. *Acta Mechanica Sinica*, Acta Mechanica Sinica, v. 29, n. 1, p. 99–109, 2013.
- [90] LIU, Y.; SNOEYINK, J. A comparison of five implementations of 3D Delaunay Tessellation. *Combinatorial and Computational Geometry*, MSRI Publications, v. 52, p. 439–458, 2005.
- [91] LO, S. H. Delaunay triangulation of non-uniform point distributions by means of multi-grid insertion. *Finite Elements in Analysis and Design*, v. 63, p. 8–22, 2013.
- [92] MADUREIRA, M. L. R. *Estudo do erro e processos adaptativos em problemas de enformação pelo método dos elementos finitos*. Tese (Doutorado) — Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 1996.
- [93] MÄNTYLÄ, M. *An Introduction to Solid Modeling*. Rockville, Maryland: Computer Science Press, 1988.
- [94] MCLAIN, D. H. Two dimensional interpolation from random data. *The Computer Journal*, v. 19, n. 2, p. 178–181, 1976.
- [95] MCMAINS, S. *Geometric Algorithms and Data Representation for Solid Free-form Fabrication*. Tese (Doutorado) — University of California at Berkeley, Berkeley, CA, 2000.

- [96] MUELLER, D. E.; PREPARATA, F. P. Finding the intersection of two convex polyhedra. *SIAM Theoretical Computer Science*, v. 7, p. 217–236, 1978.
- [97] NIE, Y. F.; LI, Y. Q.; WANG, L. Parallel node-based local tetrahedral mesh generation. *Computer Modeling in Engineering & Sciences (CMES)*, v. 83, n. 6, p. 575–597, 2012.
- [98] NIELSON, G.; FRANKE, R. Surface construction based upon triangulations. In: BARNHILL, R. E.; BOEHM, W. (Ed.). *Surfaces in Computer Aided Design*. Amsterdam: North-Holland, 1983. p. 163–177.
- [99] NIELSON, G. M. Tools for triangulation and tetrahedralizations and constructing functions defined over them. In: NIELSON H. HAGEN, H. M. G. M. (Ed.). *Scientific Visualization: overviews, Methodologies and Techniques, ch. 20*. Silver Spring, MD: IEEE Computer Society, 1997. p. 429–525.
- [100] NOGUEIRA, J. R. *Geração da triangulação de Delaunay pelo algoritmo de Green-Sibson*. Lavras-MG: Departamento de Ciência da Computação, Universidade Federal de Lavras, 2013. Monografia, Bacharelado em Ciência da Computação. Orientador: S. L. Gonzaga de Oliveira.
- [101] NOGUEIRA, J. R.; GONZAGA DE OLIVEIRA, S. L. Introdução sucinta à triangulação de Delaunay e suas propriedades. In: *Anais do I Congresso de Matemática Aplicada e Computacional - CMAC-SE 2011*. Uberlândia: SBMAC, 2011.
- [102] NOGUEIRA, J. R.; GONZAGA DE OLIVEIRA, S. L. Introduction to triangular mesh generation and the Delaunay triangulation. In: *Proceedings of the XXXII Iberian Latin-American Congress on Computational Methods in Engineering - XXXII CILAMCE (Mini-symposia Research Beginners on Computational Methods in Engineering)*. Ouro Preto: ABMEC, 2011.
- [103] NORDIN, A. et al. An approach to constraint-based and mass-customizable product design. *Journal of Computing and Information Science in Engineering*, ASME, v. 11, n. 011006, p. 1–7, March 2011.
- [104] OLIVEIRA, F. S. de. *Soluções numéricas de equações diferenciais parciais com discretização por volumes finitos e malhas refinadas adaptativamente e móveis*. Dissertação (Mestrado) — Universidade Federal de Lavras, Lavras, 2014.
- [105] OLIVEIRA, F. S. de et al. Malhas móveis para solução numérica de equações diferenciais parciais. *Revista de Sistemas de Informação da FSMA*, v. 11, p. 11–16, Junho 2013.
- [106] OLIVEIRA, T. H. de; GONZAGA DE OLIVEIRA, S. L. *Uma aplicação de volumes finitos com refinamento de Ruppert e malhas de qualidade por off-centers*. Departamento de Ciência da Computação, Universidade Federal de Lavras. Relatório técnico. Lavras, MG, Julho 2012.
- [107] OLIVEIRA, T. H. de; GONZAGA DE OLIVEIRA, S. L. Uma resolução da equação de Laplace por Volumes Finitos e diagrama de Voronoi com refinamento adaptativo e de Ruppert. In: *Proceedings of the SIBGRAPI 2012 - International Conference on Graphics, Patterns and Images - Workshop of Undergraduate Works (WUW)*. Ouro Preto: SBC, 2012.



