



**MODELAGEM TRIDIMENSIONAL DE PROBLEMAS  
POTENCIAIS USANDO O MÉTODO DOS  
ELEMENTOS DE CONTORNO**

**FERNANDO MORAIS DE LOYOLA**

**DISSERTAÇÃO DE MESTRADO EM INTEGRIDADE DE MATERIAIS DA  
ENGENHARIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**FACULDADE DE TECNOLOGIA  
UNIVERSIDADE DE BRASÍLIA**

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**MODELAGEM TRIDIMENSIONAL DE PROBLEMAS  
POTENCIAIS USANDO O MÉTODO DOS  
ELEMENTOS DE CONTORNO**

**FERNANDO MORAIS DE LOYOLA**

**ORIENTADOR: PROF. DR. ÉDER LIMA DE ALBUQUERQUE, ENM/UNB**

**DISSERTAÇÃO DE MESTRADO EM INTEGRIDADE DE MATERIAIS DA  
ENGENHARIA**

**PUBLICAÇÃO PPINTEGR.DM - 47A/2017  
BRASÍLIA-DF, 23 DE FEVEREIRO DE 2017.**

**UNIVERSIDADE DE BRASÍLIA  
FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA MECÂNICA**

**MODELAGEM TRIDIMENSIONAL DE PROBLEMAS  
POTENCIAIS USANDO O MÉTODO DOS  
ELEMENTOS DE CONTORNO**

**FERNANDO MORAIS DE LOYOLA**

DISSERTAÇÃO DE Mestrado Acadêmico submetida ao Departamento de Engenharia Mecânica da Faculdade de Tecnologia da Universidade de Brasília, como parte dos requisitos necessários para a obtenção do grau de Mestre em Integridade de Materiais da Engenharia.

**APROVADA POR:**

---

Prof. Dr. Éder Lima de Albuquerque, ENM/UnB  
Orientador

---

Prof. Dra. Carla Tatiana Mota Anflor, FGA/UnB  
Examinador interno

---

Prof. Dr. Artur Antônio de Almeida Portela, ENC/UnB  
Examinador externo

**BRASÍLIA, 23 DE FEVEREIRO DE 2017.**

## **FICHA CATALOGRÁFICA**

FERNANDO MORAIS DE LOYOLA

**Modelagem tridimensional de problemas potenciais usando o método dos elementos de contorno**

**2017xv, 90p., 201x297 mm**

(PPIntegr/FT/UnB, Mestre, Integridade de Materiais da Engenharia, 2017)

Dissertação de Mestrado - Universidade de Brasília

Faculdade de Tecnologia - Departamento de Integridade de Materiais da Engenharia

## **REFERÊNCIA BIBLIOGRÁFICA**

FERNANDO MORAIS DE LOYOLA (2017) Modelagem tridimensional de problemas potenciais usando o método dos elementos de contorno. Dissertação de Mestrado em Integridade de Materiais da Engenharia, Publicação 47A/2017, Departamento de Integridade de Materiais da Engenharia, Universidade de Brasília, Brasília, DF, 90p.

## **CESSÃO DE DIREITOS**

AUTOR: FERNANDO MORAIS DE LOYOLA

TÍTULO: Modelagem tridimensional de problemas potenciais usando o método dos elementos de contorno.

GRAU: Mestre ANO: 2017

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de Mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor se reserva a outros direitos de publicação e nenhuma parte desta dissertação de Mestrado pode ser reproduzida sem a autorização por escrito do autor.

---

FERNANDO MORAIS DE LOYOLA

Correio Eletrônico: f.loyola91@gmail.com

*A Deus pela motivação e  
pela oportunidade de realizar esse sonho.*

*Aos meus pais, José Alberto e Socorro,  
ao meu irmão Fábio e à tia Rita,  
pelo apoio incondicional.*

# Agradecimentos

*Gostaria de agradecer, primeiramente, ao Prof. Dr. Éder Lima de Albuquerque, pela orientação no trabalho, pelo apoio e pela ajuda na solução dos problemas enfrentados ao longo dessa fase.*

*Agradeço também ao Prof. Dr. Artur Antônio de Almeida Portela e aos professores do Programa de Pós Graduação em Integridade de Materiais da Engenharia, que contribuíram para a minha formação.*

*Gostaria de dizer obrigado aos colegas do Mestrado, que tornaram os momentos de trabalho mais agradáveis.*

*Aos meus familiares e amigos, que sempre foram compreensivos e me apoiaram nos momentos de dificuldades.*

*Por fim, agradeço à CAPES pelo apoio financeiro.*

*Fernando Morais de Loyola*

---

## ABSTRACT

This work presents a complete approach for the analysis of heat conduction problems in solids using the boundary element method. The analysis begins with the drawing of the solid in a computer aided design program (CAD program). This drawing is then exported in the IGES (Initial Graphics Exchange Specification) format which is read in the developed program. To read the file with the .igs extension, an open source package available on the internet is used. Then, the geometry is decomposed into three-dimensional surfaces in space that are parametrized in a parametric plan using non-uniform rational B-splines, i.e., NURBS. Each of these surfaces are discretized in the parametric plane using the Delaunay triangulation. This triangulation is done using a two-dimensional triangular finite element method mesh generator which is also an open source package. The triangular mesh in the plane is then transported into space. The boundary conditions of the problem are imposed, the influence matrices of the boundary element method are assembled and a linear system is solved to compute the unknown variables in the boundary. Finally, the temperature in the surface of the solid is shown on a heat map. The program is applied in problems with high geometric complexity in order to demonstrate the developed formulation ability to analyze complex geometry problems.

Keywords: Boundary Element Method, Mesh generation, Delaunay triangulation, Non uniform rational B-splines (NURBS).

---

## RESUMO

Este trabalho apresenta uma abordagem completa para a análise de problemas de condução de calor em sólidos usando o método dos elementos de contorno. A análise começa com o desenho do sólido em programas de desenho assistido por computador (programas CAD). Em seguida, este desenho é exportado em formato IGES (*Initial Graphics exchange specification*, arquivos com extensão .igs) que então são lidos no programa desenvolvido. Para a leitura destes arquivos com extensão .igs, utiliza-se um pacote de código aberto disponível na rede internacional de computadores. Uma vez lida, a geometria é decomposta em superfícies tridimensionais, no espaço  $(x_1, x_2, x_3)$  que são parametrizadas em um espaço paramétrico  $(u, v)$  usando B-splines racionais não uniformes, ou seja, as NURBS (non uniform rational B-splines). Cada uma destas superfícies é discretizada no espaço paramétrico, ou seja, no plano  $(u, v)$  usando a triangulação de Delaunay. Esta triangulação é feita usando um gerador de malhas triangulares bi-dimensionais do método dos elementos finitos que também é um pacote com código fonte aberto. A malha triangular no plano  $(u, v)$  é então transportada para o espaço. As condições de contorno do problema são impostas, as matrizes de influência do método dos elementos de contorno são montadas e um sistema linear é resolvido para se calcular as variáveis desconhecidas no contorno. Por fim, a temperatura na superfície do sólido é mostrada em um mapa de temperaturas. O programa é aplicado em problemas de elevada complexidade geométrica para demonstrar a capacidade de análise da formulação desenvolvida.

Palavras-chave: Método dos elementos de contorno, Geração de malhas, Triangulação de Delaunay, B-splines racionais não uniformes (NURBS).



# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>1</b>
1.1	A IMPORTÂNCIA DOS MÉTODOS NUMÉRICOS .....	1
1.2	COMPARAÇÃO ENTRE MEF E MEC .....	3
1.3	TROCA DE ARQUIVOS DE DESENHO ENTRE DIFERENTES PROGRAMAS ..	4
1.4	HISTÓRIA DA GERAÇÃO DE MALHA .....	5
1.5	PRINCIPAIS CONTRIBUIÇÕES DO TRABALHO .....	6
<b>2</b>	<b>MODELAGEM GEOMÉTRICA .....</b>	<b>8</b>
2.1	INTRODUÇÃO .....	8
2.2	CURVAS DE BÉZIER.....	8
2.3	B-SPLINES .....	9
2.4	SUPERFÍCIES B-SPLINES .....	11
2.5	B-SPLINES RACIONAIS NÃO UNIFORMES - NURBS.....	11
2.6	SUPERFÍCIES NURBS .....	12
2.7	FORMATO DO ARQUIVO IGES .....	13
2.7.1	ENTIDADES DO ARQUIVO .....	14
2.7.2	EXEMPLO DE ARQUIVO IGES .....	18
<b>3</b>	<b>FORMULAÇÃO TRIDIMENSIONAL DO MÉTODO DOS ELEMENTOS DE CON-</b>	
	<b>TORNO .....</b>	<b>20</b>
3.1	INTRODUÇÃO .....	20
3.1.1	EQUAÇÃO DE LAPLACE PARA PROBLEMAS TRI-DIMENSIONAIS .....	20
3.2	SOLUÇÃO FUNDAMENTAL .....	21
3.3	EQUAÇÃO INTEGRAL DE CONTORNO .....	22
3.4	DISCRETIZAÇÃO DAS EQUAÇÕES INTEGRAIS DE CONTORNO .....	24
3.5	ELEMENTOS TRIANGULARES.....	27
3.6	ELEMENTOS DE CONTORNO TRIANGULARES CONSTANTES .....	29
<b>4</b>	<b>GERAÇÃO DE MALHA .....</b>	<b>31</b>
4.1	INTRODUÇÃO .....	31
4.2	MALHAS ESTRUTURADAS E NÃO-ESTRUTURADAS .....	31
4.2.1	MALHAS NÃO-ESTRUTURADAS EM DOMÍNIOS PLANOS.....	33
4.3	PROPRIEDADES DAS MALHAS .....	34

4.4	QUALIDADE DA MALHA .....	35
4.4.1	O PROBLEMA DA MELHORIA DA QUALIDADE DA MALHA .....	35
4.5	GERAÇÃO DE MALHA POR TRIANGULAÇÃO DE DELAUNAY .....	36
4.5.1	INTRODUÇÃO .....	36
4.5.2	PROPRIEDADES DA PARTIÇÃO DE VORONOÏ.....	39
4.5.3	PROPRIEDADES DA TRIANGULAÇÃO DE DELAUNAY .....	40
4.5.4	O CONTORNO COMO RESTRIÇÃO DA TRIANGULAÇÃO DE DELAUNAY ...	40
4.5.5	CAVIDADES .....	41
4.5.6	INSERÇÃO DE PONTOS.....	43
4.5.7	RECUPERAÇÃO DA FRONTEIRA DO DOMÍNIO .....	44
4.5.8	GERAÇÃO DE MALHA SOBRE SUPERFÍCIES CURVAS.....	46
<b>5</b>	<b>DESCRIÇÃO DOS PACOTES USADOS .....</b>	<b>53</b>
5.1	INTRODUÇÃO .....	53
5.2	IGES TOOLBOX.....	54
5.3	MESH2D TOOLBOX .....	55
5.3.1	TRANSFERINDO OS PONTOS DA MALHA GERADA NO PLANO 2D PARA O ESPAÇO 3D .....	58
<b>6</b>	<b>RESULTADOS.....</b>	<b>59</b>
6.1	MANCAL .....	59
6.2	PLACA COM FURO .....	65
<b>7</b>	<b>CONCLUSÃO E PROPOSTA PARA TRABALHOS FUTUROS .....</b>	<b>69</b>
7.1	CONCLUSÃO .....	69
7.2	PROPOSTA PARA TRABALHOS FUTUROS .....	70
	<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>71</b>
<b>A</b>	<b>ROTINA PARA TRANSFERÊNCIA DA MALHA GERADA NO PLANO 2D PARA O ESPAÇO 3D.....</b>	<b>75</b>

# LISTA DE FIGURAS

1.1	Solução de problemas de engenharia. ....	2
1.2	Discretização com MEF (esq.) e MEC (dir.). Fonte: Gaul, Kögl e Wagner (2003) .....	4
2.1	Curva de Bézier com 4 pontos de controle (P1, P2, P3 e P4).....	9
2.2	Funções B-spline de ordem 0, 1 e 2 para o vetor de nós $U = \{0, 1, 2, 3, 4, \dots\}$ Fonte: Gonçalves (2016).....	10
2.3	Função aproximada com NURBS e diferentes valores de peso para o terceiro ponto de controle. ....	12
2.4	Parte frontal da superfície com vetores normais. ....	13
2.5	Exemplo de superfície aparada por uma curva. Fonte: Systemes (2017a).....	16
2.6	Processo para aparar uma superfície. Fonte: Systemes (2017a). ....	16
2.7	Parte de arquivo .igs aberto em um editor de textos.....	19
2.8	Imagem renderizada no SolidWorks. Fonte: Balan (2016) .....	19
3.1	Sólido com volume $V$ e superfície $S$ . ....	21
3.2	Superfície dividida em pequenas partes.....	25
3.3	Contorno tridimensional dividido em: (a) elementos constantes; (b) elementos lineares; e (c) elementos quadráticos.....	26
3.4	Elemento triangular linear: (a) no espaço real $(x_1, x_2, x_3)$ e (b) no espaço paramétrico $(\xi, \eta)$ . ....	27
4.1	Elementos não-conformes .....	32
4.2	Numeração dos elementos. Fonte: Aguirre (2017).....	32
4.3	Malha estruturada. Fonte: Maliska (2004). ....	33
4.4	Malha não estruturada. Fonte: Maliska (2004).....	34
4.5	Diagrama (ou partição) de Voronoï de 10 pontos. ....	37
4.6	Triangulação de Delaunay de 10 pontos. ....	39
4.7	Sobreposição da Triangulação de Delaunay e da Partição de Voronoï de 10 pontos.....	39
4.8	Inserção de ponto $P$ . Fonte: Oliveira (1995). ....	42
4.9	Cavidade generalizada (a) e componente principal (b). Fonte: Oliveira (1995). ....	43
4.10	Domínio inicial (a) e perda da fronteira (b). Fonte: Oliveira (1995). ....	45

4.11	Recuperação de uma aresta ausente em 2D. Fonte: Lo (2015) (modificado). ..	48
4.12	Malhas triangulares geradas diretamente no espaço. Fonte: Lo (2015). .....	49
4.13	Malha quadrilateral gerada diretamente no espaço. Fonte: Lo (2015). .....	50
4.14	Superfície curva $S$ produzida por um processo de mapeamento paramétrico. Fonte: Lo (2015).....	50
4.15	Malha estruturada paramétrica. Fonte: Lo (2015).....	51
4.16	Malha não-estruturada paramétrica. Fonte: Lo (2015). .....	51
4.17	Malha anisotrópica paramétrica. Fonte: Lo (2015). .....	52
5.1	Fluxograma de processos do trabalho. ....	54
5.2	Exemplo de arquivo IGES lido no Matlab pelo pacote. ....	55
5.3	Exemplo 1 do pacote Mesh2D. ....	56
5.4	Exemplo 2 do pacote Mesh2D. ....	57
5.5	Malha gerada (a) pela função 5.3 (b). ....	57
6.1	Mancal e superfícies. ....	59
6.2	Geração de pontos do contorno da superfície 2. ....	61
6.3	Geração de pontos internos da superfície 2. ....	62
6.4	Transferência da malha do plano 2D para o espaço 3D da superfície 2. ....	62
6.5	Mancal com as normais corrigidas. ....	63
6.6	Detalhe para visualização da malha. ....	63
6.7	Temperaturas no mancal - mapa de cores. ....	64
6.8	Placa com furo e superfícies. ....	65
6.9	Placa com as normais corrigidas. ....	65
6.10	Geração de pontos do contorno da superfície 6 .....	66
6.11	Geração de pontos internos da superfície 6 .....	67
6.12	Transferência da malha do plano 2D para o espaço 3D da superfície 6. ....	67
6.13	Temperaturas na placa com furo. ....	68

# LISTA DE TABELAS

2.1	Tipos de entidades de arquivo IGES .....	15
2.2	Parâmetros para a entidade 143 .....	17
2.3	ParameterData.....	18
4.1	Definição de métricas de qualidade de malha .....	36
5.1	Dados de entrada .....	54
5.2	Dados de saída .....	55
6.1	Entidades no mancal. ....	60
6.2	Iterações e convergência do mancal. ....	60
6.3	Entidades no placa com furo. ....	66
6.4	Iterações e convergência da placa com furo. ....	66
6.5	Quantidades de elementos dos exemplos. ....	68

# Capítulo 1

## Introdução

### 1.1 A importância dos métodos numéricos

As variadas áreas de engenharia buscam encontrar resultados suficientemente precisos para modelar os problemas reais. Normalmente procuram soluções analíticas, que são exatas, mas difíceis de encontrar para problemas mais complexos. A definição de uma solução analítica requer simplificações, visando a diminuir a complexidade das equações que regem o problema, o que pode acabar gerando um modelo não condizente com o real.

Diante desse escopo, cresceu a importância dos métodos numéricos, que passaram a ser mais estudados para encontrar soluções aproximadas dos problemas, mesmo os complexos. Os métodos numéricos possuem a vantagem de não demandarem a consideração de muitas dessas simplificações, permitindo análises mais realistas. Isso ocorre porque os métodos numéricos usam um modelo discreto – ao contrário da análise analítica, que usa um modelo contínuo para resolver o problema. De acordo com Sanches e Furlan (2007), um método numérico é um algoritmo composto por um número finito de operações matemáticas. Já de acordo com Gaul, Kögl e Wagner (2003), a resolução de problemas de engenharia pode ser feita de acordo com a Figura 1.1.

Para resolução de problemas de engenharia, destacam-se três modelos: Método das Diferenças Finitas (MDF); Método dos Elementos Finitos (MEF); Método dos Elementos de Contorno (MEC). Além destes, ressalta-se o crescente uso dos métodos sem malha.

O Método das Diferenças Finitas (MDF) ou Finite Difference Method (FDM) trabalha com resolução de equações diferenciais, com base na aproximação de derivadas por diferenças finitas. As aproximações são obtidas a partir da série de Taylor. Como pode ser visto em Deus (2010), alguns exemplos de áreas de aplicação do MDF são: equilíbrio estrutural de vigas, placas e cascas; estabilidade elástica de colunas; fluxo em meio poroso; linha de corrente em escoamento. O Método dos Elementos Finitos (MEF) ou Finite Element Method (FEM) também é uma forma de resolução numérica de um sistema de equações diferenciais parciais. A formulação do MEF requer a existência de uma equação integral, de modo que

seja possível substituir a integral sobre um domínio complexo por um somatório de integrais sobre domínios de geometria simples (AZEVEDO, 2003). Ressalta-se que a formulação do MEF para aplicação à análise de estruturas é baseada no Princípio dos Trabalhos Virtuais (PTV). Uma característica que todos os métodos têm em comum é que resolvem sistemas de equações algébricas no lugar de um sistema de equações diferenciais.

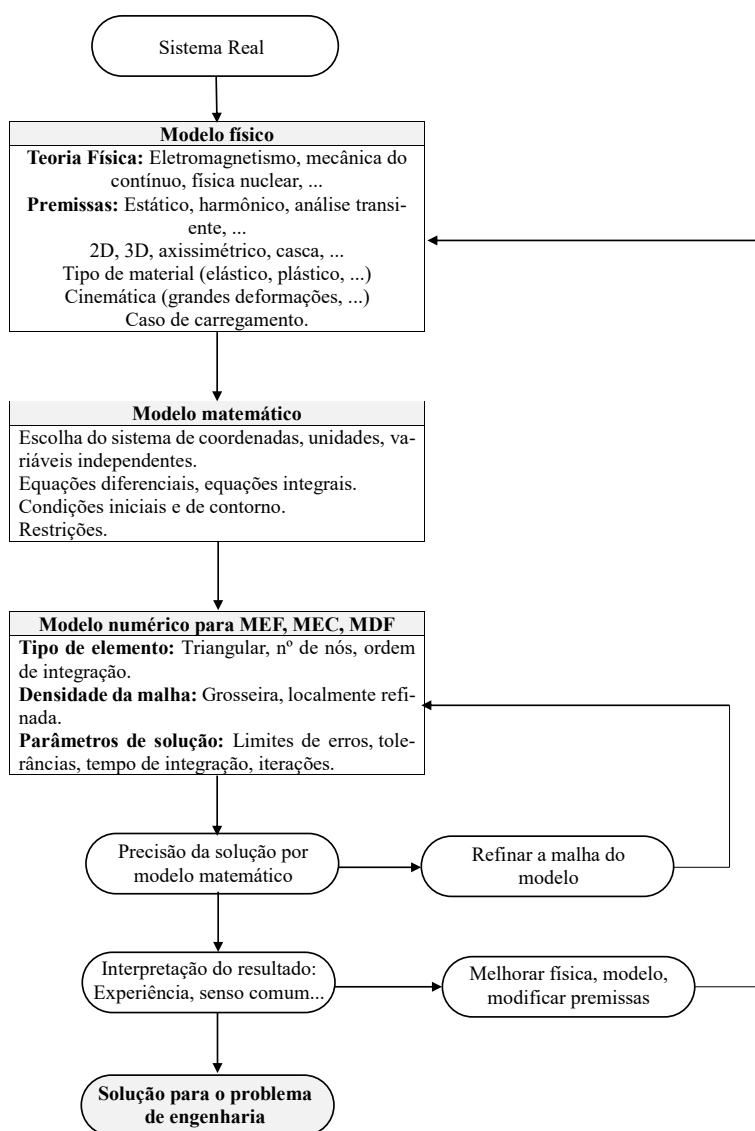


Figura 1.1: Solução de problemas de engenharia.

O MEF, segundo Kane (1994) apud Júnior (2014), transforma uma equação diferencial numa equação integral. Como pode ser visto em Cheng e Cheng (2005) apud Reboredo (2015), até 2005 o MEF possui o maior número de publicações registradas na plataforma

Web of Science, com 65 mil entradas. Isso torna o MEF o método numérico mais popular.

Já o Método dos Elementos de Contorno (MEC) ou Boundary Element Method (BEM), o qual será estudado de forma mais aprofundada nesse trabalho, é um método integral. Ele consiste em obter equações integrais apenas com as informações do contorno do problema, sem precisar discretizar o domínio. Essa característica do MEC faz com que ele diminua a dimensão do problema em uma ordem (apenas uma superfície é discretizada, não um volume), obtendo vantagem sobre os outros dois métodos citados. Logo, existe uma diminuição na quantidade de dados de entrada, no tempo de processamento e no armazenamento das informações processadas, propiciando uma menor quantidade de operações aritméticas (ALBUQUERQUE, 2012).

Ainda, segundo Azevedo (2007), o MEC dá soluções mais precisas, tendo em vista que utiliza soluções de equações integrais sem aproximação no domínio. As informações do domínio são obtidas a partir das variáveis do contorno, pois os valores das variáveis nos pontos internos de um problema podem ser escritos como funções das variáveis externas presentes no contorno.

Finalmente, a característica que talvez seja a mais interessante no MEC em relação aos outros, é que facilita os problemas que requerem alteração constante na malha para aperfeiçoar a solução, o caso da propagação de trincas na Mecânica da Fratura e problemas de fronteiras móveis.

Apesar das vantagens, o MEC não é muito utilizado comercialmente por ser limitado a problemas de pequena e média escala pois o mesmo gera matrizes cheias e não simétricas. Este fato torna o uso da memória e o tempo computacional extremamente altos para problemas de larga escala - problemas com mais de um milhão de graus de liberdade. Entretanto, os métodos dos Elementos de Contorno Rápidos são capazes de reduzir o uso de memória e o tempo computacional, segundo Campos (2016) e Liu (2009), tornando o MEC competitivo em problemas de larga escala.

## **1.2 Comparação entre MEF e MEC**

O MEF e o MEC são os métodos mais versáteis para análises de engenharia, conforme Gaul, Kögl e Wagner (2003), e por isso é fundamental saber as respectivas vantagens e desvantagens de ambos. Este trabalho tem como foco o MEC e não irá descrever de forma aprofundada o MEF, mas é importante saber as diferenças entre eles.

Como dito anteriormente, uma das maiores vantagens do MEC é no momento da discretização, e essa é uma notória diferença entre os métodos. No MEF, todo o domínio deve ser discretizado, ao contrário do MEC no qual apenas o contorno deve ser discretizado, como visto na Figura 1.2. A depender da complexidade da geometria e dos casos de carregamento, isso pode levar a importantes ganhos de tempo na criação e modificação da malha (GAUL;



KöGL; WAGNER, 2003).

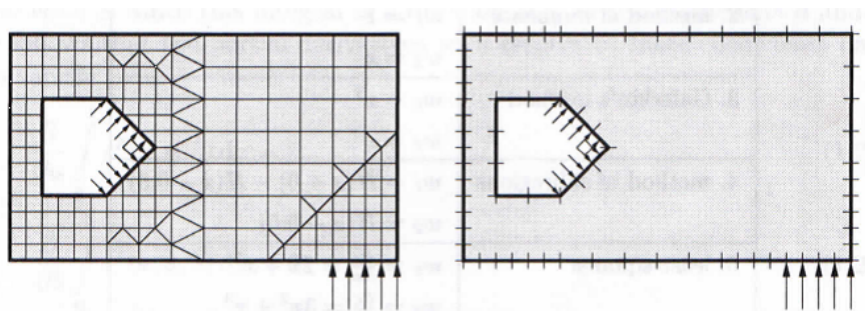


Figura 1.2: Discretização com MEF (esq.) e MEC (dir.). Fonte: Gaul, Kögl e Wagner (2003)

Conforme será mostrado ao longo deste trabalho, algoritmos de geração de malhas bidimensionais do MEF podem ser usados com sucesso e com poucas modificações para a geração de malhas tridimensionais para o MEC.

### 1.3 Troca de arquivos de desenho entre diferentes programas

Dentre as várias análises numéricas, as análises de problemas tridimensionais são as que apresentam os maiores desafios, principalmente devido a geometria do problema que pode ter elevada complexidade. Neste trabalho utilizar-se-á a conveniência de importar arquivos CAD para o Matlab/Octave e em seguida fazer uma análise com elementos de contorno. O programa desenvolvido fará análise de temperatura e fluxo de calor em sólidos. Para isso, será utilizado o pacote IGES toolbox (BERGSTRÖM, 2017), que, como o nome sugere, importa arquivos no formato IGES para o Matlab/Octave. Esse processo agiliza o trabalho do engenheiro, pois este consegue utilizar um programa CAD de sua preferência para gerar a geometria, exportá-la em formato IGES e, em seguida, lê-la e gerar a malha em um programa de elementos de contorno.

O formato de arquivo IGES (Initial Graphics Exchange Specification) é neutro e permite o intercâmbio de informações gráficas vetoriais 2D ou 3D entre sistemas CAD/CAM/CAE. Usando IGES, o operador pode trocar dados de modelos em forma de diagramas de circuito, “wireframes”, superfícies livres ou representações de sólidos e atualmente é um dos formatos mais populares para intercâmbio de arquivos. O IGES é suportado por softwares comerciais como o Solidworks (SYSTEMES, 2017b) e softwares livres como o Freecad (HAVRE, 2016).

O IGES foi uma iniciativa da Força Aérea dos Estados Unidos, por meio do projeto Manufatura Integrada Auxiliada por Computador (Integrated Computer Aided Manufacturing), entre os anos de 1976 e 1984. O foco era desenvolver procedimentos, processos e softwares CAD/CAM que integrassem todas as operações necessárias à indústria aeroespacial e que

reduzissem os custos. Na época, os dados gerados por diferentes sistemas CAD eram incompatíveis entre si, dificultando o desenvolvimento de novas tecnologias. O governo americano, em parceria com grandes empresas como Boeing, General Eletric, Xerox, Computervision, Applicon e outros, ficaram responsáveis por desenvolver e testar o programa.

Desde 1980, com a criação de uma norma ANSI, o IGES passou a ser utilizado nas indústrias automotiva, aeroespacial, naval e bélica. Apesar da grande utilização, o interesse pelo uso do IGES caiu após o lançamento de outro formato, o STEP, no ano de 1994. A norma ISO 10303 descreve o formato STEP, que convive com o IGES atualmente.

O arquivo IGES tem como base as funções NURBS - Non Uniform Rational Basis Spline -, que são modelos matemáticos utilizados para gerar e representar as curvas e superfícies de geometrias complexas. Conforme Campos (2016), isso possibilita reproduzir de maneira exata entidades que seriam apenas aproximadas por funções polinomiais, incluindo seções cônicas circulares.

## 1.4 História da Geração de Malha

A pesquisa e o interesse pela geração de malha começou formalmente por volta do início da década de 1970, segundo Mackerle (2001). Desde então, o processo evoluiu junto com o desenvolvimento do poder de processamento dos computadores e do próprio método numérico. As malhas irregulares geradas por computador tornaram-se populares por dois motivos, segundo Lo (2015): (i) elas permitem que pontos sejam situados em contornos curvos de domínios irregulares; (ii) elas permitem que pontos sejam distribuídos no interior do domínio com espaçamento variável.