- [108] PAOLUZZI, A. et al. Dimension-independent modeling with simplicial complexes. *ACM Transactions on Graphics*, v. 12, n. 1, p. 56–102, 1993.
- [109] PARTHASARATHY, V. T.; GRAICHEN, C. M.; HATHAWAY, A. F. A comparison of tetrahedral quality measures. *Finite Elements in Analysis and Design*, v. 15, p. 255–261, 1993.
- [110] PEANO, G. Sur une courbe, qui remplit toute une aire plane. *Mathematische Annalen*, v. 36, n. 1, p. 157–160, 1890.
- [111] PEBAY, P. P.; BAKER, J. B. A comparison of triangle quality measure. In: *Proceedings of the 8th International Meshing Roundtable*. South Lake Tahoe, CA: Sandia National Laboratories, 2001. p. 327–340.
- [112] PÉBAY, P. P.; BAKER, T. J. Analysis of triangle quality measures. *Mathematics of Computation*, American Mathematical Society, Boston, MA, v. 72, p. 1817–1839, october 2003.
- [113] PÉBAY, P. P.; BAKER, T. J. Analysis of triangle quality measures. *Math. Comput.*, American Mathematical Society, Boston, MA, v. 72, n. 244, p. 1817–1839, oct 2003.
- [114] PITTEWAY, N. L. K. Computer graphics research in an academic environment. In: *Datafair 73 Conference Proceedings*. London: British Computer Society, 1973. v. 2, p. 471–478.
- [115] PLAZA, A.; RIVARA, M. C. On the adjacencies of triangular meshes based on skeleton-regular partitions. *Journal of Computational and Applied Mathematics*, v. 140, p. 673–693, 2002.
- [116] PÓLYA, G. Über eine Peanosche Kurve. *Bulletin International de l'Académie des Sciences de Cracovie A*, p. 305–313, 1913.
- [117] POWELL, M. J. D.; SABIN, M. A. Piecewise quadratic approximations on triangles. *ACM Transactions on Mathematical Software*, v. 3, n. 4, p. 316–325, December 1977.
- [118] PREVIATO, E. (Ed.). *Dictionary of Applied Math for Engineers and Scientists*. New York, NY: CRC Press, 2002.
- [119] PUENTES, J. et al. Computer-assisted venous thrombosis volume quantification. *IEEE Transactions on Information Technology in Biomedicine*, v. 13, n. 2, p. 174–183, 2009.
- [120] RAJAN, V. T. Optimality of the Delaunay triangulation in  $R^d$ . In: *Proceedings of the seventh annual symposium on Computational Geometry*. New York, NY: ACM, 1991. (SCG'91), p. 357–363.
- [121] RIPPA, S. Minimal roughness property of the Delaunay triangulation. *Computer Aided Geometric Design*, Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, v. 7, n. 6, p. 489–497, 1990.
- [122] RIVARA, M. C. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *International Journal for Numerical Methods in Engineering*, v. 20, p. 745–756, 1984.

- [123] ROSENBERG, I. G.; STENGER, F. A lower bound on the angles of triangles constructed by bisecting the longest side. *Mathematics of Computation*, American Mathematical Society, v. 29, p. 390–395, 1975.
- [124] RUPPERT, J. *A new and simple algorithm for quality 2-dimensional mesh generation*. Computer Science Division, University of California at Berkeley. Technical report UCB/CSD 92/694. Berkeley, CA, 1992.
- [125] RUPPERT, J. A new and simple algorithm for quality 2-dimensional mesh generation. In: *Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*. Austin, TX: Society for Industrial and Applied Mathematics, 1993. p. 83–92.
- [126] RUPPERT, J. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *Journal of Algorithms*, Academic Press, Inc., Duluth, MN, v. 18, n. 3, p. 548–585, maio 1995.
- [127] SCHRIJVERS, O.; van BOMMEL, F.; BUCHIN, K. Delaunay triangulations on the word RAM: Towards a practical worst-case optimal algorithm. In: *28th European Workshop on Computational Geometry (EuroCG), Booklet of Abstracts*. Assisi, Italy: [s.n.], 2012. p. 13–16.
- [128] SCHRIJVERS, O.; van BOMMEL, F.; BUCHIN, K. Delaunay triangulations on the word RAM: Towards a practical worst-case optimal algorithm. In: *10th International Symposium on Voronoi Diagrams in Science and Engineering (ISVD)*. Saint Petersburg, Russia: [s.n.], 2013. p. 7–15.
- [129] SEIDEL, R. *A convex hull algorithm optimal for point sets in even dimensions*. Dissertação (Mestrado) — Department of Computer Science, University of British Columbia, Vancouver, CA, 1981. Technical report 81-14.
- [130] SHAMOS, M. I.; HOEY, D. Closest-point problems. In: *Proceedings of the 16th Annual Symposium on Foundations of Computer Science*. Washington, DC: IEEE Computer Society, 1975. p. 151–162.
- [131] SHEWCHUK, J. R. *Delaunay Refinement Mesh Generation*. 207 p. Tese (Doutorado) — School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, May 1997. Available as Technical Report CMU-CS-97-137.
- [132] SHEWCHUK, J. R. *Lecture Notes on Geometric Robustness*. Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, CA, April 2013.
- [133] SIBSON, R. Locally equiangular triangulations. *The Computer Journal*, v. 21, p. 243–245, 1978.
- [134] SIERPIŃSKI, W. Sur une nouvelle courbe continue qui remplit toute une aire plane. *Bulletin International de l'Académie des Sciences de Cracovie A*, p. 462–478, 1912.
- [135] STYNES, M. On faster convergence of the bisection method for all triangles. *Mathematics of Computation*, v. 35, n. 152, p. 1195–1201, October 1980.
- [136] TALMOR, D. *Well-Spaced Points for Numerical Methods*. Tese (Doutorado) — School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, August 1997. Available as Technical Report CMU-CS-97-164.