Uma tentativa inicial de mapear um domínio regular de referência em um domínio físico computacional geometricamente irregular foi feita com a transformação de coordenadas, com a possibilidade de uma transição suave entre os tamanhos de elementos. O Método das Diferenças Finitas (MDF) também podia ser aplicado à malhas construídas em computador baseadas na transformação de coordenadas. Com isso, as malhas podiam ser suavizadas de tal forma que os pontos interiores deviam ter suas posições determinadas pela média das coordenadas dos vizinhos.

A interpolação de elementos finitos como meio de geração de malha foi apresentado por Zienkiewicz e Phillips (1971), trabalho no qual um domínio curvo é representado por um super-elemento, que poderia ser dividido em elementos menores, seguindo a coordenada de referência dos elementos. A função de interpolação desenvolvida para refinamentos locais para minimizar a energia do sistema é relacionada ao processo de refinamento que é utilizado atualmente. A ideia intuitiva e extremamente útil como meio de geração de malha de decompor um domínio em sub-regiões mais simples foi desenvolvida há pouco tempo, conforme Lo (2015), para a geração de malhas estruturadas. Outra ideia foi a de remover pontos de uma malha refinada preexistente, gerada por transformação, a fim de gerar malhas com

elementos de tamanhos variados.

Antes da triangulação de Delaunay ser amplamente utilizada, as malhas eram construídas unindo-se os pontos gerados aleatoriamente por meio de métodos heurísticos de conexão. O método proposto por Park e Washam (1979) talvez tenha sido o predecessor dos métodos mais sofisticados utilizados atualmente. A importância da gradação das malhas foi devidamente reconhecida e vários métodos de geração de malha foram desenvolvidos com base na equação de Poisson com um termo fonte, no mapeamento e na remoção de pontos e geração de pontos aleatórios com diferentes densidades. Além disso, os problemas tridimensionais também receberam a devida atenção.

Paralelamente, houve progressos substanciais em técnicas auxiliares à geração de malha, como: nos esquemas para renumeração de nós a fim de reduzir a matriz na resolução de um sistema linear de equações; na prescrição de densidades diferentes de pontos em diferentes regiões do domínio e do contorno; na entrada de dados para a geração de malha. Esta evoluiu e agora é interativa, podendo ser utilizada como um programa CAD.

O trabalho de Thacker (1980) foi o início do uso das malhas não-estruturadas no começo dos anos 1980, principalmente pelo desenvolvimento de três métodos populares - triangulação de Delaunay, avanço de fronteira e decomposição *octree*. A base teórica da triangulação de Delaunay foi estabelecida por Dirichlet (1850), Voronoi e Delaunay, entretanto, um algoritmo de inserção de pontos foi desenvolvido apenas em 1981 por Bowyer (1981) e Watson (1981).

## 1.5 Principais contribuições do trabalho

Este trabalho apresenta uma abordagem do MEC para a análise de problemas tridimensionais, desde a construção de um modelo tridimensional em programas CAD, a exportação em formato de troca de arquivo de desenho, a geração da malha superficial triangular de superfície, a verificação das normais, a aplicação das condições de contorno, a análise do MEC e o pós-processamento.

A modelagem computacional da geometria é feita em quaisquer softwares CAD como, por exemplo, o FreeCad (HAVRE, 2016; COLLETTE; FALCK, 2012) que tem código aberto ou no SolidWorks (SYSTEMES, 2017b) que é um programa comercial. Estes modelos geométricos são exportados em arquivos em formato de troca IGES (arquivos com extensão .igs). Os arquivos IGES produzidos pelos programas CAD são lidos no programa desenvolvido neste trabalho. O sólido é decomposto em superfícies NURBS. No espaço paramétrico, estas superfícies são representadas por figuras planas. Estas figuras planas são, então, discretizadas, gerando uma malha triangular plana. Em seguida, usando as funções NURBS, estes triângulos são mapeados no espaço, produzindo superfícies discretizadas em triângulos que aproximam as superfícies do sólido.

A principal contribuição deste trabalho é a utilização de um gerador de malhas bidimensionais do método dos elementos finitos para geração de malhas tridimensionais do método dos elementos de contorno.

Destaca-se que os pacotes - ou *toolboxes* - utilizados para a leitura do arquivo de desenho e geração de malha são códigos abertos e podem ser obtidos sem qualquer custo. Para a leitura de arquivo IGES foi utilizado o pacote IGES Toolbox (BERGSTRÖM, 2017) enquanto que para a geração da malha foi utilizado o pacote Mesh2D (ENGWIRDA, 2017).

Apesar de a maioria da implementação ter sido feita em Matlab/Octave, há ainda rotinas em C, que são as que apresentam alto custo computacional, como a montagem das matrizes de influência. Estas foram compiladas como *oct-files*, gerando um objeto carregável dinamicamente no ambiente Octave.

O programa foi aplicado a problemas de condução de calor com alta complexidade geométrica, modelados em CAD. O programa mostrou-se robusto e eficiente, podendo ser facilmente estendido para outras formulações como, por exemplo, problemas elásticos tridimensionais.

# Capítulo 2

## Modelagem geométrica

### 2.1 Introdução

Este capítulo faz uma breve descrição das curvas de Bézier e das curvas e superfícies NURBS. Em seguida exemplifica como é o formato do arquivo IGES e explica brevemente quais são suas principais entidades. Os trabalhos de Rogers (2000), Piegl e Tiller (2012), Farin (1995) e Farin, Hoschek e Kim (2002) são boas fontes para compreender as NURBS e o desenho geométrico no computador. Já os trabalhos de Smith (1983), Smith e Wellington (1986) e Ito e Nakahashi (2002) falam sobre IGES, sendo que o último aborda também a geração de malha triangular diretamente sobre a superfície 3D.

### 2.2 Curvas de Bézier

As curvas de Bezier são utilizadas em uma vasta quantidade de softwares gráficos, desde que foram criadas por Pierre Bézier em 1962, quando era funcionário da Renault. Dado  $n + 1$  pontos de controle, uma curva de Bézier de grau  $n$  é escrita como:

$$P(t) = \sum_{i=0}^n B_i J_{n,i}(t), \quad 0 \leq t \leq 1 \quad (2.1)$$

onde  $B_i$  são os pontos de controle e  $J_{n,i}$  é a base de Bernstein, definida como:

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i} \quad (2.2)$$

O polígono de controle é definido como uma linhas que ligam os pontos de controle. Na Figura 2.1, o polígono de controle é representado em preto.

Em uma curva de Bézier observa-se que:

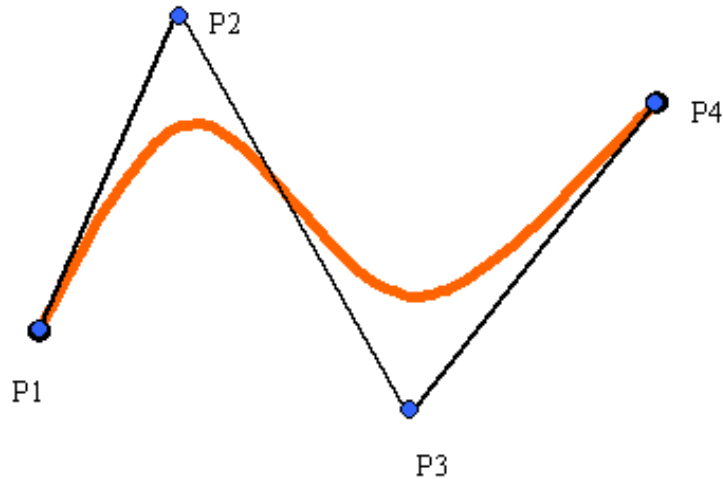


Figura 2.1: Curva de Bézier com 4 pontos de controle (P1, P2, P3 e P4)

- Para uma curva de ordem  $n$ , necessitam-se  $n+1$  pontos de controle.
- O início e final da curva coincidem, respectivamente, com o primeiro e o último pontos de controle.
- Os vetores tangentes, no final e no início da curva, coincidem com os segmentos do polígono de controle.
- A curva se mantém dentro da envoltória definida pelos pontos de controle.

## 2.3 B-splines

As predecessoras das B-splines são os polinômios de Bernstein que são definidos pela Equação (2.2). Os polinômios de Bernstein de ordem 1 são idênticos às funções de Lagrange. Nota-se que as B-splines são generalizações das curvas de Bézier. Aquelas são formadas por uma ou mais destas, com continuidade entre os segmentos. A descrição matemática de uma B-spline é dada por:

$$P(t) = \sum_{i=1}^{n+1} B_i N_{i,k}(t), \quad t_{min} \leq t \leq t_{max}, \quad 2 \leq k \leq n+1 \quad (2.3)$$

onde  $B_i$  são os pontos de controle e  $N_{i,k}$  são as funções de base de ordem  $k$  e grau  $k-1$ .

Para definir as B-splines, primeiramente define-se um vetor nó.

$$U = (\xi_0, \xi_1, \dots, \xi_{n+p+1}) \quad (2.4)$$

onde  $\xi_i \in \mathbb{R}$  é o  $i$ -ésimo nó,  $i$  é o índice do nó,  $i = 1, 2, \dots, n+p+1$ ,  $p$  é a ordem do polinômio e  $n$  é o número de funções utilizadas para construir a B-spline.

Como exemplo, utiliza-se o vetor dado em (2.5) para gerar a Figura 2.2.

$$\mathbf{U} = (\xi_0 = 1, \xi_2 = 2, \xi_3 = 3, \dots) \quad (2.5)$$

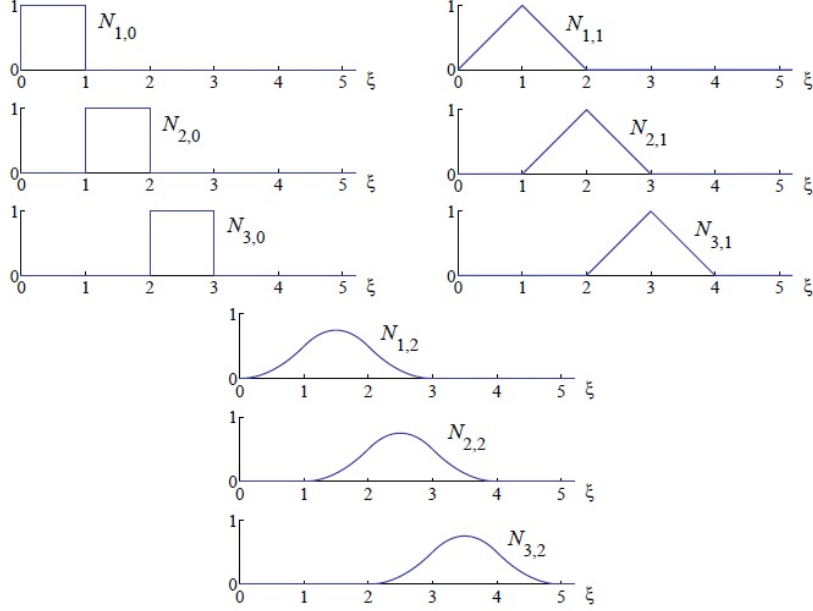


Figura 2.2: Funções B-spline de ordem 0, 1 e 2 para o vetor de nós  $\mathbf{U} = \{0, 1, 2, 3, 4, \dots\}$   
Fonte: Gonçalves (2016).

Definem-se as entradas no vetor como *nós* e o intervalo entre *nós* como *vão*. As funções são calculadas recursivamente como na Equação (2.7), chamada de *Fórmula Recursiva de Cox-de-Boor*.

$$N_{i,1}(t) = \begin{cases} 1 & \text{se } u_i \leq t < u_{i+1} \\ 0 & \text{restante do intervalo} \end{cases} \quad (2.6)$$

Bases de ordens superiores são definidas tomando as de ordens inferiores, da seguinte forma:

$$N_{i,k}(t) = \frac{(t - u_i)N_{i,k-1}(t)}{u_{i+k-1} - u_i} + \frac{(u_{i+k} - t)N_{i+1,k-1}(t)}{u_{i+k} - u_{i+1}} \quad (2.7)$$

onde  $u$  são os valores do vetor de nós  $\mathbf{U}$ .

Nota-se que as funções de ordens superiores são combinações lineares das de ordens inferiores. As coordenadas no *vetor de nós* podem ser distribuídas não-uniformemente, situação em que são chamadas de *funções B-spline não-uniformes*. O vetor de nós também pode ser utilizado para controlar a continuidade das funções. Segundo Beer (2015), a continuidade da função pode ser controlada pela repetição de valores no vetor de nós. A repetição de valores  $p+1$  vezes significa continuidade  $C^{-1}$  (ou seja, descontinuidade) e uma repetição  $p$  vezes significa continuidade  $C^0$  (ou seja, continuidade dos valores das funções, não das

tangentes).

O número de intervalos do vetor  $U$  define o número de segmentos na curva, os quais sofrem influência de  $k$  pontos de controle. Como pode ser visto no trabalho de Gonçalves (2016), cada vez que o parâmetro  $t$  entra em um novo segmento, um novo ponto de controle é ativado e o antigo é desativado. O número de nós,  $m$ , o de pontos de controle,  $n$ , e a ordem da curva,  $k$ , relacionam-se pela seguinte relação:

$$m = k + n + 1 \quad (2.8)$$

Quando nós consecutivos apresentam o mesmo valor, ocorre a chamada *multiplicidade de nós*. Vale ressaltar que esses casos devem ser analisados com atenção, tendo em vista que o segmento entre os nós com o mesmo valor teria comprimento nulo. Como consequência disso, dois pontos de controle seriam ativados e outros dois seriam desativados simultaneamente, o que pode alterar a continuidade da curva. Destaca-se ainda que a multiplicidade de um nó não pode ser maior que a ordem da curva, dado que caso a multiplicidade fosse maior, a curva seria descontínua e pontos de controle seriam perdidos.

## 2.4 Superfícies B-splines

As superfícies B-splines são descritas por meio de um produto tensorial :

$$N_{i,j}^{p,q} = N_{i,p}(u) \cdot N_{j,q}(v) \quad (2.9)$$

onde  $N_{i,p}(u)$  já foi descrito anteriormente e  $N_{j,q}(v)$  é uma B-spline de ordem  $q$  na direção  $v$ .

## 2.5 B-Splines racionais não uniformes - NURBS

As B-Splines racionais não uniformes (do inglês NURBS: *Non Uniform Rational Basis Spline*) são baseadas nas B-splines, mas com a diferença de controlar a curva usando pesos. A função torna-se:

$$R_{i,p}(u) = \frac{N_{i,p}(u)w_i}{\sum_{j=0}^I N_{j,p}(u)w_j} \quad (2.10)$$

onde  $N_{i,p}(u)$  são as B-splines e  $w_i$  são os pesos. Para conservar a *partição da unidade*, uma divisão pela soma das B-splines multiplicado pelos pesos é requerida.

A função  $R_{i,p}(u)$  deixa de ser polinomial e passa então a ser *racional* e a chamar-se *non-uniform rational B-spline*.

Para ilustrar a influência dos pesos nas NURBS, a Figura 2.3 mostra uma função aproxi-



mada com NURBS com diferentes valores de peso para o terceiro ponto de controle.

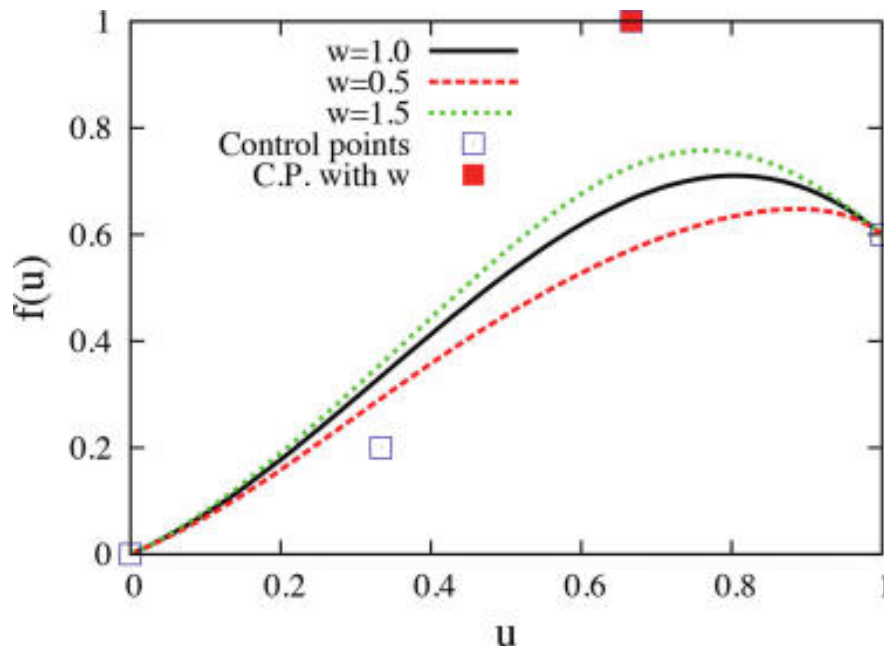


Figura 2.3: Função aproximada com NURBS e diferentes valores de peso para o terceiro ponto de controle.

## 2.6 Superfícies NURBS

Superfícies NURBS são geradas a partir da multiplicação de duas curvas B-splines por pesos e garantindo a condição de partição da unidade. Para o caso específico nos quais todos os pesos apresentam o mesmo valor, a NURBS será uma B-spline.

$$R_{i,j}^{p,q} = \frac{N_{i,p}(u) \cdot N_{j,q}(v) \cdot w_{i,j}}{W} \quad (2.11)$$

com

$$W = \sum_{j=0}^J \sum_{i=0}^I N_{i,p}(u) \cdot N_{j,q}(v) \cdot w_{i,j} \quad (2.12)$$

onde  $N_{i,p}(u)$  e  $N_{j,q}(v)$  são funções B-spline com coordenadas locais  $u$  ou  $v$  de ordem  $p$  ou  $q$  e  $w_{i,j}$  são os pesos. O que difere as NURBS das B-splines é que a amplitude das funções de base são determinadas pelos pesos.

Quando uma superfície NURBS é gerada, há duas faces: a frontal e a posterior. Os vetores normais à superfície apontam para fora da frontal, como na Figura 2.4. Neste trabalho, em vez de exibir os vetores, utiliza-se a cor vermelha para representar a parte frontal da superfície e a cor azul para representar a posterior.

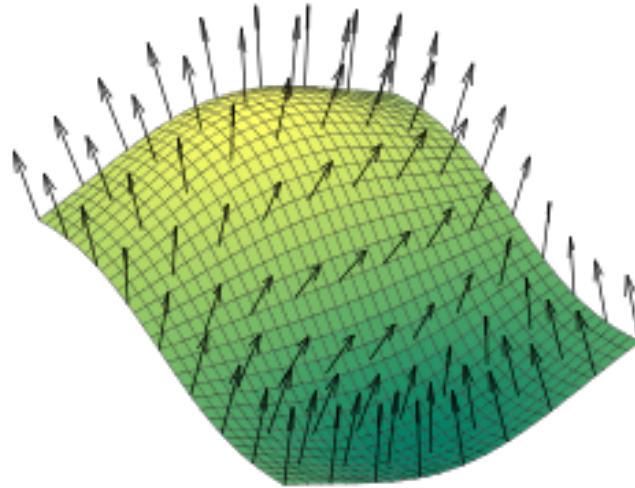


Figura 2.4: Parte frontal da superfície com vetores normais.

## 2.7 Formato do arquivo IGES

Os arquivos IGES tem uma herança da era dos cartões perfurados e são compostos por linhas de 80 caracteres ASCII, fazendo com que qualquer editor de texto seja capaz de lê-los. As variáveis do tipo “*string*” são representadas com o número de caracteres, seguido pela letra “H”, seguido pela string propriamente dita.

Conforme Gonçalves (2016), uma curva NURBS é especificada em IGES por:

- Grau,  $p$ , e o número de pontos de controle,  $n+1$ ;
- Pontos de controle,  $P_i$ , e os pesos,  $w_i$ ;
- Um vetor nodal,  $U$ , contendo  $m+1 = n+p+2$  nós;
- Parâmetros inicial e final de valores,  $s_0$  e  $s_1$ ;

Nota-se na Figura 2.7 que o arquivo IGES é um arquivo de texto, com informações divididas por setores. A primeira parte revela as informações gerais do arquivo, como diretório, data, autor, entre outras. Em seguida, é feita a definição das cores das superfícies. Após isso, a definição das superfícies e curvas, com seus respectivos dados NURBS. Cabe destacar que apenas a segunda parte é relevante para a simulação com o MEC, pois é onde estão localizadas as informações do modelo.

Este tipo de arquivo possui dois tipos de entidades - as geométricas e as não geométricas. Aquelas são os pontos, as linhas, os perfis, as superfícies, entre outros. Já estas são dados de cores ou textos explicativos, por exemplo.

Por padrão, o IGES possui 80 colunas de caracteres, nas quais as de número 1 a 64 contêm informações sobre o objeto. Os dados descritos nesse formato são separados por vírgula e o ponto-e-vírgula faz o papel de finalizar os dados.

## 2.7.1 Entidades do arquivo

Cada entidade do IGES corresponde a um número que o identifica. A Tabela 2.1 exemplifica as entidades, com dados retirados de Systemes (2017b).

### Detalhes sobre alguns tipos de entidades

Informações detalhadas sobre as entidades IGES podem ser encontradas em USPRO (1996). É de suma importância discorrer acerca das superfícies aparadas (*trimmed surfaces*), a entidade de tipo 144. Para isso, entretanto, vale apresentar antes a entidade 143 - a superfície confinada (*bounded surface*) - já que esta é utilizada para representar aquela.

### Superfície confinada ou *bounded surface* - entidade 143

Conforme dito no parágrafo anterior, as superfícies confinadas são utilizadas para representar as superfícies aparadas. Tanto a superfície quanto a curva utilizada para apará-la são assumidas como sendo parametricamente representadas. Há dois tipos de transferência suportados pela superfície confinada: tipo 0, que representa a superfície e os contornos; tipo 1, que representa a superfície, seus contornos e a curva paramétrica associada para aparar cada contorno.

A fim de representar uma superfície confinada, utilizam-se várias entidades. São elas: entidade 143 (superfície confinada); entidade 141 (contorno); entidades para representação paramétrica de superfícies não-aparadas; entidades para representação paramétrica de curvas. Os parâmetros utilizados estão na Tabela 2.2.

### Superfície aparada (paramétrica) ou *trimmed parametric surface* - entidade 144

Uma curva simples e fechada no plano Euclidiano divide o plano em dois componentes abertos, conectados e disjuntos, sendo um confinado e o outro não. O componente confinado pode ser chamado de região interior à curva e o não-confinado de região exterior à curva. Retirou-se um exemplo de Systemes (2017a) para ilustrar uma superfície aparada, que pode ser visto na Figura 2.5.

O processo realizado para aparar uma superfície pode ser visto na Figura 2.6, retirada de Systemes (2017a). Sobre uma superfície não-aparada (*untrimmed surface*), gera-se uma curva paramétrica, que irá servir para apará-la. A região em amarelo na Figura 2.6 é retirada, sobrando a superfície aparada no fim.

O domínio da superfície aparada é definida como a região comum interior da envoltória externa e o exterior de cada uma das envoltórias internas, incluindo as curvas da envoltória (contorno). Ressalta-se que a superfície aparada é definida de maneira análoga  $S(u, v)$  à superfície original não-aparada, mas com um domínio diferente. As curvas delimitadoras de cada uma das fronteiras interna e externa da superfície aparada são curvas na superfície  $S$ ,

Tabela 2.1: Tipos de entidades de arquivo IGES

Tipo de entidade IGES	Nome da entidade
514	Entidade de casca
510	Entidade de face
508	Entidade de loop
504	Entidade de aresta
502	Entidade de vértice
416	Entidade de referência externa
408	Entidade de instância de subfigura singular
406 Forma 12	Arquivo de referência externa
402 Forma 7	Entidade de grupo
314	Entidade de definição de cor
308	Entidade de definição de subfigura
198	Superfície toroidal
196	Superfície esférica
194	Superfície cônica circular direita
192	Superfície cilíndrica circular direita
190	Superfície do plano
186	Entidade de objeto manifold solid B-Rep
144	Superfície aparada (paramétrica)
143 somente para importação	Superfície limitada
142	Curva em uma superfície paramétrica
140 somente para importação	Superfície para offset
128	Superfície B-spline racional
126	Curva B-spline racional
124	Entidade de matriz de transformação
123	Entidade de direção
122	Cilindro tabulado
120	Superfície de revolução
118 somente para importação	Superfície regrada
116	Ponto
114 somente para importação	Superfície de spline paramétrica
112	Curva de spline paramétrica
110	Linha
108	Plano
106 forma 12 somente para importação	Dados copiosos - Curva linear 3D por partes
106 forma 11 somente para importação	Dados copiosos - Curva linear 2D por partes
104 somente para importação	Arco cônico
102	Curva composta
100	Arco circular

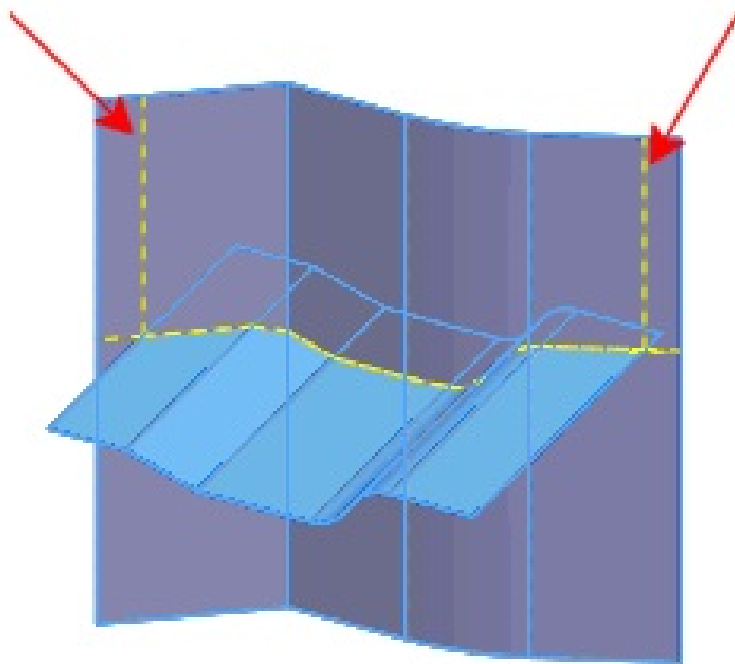


Figura 2.5: Exemplo de superfície aparada por uma curva. Fonte: Systemes (2017a).

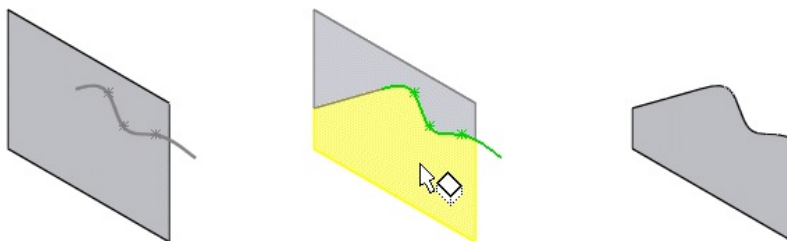


Figura 2.6: Processo para aparar uma superfície. Fonte: Systemes (2017a).

Tabela 2.2: Parâmetros para a entidade 143

Índice	Nome	Tipo	Descrição
1	TYPE	Inteiro ( <i>integer</i> )	0 = As entidades de contorno devem fazer referência apenas a curvas. A representação da superfície associada (localizada em SPTR) pode ser paramétrica. 1 = As entidades de contorno devem fazer referência tanto a curvas quanto ao parâmetro associado. A representação da superfície associada (localizada em SPTR) deve ser paramétrica.
2	SPTR	Ponteiro	Apontado para DE da entidade de superfície não-aparada a ser confinada. Se as curvas de aparamento estão sendo transferidas (TYPE = 1), a representação das superfícies deve ser paramétrica.
3	N	Inteiro ( <i>integer</i> )	O número de entidades de contorno.
4	BDPT(1)	Ponteiro	Apontado para o DE da primeira entidade de contorno (tipo 141).
⋮	⋮	⋮	⋮
3+N	BDPT(N)	Ponteiro	Apontado para o DE da última entidade de contorno (tipo 141).

que são permutadas pelas médias das entidades 142 (curvas em uma superfície paramétrica).

Seja  $S(u, v)$  uma superfície regular parametrizada, cujo domínio não-aparado (*untrimmed*) é um retângulo  $D$  que consiste nos pontos  $(u, v)$  tais que  $a \leq u \leq b$  e  $c \leq v \leq d$  para  $a, b, c$  e  $d$  constantes com  $a \leq b$  e  $c \leq d$ . Assumir que a superfície  $S$  tem seus valores no espaço Euclidiano tridimensional e pode ser expressa como:

$$S = S(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix}$$

para cada par ordenado  $(u, v)$  em  $D$ .

Assumir também que  $S$  esteja sujeita às seguintes condições de regularidade:

- Possui um vetor normal contínuo no interior de  $D$ .
- Cada elemento de  $S$  possui um e somente um elemento correspondente em  $D$ .
- Não existem pontos de singularidade em  $D$ , ou seja, os vetores das primeiras derivadas parciais de  $S$  em cada ponto de  $D$  são linearmente independentes.