- [137] TAUBIN, G. Curve and surface smoothing without shrinkage. In: *Proceedings of the Fifth International Conference on Computer Vision*. Washington, DC: IEEE Computer Society, 1995. (ICCV '95, v. 5), p. 852–857.
- [138] TAUBIN, G. A signal processing approach to fair surface design. In: *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*. New York, NY: ACM, 1995. (SIGGRAPH '95), p. 351–358.
- [139] TAUBIN, G.; SHANG, T.; GOLUB, G. Optimal surface smoothing as filter desing. In: BUXTON, B.; CIPOLLA, R. (Ed.). *Proceedings of the 4th European Conference on Computer Vision*. London, UK: Springer-Verlag, 1996. (Lecture Notes in Computer Science, Volume 1), p. 283–292.
- [140] The CGAL Project. *CGAL - The Computational Geometry Algorithms Library*. 1995–2015. Disponível em: <<http://www.cgal.org/>>.
- [141] THIESSEN, A. H. Precipitation average for large area. *Monthly Weather Review*, v. 39, p. 1982–1084, 1911.
- [142] THOMPSON, J. E.; WARSI, Z. U. A.; MASTIN, C. W. *Numerical Grids Generation, foundations and applications*. Amsterdam: North Holland, 1985.
- [143] THOMPSON, J. F.; SONI, B. K.; WEATHERHILL, N. P. (Ed.). *Handbook of Grid Generation*. New York, NY: CRC Press, 1999.
- [144] TUCKER, G. E. et al. An object-oriented framework for distributed hydrologic and geomorphic modeling using triangulated irregular networks. *Computers & Geosciences*, v. 27, n. 8, p. 959 – 973, 2001.
- [145] ÜNGÖR, A. Off-centers: A new type of Steiner points for computing size-optimal quality-guaranteed Delaunay triangulations. In: FARACH-COLTON, M. (Ed.). *LATIN 2004: Theoretical Informatics*. Buenos Aires, Argentina: Springer Berlin Heidelberg, 2004. (Lecture Notes in Computer Science, v. 2976), p. 152–161.
- [146] ÜNGÖR, A. Off-centers: A new type of Steiner points for computing size-optimal quality-guaranteed Delaunay triangulations. *Computational Geometry: Theory and Applications*, v. 42, n. 2, p. 109–118, 2009.
- [147] VELHO, L.; FIGUEIREDO, L. H. de; GOMES, J. Hierarchical generalized triangle strips. *The Visual Computer*, v. 15, p. 21–35, 1999.
- [148] VELHO, L.; ZORIN, D. 4-8 subdivision. *Computer Aided Geometric Design*, v. 18, p. 397–427, 2001.
- [149] VIGNERON, A. *Computing a Delaunay Triangulation*. National of University of Singapore: INRA, 2004. Disponível em <http://www.docstoc.com/docs/46656668/Computing-a-Delaunay-triangulation>. Acessado em 10 de fevereiro de 2012.
- [150] VORONOI, G. F. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. *Journal für die Reine und Angewandte Mathematik*, v. 134, p. 198–287, 1908.
- [151] WATSON, D. F. Computing the  $n$ -dimensional delaunay tessellation with application to voronoi polytopes. *The Computer Journal*, Oxford University Press, v. 24, n. 2, p. 167–172, 1981.