Utilizam-se dois tipos de curvas fechadas para definir o domínio da superfície aparada:

**Envoltória (fronteira) externa** Existe uma e apenas uma, pertencente a  $D$ , que pode ser a curva da envoltória de  $D$ .

**Envoltória (fronteira) interna** Pode haver um número qualquer delas, inclusive nenhuma. O conjunto de fronteiras internas satisfaz dois critérios:

1. As curvas, assim como os seus interiores, são mutuamente disjuntos (i.e., não possuem elementos em comum).
2. Cada curva encontra-se no interior da envoltória externa.

Caso a envoltória externa da superfície que está sendo definida for a envoltória de  $D$  e não houver fronteiras internas, essa superfície é não-aparada. Os parâmetros utilizados para definir a entidade 144 - superfície aparada paramétrica -, são exibidos na Tabela 2.3.

Tabela 2.3: ParameterData

Índice	Nome	Tipo	Descrição
1	PTS	Ponteiro	Apontado para DE da superfície que será aparada
2	N1	Número inteiro	0 = a fronteira externa é a fronteira de D 1 = caso contrário
3	N2	Número inteiro	Este número indica a quantidade de curvas simples fechadas que compõem a fronteira interna de uma trimmed surface. Caso nenhuma fronteira interna seja definida, este número será igual a zero.
4	PT0	Ponteiro	Apontado para DE da Curve on a parametric surface entity que compõe a fronteira externa de uma trimmed surface ou zera.
5	PTI(1)	Ponteiro	Apontado para DE da primeira curva simples fechada de fronteira interna (Curve on a parametric surface entity) de acordo com uma ordem arbitrária destas entidades.
:	:	:	:
4+N2	PTI (N2)	Ponteiro	Apontado para DE da última curva simples fechada de fronteira interna (Curve on a parametric surface entity)

## 2.7.2 Exemplo de arquivo IGES

As Figuras 2.7 e 2.8 mostram uma parte de um arquivo .igs aberto em um editor de texto e a imagem correspondente renderizada no SolidWorks, respectivamente.

```

PTC IGES file: tyrerrimren.igs                               S      1
1H,,1H;,11HTYRERRIMREN,15Htyerrimren.igs,                 G      1
52HCreo Parametric by Parametric Technology Corporation,7H2012130,32, G      2
38,7,38,15,11HTYRERRIMREN,1.,1,4HINCH,32768,0.5,15H20160702.233119, G      3
0.115537,1155.42,8HMadhuRaj,7HUnknown,10,0,15H20160702.233119; G      4
314      1      1      1      0      0      0      001000200D      1
314      0      1      1      0      0      0      COLOR      1D      2
314      2      1      1      0      0      0      001000200D      3
314      0      4      1      0      0      0      COLOR      2D      4
314      3      1      1      0      0      0      001000200D      5
314      0      1      1      0      0      0      COLOR      3D      6
314      4      1      1      0      0      0      001000200D      7
314      0      1      1      0      0      0      COLOR      4D      8
314      5      1      1      0      0      0      001000200D      9
314      0      1      1      0      0      0      COLOR      5D     10
314      6      1      1      0      0      0      001000200D     11
314      0      8      1      0      0      0      COLOR      6D     12
314      7      1      1      0      0      0      001000200D     13
314      0      8      1      0      0      0      COLOR      7D     14
314      8      1      1      0      0      0      001000200D     15
314      0      2      1      0      0      0      COLOR      8D     16
314      9      1      1      0      0      0      001000200D     17
314      0      2      1      0      0      0      COLOR      9D     18
314     10      1      1      0      0      0      001000200D     19
314      0      8      1      0      0      0      COLOR     10D     20
314     11      1      1      0      0      0      001000200D     21
314      0      5      1      0      0      0      COLOR     11D     22
314     12      1      1      0      0      0      001000200D     23
314      0      8      1      0      0      0      COLOR     12D     24
314     13      1      1      0      0      0      001000200D     25
314      0      8      1      0      0      0      COLOR     13D     26
314     14      1      1      0      0      0      001000200D     27
314      0      5      1      0      0      0      COLOR     14D     28
314     15      1      1      0      0      0      001000200D     29
314      0      8      1      0      0      0      COLOR     15D     30
124     16      1      1      0      0      0      001000000D     31
124      0      0      2      0      0      0      XFORM      1D     32
100     18      1      1      0      0      31      001010000D     33
100      0     -11      1      0      0      0      ARC       1D     34

```

Figura 2.7: Parte de arquivo .igs aberto em um editor de textos.



Figura 2.8: Imagem renderizada no SolidWorks. Fonte: Balan (2016)



# Capítulo 3

## Formulação tridimensional do método dos elementos de contorno

### 3.1 Introdução

Neste capítulo será desenvolvida a equação integral de contorno para problemas de condução de calor em sólidos (condução de calor sem geração). Esta equação será, posteriormente, discretizada em elementos de contorno, obtendo-se então a formulação do MEC.

#### 3.1.1 Equação de Laplace para problemas tri-dimensionais

A Figura 3.1 descreve um sólido de volume  $V$  e superfície  $S$ .

A condução de calor em um sólido é descrita pela equação:

$$\nabla^2 T = \frac{\dot{q}}{k} \quad (3.1)$$

onde  $T$  representa a temperatura em um dado ponto  $(x_1, x_2, x_3)$  do corpo,  $\dot{q}$  é uma fonte, se  $\dot{q} > 0$ , ou um sorvedouro de calor, se  $\dot{q} < 0$ . A constante  $k$  representa a condutividade térmica do material, determinada experimentalmente.

A equação (3.1) é conhecida como equação de Laplace (quando  $\dot{q} = 0$ ) ou equação de Poisson (quando  $\dot{q} \neq 0$ ). Embora neste trabalho esta equação seja usada em problemas de condução de calor, ela governa vários problemas, como escoamento de fluidos, escoamento em meios porosos, difusão de íons, fluxo elétrico, entre outros.

A superfície  $S$  é dividida em duas partes  $S = S_1 + S_2$ , onde

$S_1$ : Superfície onde a temperatura  $T$  é conhecida, ou seja,  $T = \bar{T}$ .

$S_2$ : Superfície onde o fluxo  $q$  é conhecido, ou seja,  $q = \bar{q}$ .

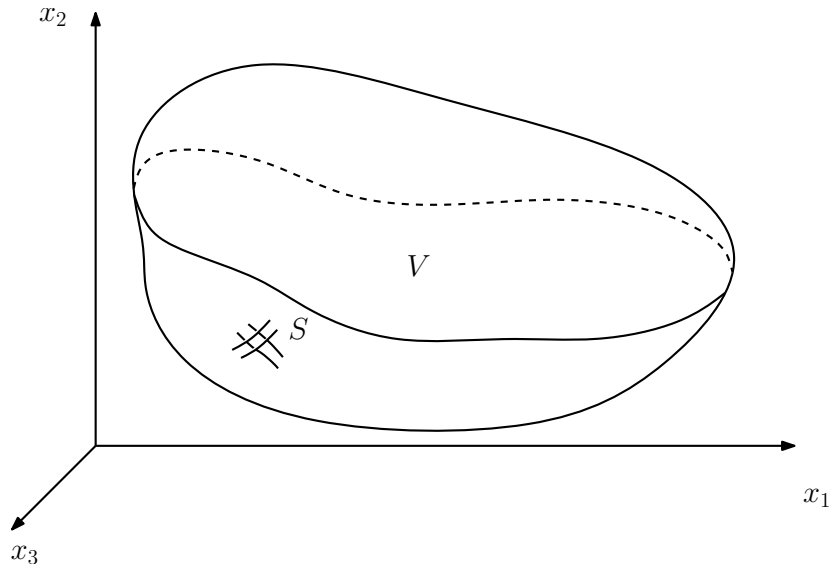


Figura 3.1: Sólido com volume  $V$  e superfície  $S$ .

sendo que

$$q = -k \frac{\partial T}{\partial n}$$

e  $\mathbf{n}$  é a normal ao contorno.

## 3.2 Solução fundamental

A formulação do MEC utiliza a solução fundamental. Para a equação de Laplace, esta é a influência do potencial em um meio infinito quando a fonte é concentrada em um ponto. Segundo Campos (2016), ela corresponde à solução particular analítica da equação de Fourier quando o termo não-homogêneo (termo referente à fonte) é igual ao delta de Dirac, ou seja:

$$\nabla^2 T^* = \Delta(P, Q). \quad (3.2)$$

onde  $P$  é o ponto fonte (ponto onde o delta de Dirac é aplicado) e  $Q$  é o ponto campo (ponto onde o valor de  $T^*$  é medido). Define-se a função delta de Dirac como:

$$\Delta(P, Q) = \begin{cases} \infty & \text{se } P = Q \\ 0 & \text{no restante do intervalo} \end{cases}$$

A solução fundamental de temperatura, considerando o espaço tridimensional, é dada por:

$$T^* = \frac{1}{4\pi R}. \quad (3.3)$$

onde  $R$  é a distância entre os pontos  $P$  e  $Q$ . A solução fundamental de fluxo é dada por:

$$q^* = -k \frac{\partial T^*}{\partial n} = -\frac{\partial R}{4\pi R} \quad (3.4)$$

### 3.3 Equação integral de contorno

Nesta seção será desenvolvida a equação integral de contorno para a equação de Laplace. Esta equação será, posteriormente, discretizada em elementos de contorno, obtendo-se então a formulação do MEC. Multiplicando a equação (3.1) por uma função peso  $\omega(x, y)$ , considerando  $\dot{q} = 0$  e integrando sobre o domínio  $V$ , assume-se que o resultado da integral é zero (método dos resíduos ponderados). Assim, tem-se:

$$\begin{aligned} \iiint_V (\nabla^2 T) \omega dV &= 0, \\ \iiint_V \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) \omega dV &= 0, \\ \iiint_V \frac{\partial^2 T}{\partial x^2} \omega dV + \iiint_V \frac{\partial^2 T}{\partial y^2} \omega dV + \iiint_V \frac{\partial^2 T}{\partial z^2} \omega dV &= 0 \end{aligned} \quad (3.5)$$

Pelo teorema de Gauss-Green, tem-se:

$$\iint_S f(x, y) n_x dS = \iiint_V \frac{\partial f}{\partial x} dV$$

onde  $f$  é uma função qualquer,  $n_x$  é a componente na direção  $x$  do vetor unitário  $\mathbf{n}$  normal ao contorno  $S$  do volume  $V$ . Aplicando o teorema dado na primeira parcela da Eq.(3.5), tem-se:

$$\iint_S \frac{\partial T}{\partial x} \omega n_x dS = \iiint_V \frac{\partial}{\partial x} \left( \frac{\partial T}{\partial x} \omega \right) dV.$$

Aplicando a regra da derivada do produto de funções, tem-se:

$$\iint_S \frac{\partial T}{\partial x} \omega n_x dS = \iiint_V \frac{\partial^2 T}{\partial x^2} \omega dV + \iiint_V \frac{\partial T}{\partial x} \frac{\partial \omega}{\partial x} dV.$$

Escrevendo os termos da equação de forma conveniente, segue:

$$\iiint_V \frac{\partial^2 T}{\partial x^2} \omega dV = \iint_S \frac{\partial T}{\partial x} \omega n_x dS - \iiint_V \frac{\partial T}{\partial x} \frac{\partial \omega}{\partial x} dV. \quad (3.6)$$

De modo análogo, obtém-se:

$$\iiint_V \frac{\partial^2 T}{\partial y^2} \omega dV = \iint_S \frac{\partial T}{\partial y} \omega n_y dS - \iiint_V \frac{\partial T}{\partial y} \frac{\partial \omega}{\partial y} dV. \quad (3.7)$$

$$\iiint_V \frac{\partial^2 T}{\partial z^2} \omega dV = \iint_S \frac{\partial T}{\partial z} \omega n_z dS - \iiint_V \frac{\partial T}{\partial z} \frac{\partial \omega}{\partial z} dV. \quad (3.8)$$

Substituindo as Eqs.(3.6) e (3.7) na Eq.(3.5), tem-se:

$$\iint_S \left( \frac{\partial T}{\partial x} \omega n_x + \frac{\partial T}{\partial y} \omega n_y + \frac{\partial T}{\partial z} \omega n_z \right) dS - \iiint_V \left( \frac{\partial T}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial T}{\partial y} \frac{\partial \omega}{\partial y} + \frac{\partial T}{\partial z} \frac{\partial \omega}{\partial z} \right) dV = 0.$$

Simplificando

$$\iint_S \frac{\partial T}{\partial n} \omega dS - \iiint_V \left( \frac{\partial T}{\partial x} \frac{\partial \omega}{\partial x} + \frac{\partial T}{\partial y} \frac{\partial \omega}{\partial y} + \frac{\partial T}{\partial z} \frac{\partial \omega}{\partial z} \right) dV = 0. \quad (3.9)$$

Considerando as igualdades:

$$\iiint_V \frac{\partial T}{\partial x} \frac{\partial \omega}{\partial x} dV = \iint_S T \frac{\partial \omega}{\partial x} n_x dS - \iiint_V T \frac{\partial^2 \omega}{\partial x^2} dV \quad (3.10)$$

$$\iiint_V \frac{\partial T}{\partial y} \frac{\partial \omega}{\partial y} dV = \iint_S T \frac{\partial \omega}{\partial y} n_y dS - \iiint_V T \frac{\partial^2 \omega}{\partial y^2} dV \quad (3.11)$$

e

$$\iiint_V \frac{\partial T}{\partial z} \frac{\partial \omega}{\partial z} dV = \iint_S T \frac{\partial \omega}{\partial z} n_z dS - \iiint_V T \frac{\partial^2 \omega}{\partial z^2} dV. \quad (3.12)$$

Substituindo as equações (3.10), (3.11) e (3.12) na equação (3.9), tem-se:

$$\iint_S \frac{\partial T}{\partial n} \omega dS - \iint_S \left( T \frac{\partial \omega}{\partial x} n_x + T \frac{\partial \omega}{\partial y} n_y + T \frac{\partial \omega}{\partial z} n_z \right) dS + \iiint_V T \left( \frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2} + \frac{\partial^2 \omega}{\partial z^2} \right) dV = 0,$$

$$\iint_S \frac{\partial T}{\partial n} \omega dS - \iint_S T \frac{\partial \omega}{\partial n} dS + \iiint_V T \nabla^2 \omega dV = 0. \quad (3.13)$$

Com o objetivo de obter uma equação integral que não possua integrais de domínio (integrais de volume) a função  $\omega$  deve ser escolhida de forma que a integral de domínio da Eq. (3.13) desapareça. Qualquer função harmônica, ou seja, função que o laplaciano é igual a zero, satisfaz essa exigência. Contudo, por razões numéricas, a escolha mais adequada é a função cujo laplaciano é o delta de Dirac.

$$\nabla^2 \omega = -\frac{\delta(P)}{k},$$

o que implica que  $\omega = T^*$ . Então, tem-se:

$$\iint_S \frac{\partial T}{\partial n} T^* dS - \iint_S T \frac{\partial T^*}{\partial n} dS + \iiint_V T \frac{[-\delta(P)]}{k} dV, \quad (3.14)$$

Tomando o ponto fonte no interior do domínio  $V$ , pela propriedade do delta de Dirac, tem-se:

$$\iint_S \frac{\partial T}{\partial n} T^* dS - \iint_S T \frac{\partial T^*}{\partial n} dS - \frac{T(P)}{k} = 0.$$

Multiplicando os termos por  $-k$ , obtém-se:

$$\iint_S -k \frac{\partial T}{\partial n} T^* dS + \iint_S T \left( \frac{k \partial T^*}{\partial n} \right) dS + T(P) = 0,$$

$$T(P) = \iint_S T q^* dS - \iint_S T^* q dS. \quad (3.15)$$

A Eq.(3.15) é a *equação integral de contorno* quando o ponto fonte encontra-se no interior do domínio.

Pode-se escrever a integral para representar um ponto interno ou no contorno, da seguinte forma:

$$cT(P) = \iint_S T q^* dS - \iint_S T^* q dS \quad (3.16)$$

onde  $c$  é uma constante que depende da posição do ponto fonte.

### 3.4 Discretização das equações integrais de contorno

Basicamente, a formulação do MEC transforma equações diferenciais em equações integrais de contorno, eliminando, assim, a discretização do domínio. Essas integrais podem ser resolvidas de forma numérica ou analítica, com a integração feita ao longo do contorno.

Uma estratégia para o cálculo das integrais ao longo da superfície  $S$  é, como indicado na Figura 3.2, a divisão da superfície  $S$  em uma soma de superfícies menores  $S_1, S_2, \dots, S_n$ , ou seja,  $S = \sum_{i=1}^n S_i$ , onde  $n$  é o número de superfícies em que o contorno foi dividido. Uma vez que estas superfícies podem ter uma forma qualquer, cada superfície  $S_i$  será aproximada por uma superfície que seja facilmente descrita por funções matemáticas. Por simplicidade, ela é quase sempre dada por superfícies elementares que podem ser mapeadas em triângulos e quadrados.

Dessa maneira, cada superfície  $S_1, S_2, \dots, S_n$  é aproximada por superfícies,  $\Gamma_1, \Gamma_2, \dots, \Gamma_n$ , com formas descritas por polinômios que podem ser planas ou não, chamados elementos de contorno.

Daí, tem-se que

$$\Gamma = \sum_{i=1}^n \Gamma_i \quad (3.17)$$

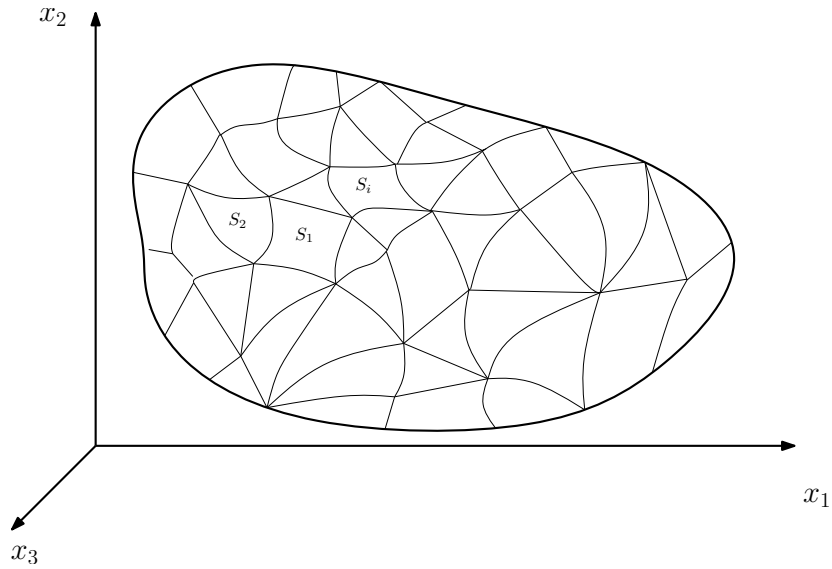


Figura 3.2: Superfície dividida em pequenas partes.

e

$$\lim_{n \rightarrow \infty} \sum_{i=1}^n \Gamma_i = S \quad (3.18)$$

na qual  $\Gamma$  = Superfície aproximada. Neste trabalho, estas superfícies são formadas por triângulos planos, o que introduz uma aproximação na superfície original se esta não for plana.

Desta forma, a equação integral de contorno (3.16) é escrita como:

$$cT(P) = \sum_{j=1}^n \iint_{\Gamma_j} Tq^* d\Gamma - \sum_{j=1}^n \iint_{\Gamma_j} T^* q d\Gamma. \quad (3.19)$$

As variáveis  $u$  e  $q$  sobre cada elemento  $j$  são definidas em termos de seus valores nodais como:

$$u = \begin{bmatrix} N_1 & N_2 & \cdots & N_N \end{bmatrix} \begin{Bmatrix} u^1 \\ u^2 \\ \vdots \\ u^N \end{Bmatrix}$$

e

$$q = \begin{bmatrix} N_1 & N_2 & \cdots & N_N \end{bmatrix} \begin{Bmatrix} q^1 \\ q^2 \\ \vdots \\ q^N \end{Bmatrix}$$

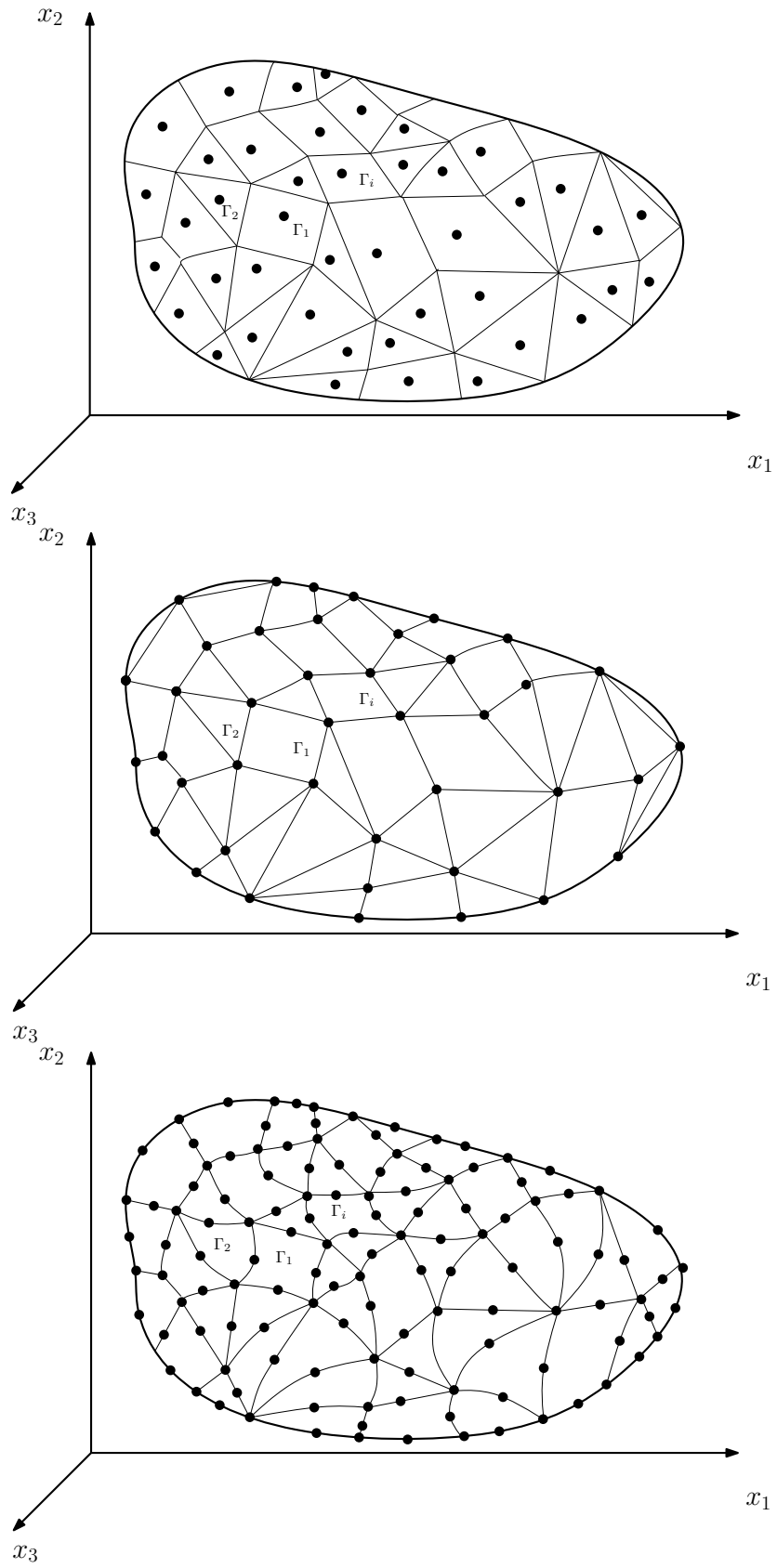


Figura 3.3: Contorno tridimensional dividido em: (a) elementos constantes; (b) elementos lineares; e (c) elementos quadráticos

onde  $N_j$  são as funções de forma.

As funções de forma (ou funções de interpolação) são dadas em termos de coordenadas homogêneas definidas sobre cada família de elementos. A geometria dos elementos pode ser definida a partir da posição dos nós utilizando as funções de forma. Assim, as coordenadas  $(x_1, x_2, x_3)$  de um ponto qualquer do elemento é escrita em função das coordenadas dos  $N$  nós do elemento:

$$\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & \cdots & N_N & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \cdots & 0 & N_N & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & \cdots & 0 & 0 & N_N \end{bmatrix} \begin{Bmatrix} x_1^1 \\ x_2^1 \\ x_3^1 \\ \vdots \\ x_1^N \\ x_2^N \\ x_3^N \end{Bmatrix} \quad (3.20)$$

A Equação (3.19) pode ser resolvida numericamente. Para tal, deve-se discretizar o contorno com elementos conforme as Figuras 3.3(a), 3.3(b) e 3.3(c) onde elementos de contorno de diferentes ordens são usados.

### 3.5 Elementos triangulares

Neste trabalho, os elementos de superfície  $\Gamma_i$  são descritos por elementos triangulares planos pois são mais fáceis de gerar e possuem jacobiano constante ao longo do elemento, como será mostrado posteriormente.

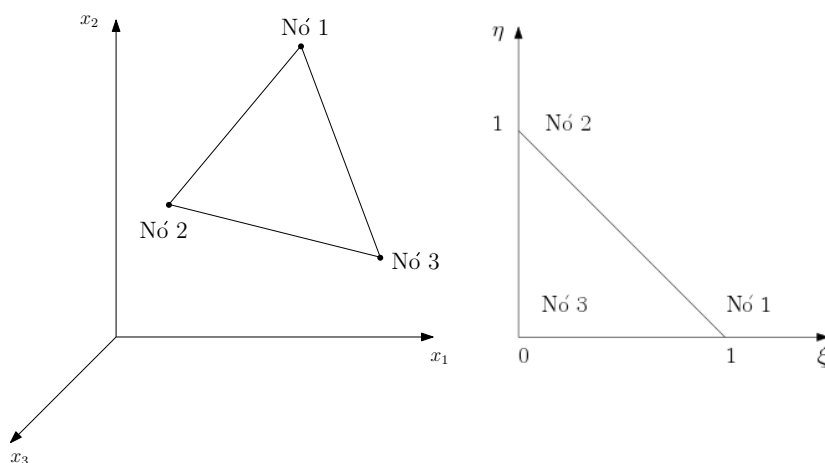


Figura 3.4: Elemento triangular linear: (a) no espaço real  $(x_1, x_2, x_3)$  e (b) no espaço paramétrico  $(\xi, \eta)$ .

Uma vez que a integração de Gauss, por simplicidade, será realizada no intervalo  $[0, 1]$ ,



é coerente fazer o mapeamento do elemento  $\Gamma_i$  no intervalo  $[0, 1]$ . Assim observamos o elemento real, mostrado na Figura 3.4(a) e o elemento de integração mostrado na Figura 3.4(b).

A partir dessas figuras chega-se às seguintes funções de forma para o elemento:

$$\begin{cases} N_1 = \xi \\ N_2 = \eta \\ N_3 = 1 - \xi - \eta \end{cases} \quad (3.21)$$

Integrando o elemento triangular, como indica a figura 3.4(b), obtemos a seguinte equação:

$$I_i = \iint_{\Gamma_i} J f(x, y, z) d\Gamma \quad (3.22)$$

na qual  $J$  é o Jacobiano dado por:

$$J = \sqrt{g_1^2 + g_2^2 + g_3^2} \quad (3.23)$$

cujos valores de  $g_1$ ,  $g_2$  e  $g_3$  são dados por:

$$\begin{cases} g_1 = \frac{dy}{d\xi} \frac{dz}{d\eta} - \frac{dy}{d\eta} \frac{dz}{d\xi} \\ g_2 = \frac{dz}{d\xi} \frac{dx}{d\eta} - \frac{dz}{d\eta} \frac{dx}{d\xi} \\ g_3 = \frac{dx}{d\xi} \frac{dy}{d\eta} - \frac{dx}{d\eta} \frac{dy}{d\xi} \end{cases} \quad (3.24)$$

O vetor normal unitário é dado por:

$$\mathbf{n} = \frac{[g_1 \ g_2 \ g_3]}{J} \quad (3.25)$$

Uma mudança de variável, então, será requerida na equação (3.22) para que as duas integrais ocorram no intervalo de  $[0, 1]$ . Para isso, consideramos uma variável  $\xi'$  função de  $\xi$  de forma que:

$$\begin{cases} \xi'(\xi = 0) = 0 \\ \xi'(\xi = 1 - \eta) = 1 \end{cases} \quad (3.26)$$

Assim, considerando uma relação linear:

$$\xi' = a\xi + b \quad (3.27)$$

Obtemos dessa relação:

$$\begin{cases} \xi = (1 - \eta)\xi' \\ \frac{d\xi}{d\xi'} = 1 - \eta \end{cases} \quad (3.28)$$

Note que as derivadas das funções de forma dadas por:

$$\begin{cases} \frac{\partial N_1}{\partial \xi} = 1 \\ \frac{\partial N_1}{\partial \eta} = 0 \\ \frac{\partial N_2}{\partial \xi} = 0 \\ \frac{\partial N_2}{\partial \eta} = 1 \\ \frac{\partial N_3}{\partial \xi} = -1 \\ \frac{\partial N_3}{\partial \eta} = -1 \end{cases} \quad (3.29)$$

não dependem de  $\xi$  e  $\eta$ . Por isso,  $g_1$ ,  $g_2$  e  $g_3$ , o vetor normal unitário  $\mathbf{n}$  de 3.25 e o jacobiano  $J$  são constantes ao longo de todo o elemento. Por fim, a integral no elemento é escrita como:

$$I_i = \int_0^1 \int_0^1 f(x, y, z) J \frac{d\xi}{d\xi} d\eta d\xi = \sum_{k=1}^{npg} \sum_{l=1}^{npg} f(x, y, z) J \frac{d\xi}{d\xi} w \quad (3.30)$$

### 3.6 Elementos de contorno triangulares constantes

Na discretização utilizando elementos constantes, a geometria é aproximada por triângulos planos com um nó no centroide de cada elemento. Além disso, tanto a temperatura quanto o fluxo são assumidos como constantes ao longo de todo o elemento.