- [152] WATSON, D. F.; PHILIP, G. M. Survey - systematic triangulation. *Computer Vision, Graphics and Image Processing*, v. 26, p. 217–223, 1984.
- [153] WEILLER, K. The radial-edge data structure: a topological representation for non-manifold geometric boundary modeling. In: AO, J. L. M. da E.; WOZNY, M. J.; MCLAUGHLIN, H. W. (Ed.). *Geometric Modeling for CAD Applications*. Amsterdam: Elsevier Science Publishers B. V. (North Holland), 1988. p. 3–36.
- [154] YANG, S.-W.; CHOI, Y.; JUNG, C.-K. A divide-and-conquer Delaunay triangulation algorithm with a vertex array and flip operations in two-dimensional space. *International Journal of Precision Engineering and Manufacturing*, Korean Society for Precision Engineering, v. 12, n. 3, p. 435–442, 2011.

# Índice

- algoritmo de
  - Lawson, 48
  - Bowyer-Watson, 50
  - Ruppert, 52
  - Shamos e Hoey, 61
  - de Üngör, 55
- baricentro, 23
- bola, 2, 3
  - disco, 2
- círculo
  - diametral, 52
  - vazio, 41
- condição CFL, 5
- curva, 3
  - de Hilbert, 51
  - de Pólya, 76
  - de preenchimento de espaço, 76
  - de Sierpiński, 51
  - modificada, 76
- diagrama de potência, 63
- diagrama de Voronoi, 39, 40, 60–62
  - pontos geradores, 40
  - região de Voronoi, 60
- diferenças finitas, 4
- discretização, 1, 2
- distância euclidiana, 2
- domínio, 3
  - aberto, 3
  - computacional, 1
  - fechado, 3
  - fronteira, 3
- erros de discretização, 18, 29
- espaços topológicos, 3
- estrutura de dados, 8
  - DCEL, 69
  - directed edges*, 79
  - half-edge*, 78
  - indexada, 78
  - com adjacências de Lawson, 78
  - kd-tree*, 51
  - Triangle-Segment*, 79
  - winged-edge*, 68
- fecho convexo, 13, 61
- formulação laplaciana ponderada, 24
- fronteira, 3
- geração de malhas, 3
- grafo, 2
  - aresta, 2
  - incidência, 2
  - caminho, 75
    - hamiltoniano, 75
    - simples, 76
  - de Gabriel, 59
  - vértice, 2
    - adjacência, 2
    - grau, 2
    - valência, 2
- graus de liberdade, 8
- grid*, 8
- homeomorfismo, 3
- lei dos senos, 30
- malha, 1–3
  - adaptatividade, 17
  - anisotrópica, 9
  - conectividade, 6
  - conforme, 7
  - irregular, 8, 9
  - isotrópica, 9
  - movimento de vértices, 23
  - não uniforme, 7
  - refinamento, 18
    - adaptativo de malhas, 18
  - regular, 8
    - multi-bloco, 10
    - semi-regular, 10

- simplificação, 19
- topologia, 8
- transformações topológicas, 19
- triangular, *ver* triangulação
- uniforme, 6
- manifold*, 7
- mediatriz, 21, 40
- número de Courant, 5
- politopo, 1
  - polígono, 1
  - tetraedro, 11
    - circunferência, 40
    - regular, 31
    - sliver*, 59
- ponto estacionário, 26
- pontos afim-independentes, 10
- potência de ponto, 63
- princípio de equidistribuição, 18, 23, 24
- raiz do valor quadrático médio, 32
- simplex*, 10
- suavização, 24
  - laplaciana, 24
- subdivisão de triângulos, 19
- técnica avanço de fronteira, 14
- tesselação, 40, 62
  - de Delaunay, 50, 61
  - de Dirichlet, 60
- triangulação, 10, 11
  - conjunto de pontos, 11
  - de Delaunay, 39–47, 59, 61, 62
    - condição do circuncírculo vazio, 41
    - propriedade do circuncírculo vazio, 41
    - refinamento, 52
    - restrita, 47
    - troca de arestas, 44
  - de Pitteway, 59
  - hamiltoniana, 76
  - restrita, 11
- triângulo, 11
  - baricentro, 23
  - bisseção, 19
  - circuncírculo, 20, 40
    - circuncentro, 20, 39
    - circunraio, 20, 21, 39
  - inraio, 21
  - mediana, 19
  - mediatriz, 21, 40
  - off-center*, 55
  - semiperímetro, 30
  - subdivisão, 21
    - de Powell-Sabin, 22
    - de Rivara, 20, 76
    - ternária, 22
  - valor eficaz, 32