Como o nó  $j$  sempre estará sempre no centroide do elemento  $j$  (será sempre em uma região suave do contorno),  $c = \frac{1}{2}$ . Desta forma, a equação integral é aproximada por:

$$\frac{1}{2}T_i = \sum_{j=1}^n \left[ T_j \iint_{\Gamma_j} q^* d\Gamma \right] - \sum_{j=1}^n \left[ q_j \iint_{\Gamma_j} T^* d\Gamma \right], \quad (3.31)$$

onde  $i$  corresponde ao nó do  $i$ -ésimo elemento. Daí, tem-se:

$$-\frac{1}{2}T_i + \sum_{j=1}^n \left[ T_j \iint_{\Gamma_j} q^* d\Gamma \right] = \sum_{j=1}^n \left[ q_j \iint_{\Gamma_j} T^* d\Gamma \right]. \quad (3.32)$$

Chamando

$$H_{ij} = \begin{cases} \iint_{\Gamma_j} q^* d\Gamma, & \text{se } i \neq j \\ -\frac{1}{2} + \iint_{\Gamma_j} q^* d\Gamma, & \text{se } i = j \end{cases}$$

e

$$G_{ij} = \iint_{\Gamma_j} T^* d\Gamma \quad (3.33)$$

pode-se escrever a equação matricial da seguinte forma:

$$\sum_{j=1}^n [H_{ij}T_j] = \sum_{j=1}^n [G_{ij}q_j]. \quad (3.34)$$

Manipulando a equação matricial de forma que os termos desconhecidos fiquem todos do lado esquerdo e os demais termos do lado direito, obtém-se um sistema linear da forma:

$$\mathbf{A} \mathbf{x} = \mathbf{b} \quad (3.35)$$

onde o vetor  $\mathbf{x}$  contém as temperaturas e os fluxos desconhecidos ao longo do contorno  $\Gamma$ .

# Capítulo 4

## Geração de Malha

### 4.1 Introdução

Para o caso dos elementos finitos, uma malha pode ser vista como a partição de um dado domínio em subdomínios, chamados elementos. Ressalta-se que todos os pontos do domínio devem estar localizados dentro de um elemento, sem sobreposição de elementos. Nos elementos de contorno, a ideia é similar, mas apenas o contorno é dividido em sub-contornos, também chamados de elementos - agora *elementos de contorno*. Em problemas tri-dimensionais, os elementos são comumente quadriláteros ou triangulares - os utilizados nesse estudo - mas podem ter outras formas. Convém destacar que a precisão do MEC, assim como do MEF, depende do número de nós na malha, de quais locais eles foram inseridos e também da forma dos elementos.

As malhas podem ser categorizadas em estruturadas e não-estruturadas. Uma malha estruturada, como uma grade cúbica, tem a propriedade de que seus vértices podem ser numerados de tal forma que simples sufixos aritméticos podem determinar quais vértices são compartilhados pelos elementos. As malhas não-estruturadas são mais versáteis por sua habilidade de combinar boas formas de elementos com domínios complexos e tamanhos de elementos variando desde muito pequenos até muito grandes. O problema da geração de malha ocorre quando há a criação de elementos não conformes, assim como os da Figura 4.1, onde um elemento compartilha uma face com dois outros elementos. A meta da geração de malha é criar elementos conformes com a geometria do domínio, além de atender às restrições em tamanho e forma. Baseado nessas características, há o conceito de qualidade do elemento na seção 4.4.

### 4.2 Malhas estruturadas e não-estruturadas

Há a classificação em malhas estruturadas e não-estruturadas, que está relacionada com a disposição relativa dos diferentes elementos. Conforme Gonçalves et al. (2007), nas ma-

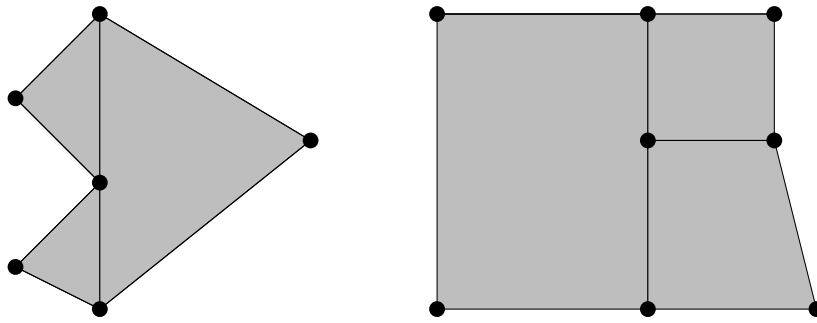


Figura 4.1: Elementos não-conformes

lhas estruturadas, os elementos são dispostos em famílias de linhas, em que membros de uma determinada família não se cruzam uns com os outros e atravessam cada membro de outras famílias apenas uma vez, permitindo que as linhas de um certo conjunto sejam numeradas consecutivamente. A Figura ?? ilustra essa ideia. Com essa técnica, é possível saber o número do elemento adjacente com uma simples relação algébrica. Isso faz com que o processo de programação seja facilitado, diminuindo o acesso à memória e a quantidade de tempo necessária para resolver o problema. Outra vantagem é a simplificação da matriz do sistema de equações algébricas. Apesar dessas vantagens, há desvantagens no uso das malhas estruturadas, tais como a dificuldade (ou até impossibilidade) de aplicação em geometrias complexas e a concentração de elementos em uma região da malha, podendo gerar desperdício de recursos. As Figuras 4.3 e 4.4, ambas retiradas de Maliska (2004), mostram exemplos de malhas estruturadas e não-estruturadas.

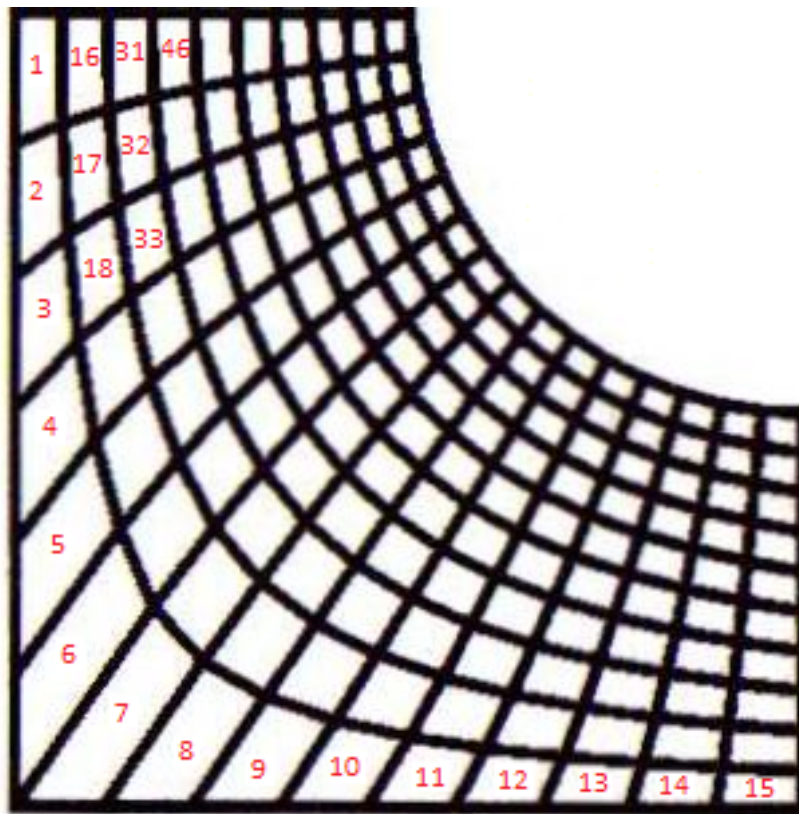


Figura 4.2: Numeração dos elementos. Fonte: Aguirre (2017).

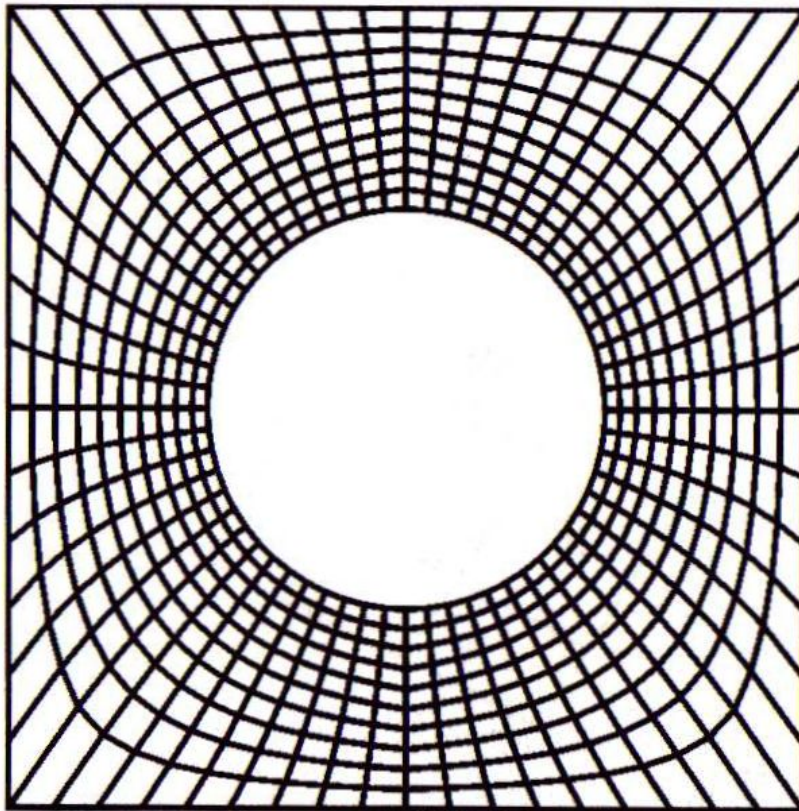


Figura 4.3: Malha estruturada. Fonte: Maliska (2004).

As malhas não-estruturadas diferem por não terem a disposição relativa dos elementos que ajuda a numerar os elementos em famílias de linhas. Apesar dessa desvantagem, apresenta a vantagem de adequar-se a domínios arbitrários e geometrias complexas com relativa facilidade. Como visto em Batista (2005), possui como principais técnicas a triangulação de Delaunay, avanço de fronteira e decomposição espacial recursiva.

#### 4.2.1 Malhas não-estruturadas em domínios planos

As malhas não-estruturadas de tamanho variável de elementos são necessárias para o refinamento adaptativo e para a adequação a contornos geométricos complicados (LO, 2015). Malhas não-estruturadas triangulares servem bem para a situação de análise adaptativa, já que os triângulos são menos sensíveis à distorção de forma, de tal modo que malhas com mudança rápida no tamanho dos elementos possam ser construídas. Algumas técnicas de geração de malha para produzir malhas não estruturadas compatíveis com o espaçamento nodal em um domínio 2D são:

1. Triangulação de Delaunay - *Delaunay triangulation (DT)*;
2. Avanço de fronteira - *Advancing-front technique (AFT)*;
3. Geração de malha utilizando contorno - *Mesh generation using contour*;

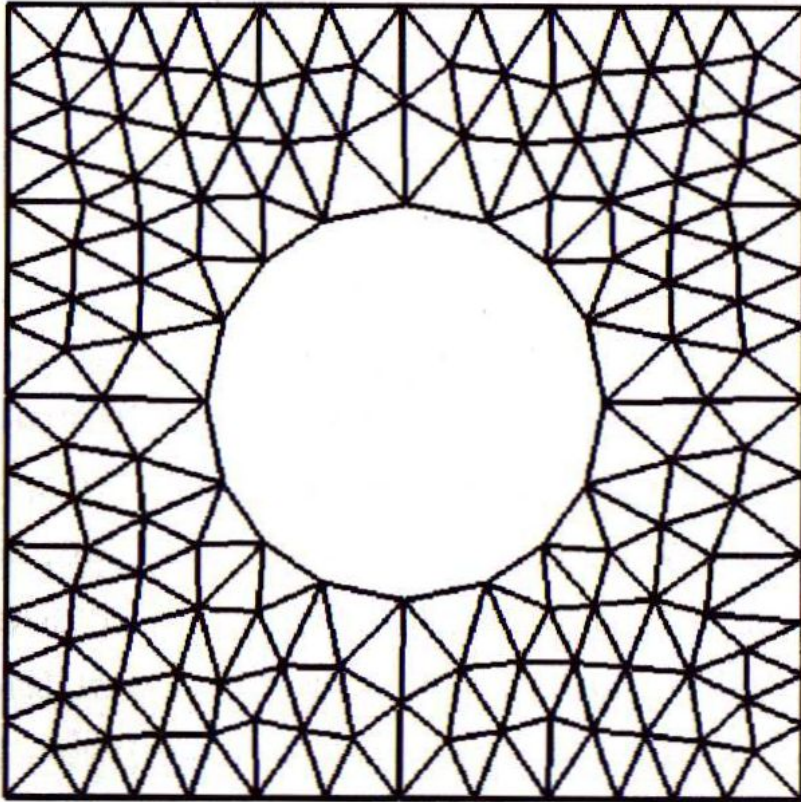


Figura 4.4: Malha não estruturada. Fonte: Maliska (2004).

4. *Coring technique*;
5. Decomposição *quadtree* - *Quadtree decomposition*;
6. Refinamento por subdivisão - *Refinement by subdivision*.

Nesse trabalho, é apresentada a técnica 1 - triangulação de Delaunay -, abordada por ser a utilizada no pacote *MESH2D*. Para maiores informações acerca das outras técnicas, consultar Lo (2015). A triangulação de Delaunay será demonstradas na seção 4.5.

### 4.3 Propriedades das malhas

Considere uma malha composta por elementos triangulares de um domínio plano  $\Omega$ , definida por um conjunto  $C$  constituído por um número finito de pontos, segmentos e triângulos. Os elementos  $\varepsilon$  dessa malha devem satisfazer as seguintes propriedades, segundo Filho (2004).

1. Os elementos devem ser gerados de forma a constituir uma malha conforme, ou seja, os elementos devem preencher todo o domínio  $\Omega$ , e a intersecção de dois triângulos em  $C$  deve ser de tal modo que ocorra nos vértices dos triângulos, ou seja, reduzida a uma aresta.

2. Se possível, a variação em tamanho entre dois elementos adjacentes seja feita de forma progressiva e gradual.
3. A densidade de elementos em determinadas regiões do domínio deve ser mais intensa, especialmente nos locais com altos gradientes de solução.
4. Para malhas com elementos triangulares, a existência de ângulos internos muito pequenos deve ser evitada. Considera-se um ângulo em torno de  $13^\circ$  para controle do ângulo mínimo.
5. É desejável que os elementos sejam gerados e orientados por algumas direções preferenciais de acordo com o problema físico.

## 4.4 Qualidade da malha

A maioria das aplicações restringe tanto as formas quanto os tamanhos dos elementos e essas restrições tem origem em vários motivos, segundo Cheng, Dey e Shewchuk (2013). O primeiro é que ângulos grandes (perto de  $180^\circ$ ) podem causar erros de interpolação significativos. Segundo que os ângulos pequenos (perto de  $0^\circ$ ) podem fazer com que a matriz de rigidez seja mal condicionada. Terceiro que os elementos com tamanhos menores oferecem maior precisão, mas tem alto custo computacional. Finalmente, o quarto motivo é que elementos muito pequenos ou esbeltos pode induzir a instabilidade nos métodos de integração empregados pelas simulações.

### 4.4.1 O problema da melhoria da qualidade da malha

Conforme Park e Shontz (2011), pode-se descrever o problema da melhoria da qualidade de malha como a minimização da função:

$$f(x) = \frac{1}{n} \sum_{i=1}^n q_i^2 \quad (4.1)$$

onde  $f$  é a qualidade global da malha, medida pela qualidade média dos elementos;  $q_i$  é a qualidade do elemento  $i$ ;  $n$  é o número de elementos da malha. Deve-se procurar minimizar a função, tendo em vista que valores menores de  $q_i$  denotam elementos de melhor qualidade por essa métrica.

A Tabela 4.1 mostra exemplos de tipos diferentes de métricas utilizadas para análise e melhoria de malhas. Considera-se  $L_{12}$ ,  $L_{13}$  e  $L_{23}$  como os comprimentos dos três lados do triângulo;  $L_{min}$  e  $L_{max}$  são, respectivamente, os comprimentos mínimo e máximo;  $S$  é a área;  $A$  é o Jacobiano para o triângulo físico;  $W$  é o Jacobiano para o triângulo de referência. Os intervalos para as métricas de qualidade são: 1 até  $\infty$  para (1), (3), (4) e (5), e 0 até  $\infty$



para (2). O elemento ideal é considerado como um triângulo equilátero - valores menores demonstram malhas com qualidade melhor e procura-se minimizar a Equação (4.1).

Tabela 4.1: Definição de métricas de qualidade de malha

Métrica de Qualidade	Fórmula ( $q_i$ )
(1) Razão entre lados (EDGE)	$L_{min}/L_{max}$
(2) Área (AREA)	$S_{cur} - S_{avg}$
(3) Raiz quadrada média dos lados (RMS)	$\sqrt{(L_{12}^2 + L_{13}^2 + L_{23}^2)/3}$
(4) Inverso da razão média (IMR)	$\ AW^{-1}\ _F^2 / (2 \det(AW^{-1}))$
(5) Razão de aspecto (AR)	$\sqrt{3} (L_{12}^2 + L_{13}^2 + L_{23}^2) / 12S_i$

## 4.5 Geração de malha por triangulação de Delaunay

### 4.5.1 Introdução

Para que bons resultados sejam obtidos, é necessário que a análise do problema seja feita com uma malha adequada. Primeiramente, deve-se definir o que é uma malha adequada, que varia entre os diferentes problemas. Para reduzir o erro numérico de discretização na análise, a qualidade da malha deve ser otimizada de tal forma que o tamanho do elemento esteja de acordo com o espaçamento nodal especificado e fazer com que o elemento seja o mais equilátero possível.

No problema estudado, atenção especial deve ser dada aos problemas com geometria com faces curvas, onde pode haver a degeneração de elementos. Isso ocorre quando o triângulo gerado na malha em duas dimensões passa a ter aresta de comprimento zero em três dimensões. Antes de falar sobre a Triangulação de Delaunay (DT) propriamente dita, convém discorrer sobre a partição de Voronoï (ou partição de Dirichlet), tendo em vista que aquela é a forma dual desta (OLIVEIRA, 1995; LO, 2015). Dirichlet mostrou, em 1850, que para um dado conjunto de pontos em duas dimensões, é possível a partição do plano em células convexas baseado no critério da proximidade. Voronoï, em 1908, complementou a teoria proposta por Dirichlet com estudos de formas algébricas relacionadas a esse critério geométrico. Ressalta-se que a forma algébrica pode ser aplicada também para casos tridimensionais.

No ano de 1934, Delaunay determinou que há a possibilidade de construir uma triangulação baseando-se no diagrama de Voronoï por dualidade, conectando pontos associados com as células às vizinhas - Figura 4.5. Conforme Oliveira (1995), a estrutura dual dessa partição é uma triangulação única com notáveis propriedades.

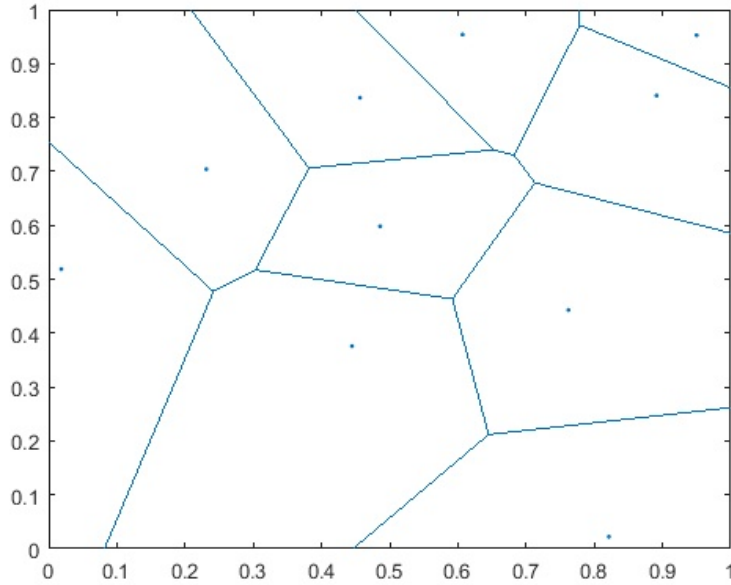


Figura 4.5: Diagrama (ou partição) de Voronoi de 10 pontos.

A DT é utilizada desde o princípio da geração de malha e possui uma vantagem significativa para casos bidimensionais: dentre todas as possíveis triangulações para um dado conjunto de pontos, a DT maximiza os ângulos mínimos (CHENG; DEY; SHEWCHUK, 2013). Além disso, ainda segundo Cheng, Dey e Shewchuk (2013), ela também otimiza outros critérios geométricos relacionados à precisão de interpolação.

Para esse capítulo é oportuno discorrer sobre a definição de simplex, que é a figura geométrica mais elementar em um espaço  $R^n$ . Os simplexes dos espaços utilizados nesse trabalho são:

- $R^0$ : ponto;
- $R^1$ : segmento de reta;
- $R^2$ : triângulo;
- $R^3$ : tetraedro.

#### 4.5.1.1 Triangulação de Delaunay e a envoltória de um conjunto de pontos

Seja  $P = \{p_i, i = 1, n\}$  um conjunto de  $n$  pontos distintos em um plano, e definindo um conjunto de polígonos  $V = \{V_i, i = 1, n\}$ , onde

$$V_i = \{x \in R^2 \mid \|x - p_i\| < \|x - p_j\|, \forall j \neq i\}$$

e  $\|x - p\|$  representa a distância entre os pontos  $x$  e  $p$  e  $V_i$  representa a região no plano cujos pontos estão mais próximos ao ponto  $p_i$  do que de qualquer outro ponto  $p$  do conjunto e dá

origem a um poliedro. Além disso,  $V_i$  é um polígono convexo aberto, também conhecido como polígono de Voronoï. A coleção dos polígonos de Voronoï  $\mathbf{V}$  é chamada de partição de Voronoï ou de Dirichlet.

É comum que o vértice de um polígono de Voronoï seja compartilhado por outros dois polígonos vizinhos, conectando, assim, os três pontos associados aos polígonos para formar um triângulo - Figura 4.5.

Conforme Oliveira (1995), os polígonos de Voronoï podem ser vistos como células em processo de crescimento. Cada uma se propaga a partir de seu núcleo, crescendo simultânea e uniformemente em todas as direções; o crescimento cessa quando houver contato com outra célula em crescimento. Todas as células (exceto as cujos núcleos situam-se na envoltória  $V$ ) dividem, então, a região do plano em polígonos convexos fechados e não superpostos, sendo um polígono em torno de cada núcleo. Somando-se os polígonos fechados e os polígonos abertos da envoltória, obtém-se a partição de Voronoï para o plano.

Percebe-se que caso a taxa de crescimento seja igual para todas as células, o primeiro ponto de contato entre duas células ocorre no ponto médio entre os núcleos. Analogamente, os outros pontos de contato são equidistantes dos dois núcleos e situam-se sobre uma aresta comum entre os dois polígonos. A aresta continua crescendo até que ocorre a intersecção com a borda de uma terceira célula, formando o vértice do polígono de Voronoï. Cabe ressaltar que o vértice se encontra equidistante dos núcleos das três células, tornando-se o circuncentro do triângulo definido pelos núcleos.

Definem-se células de Voronoï vizinhas como aquelas que compartilham um lado. Uma nova partição da envoltória convexa de  $V$  pode ser criada ao se anexarem os triângulos formados pela conexão dos núcleos de todas as células vizinhas. À nova partição dá-se o nome de triangulação de Delaunay de  $V$ . A Figura 4.5 exemplifica uma partição de Voronoï, enquanto a Figura 4.6 sua respectiva triangulação de Delaunay e a Figura 4.7 mostra a superposição de ambas.

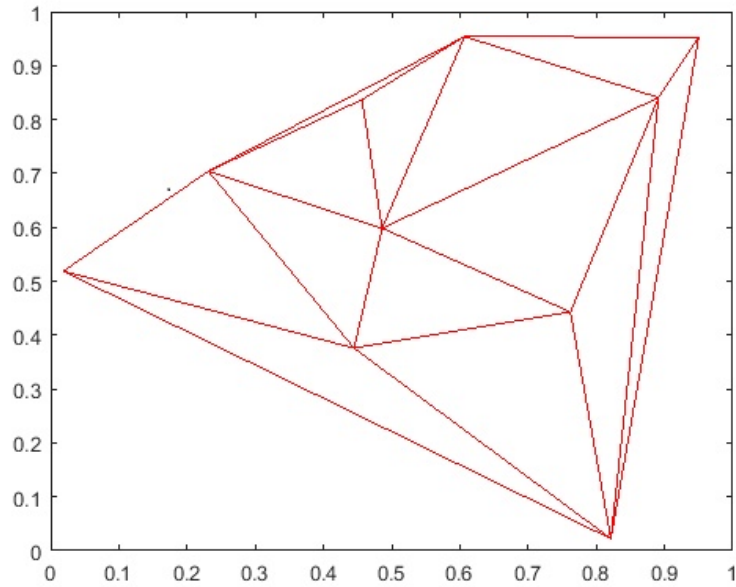


Figura 4.6: Triangulação de Delaunay de 10 pontos.

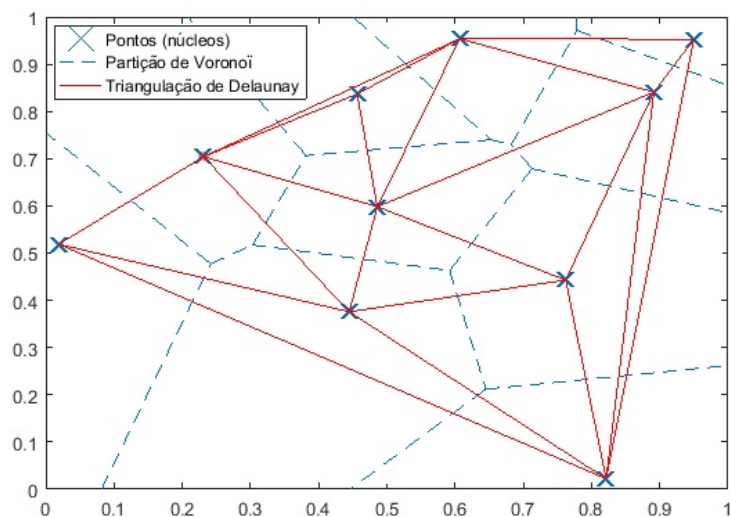


Figura 4.7: Sobreposição da Triangulação de Delaunay e da Partição de Voronoi de 10 pontos.

#### 4.5.2 Propriedades da partição de Voronoi

Algumas das principais propriedades da partição de Voronoi podem ser vistas em Oliveira (1995). Quatro delas são listadas sem demonstrações:

- O poliedro de Voronoi  $V_i$  é convexo.
- Uma face de um poliedro de Voronoi é compartilhada por apenas um outro poliedro.

- As faces do poliedro de Voronoï são convexas.
- Um ponto não pode se situar dentro de mais de um poliedro de Voronoï.

### 4.5.3 Propriedades da triangulação de Delaunay

A triangulação de Delaunay possui algumas propriedades e suas demonstrações estão em Lo (2015), Oliveira (1995), Cheng, Dey e Shewchuk (2013). As principais são listadas a seguir:

- Entre todas as possíveis triangulações de um conjunto de pontos no plano, a DT maximiza o ângulo mínimo e minimiza a circunferência máxima.
- Se todos os triângulos em uma triangulação são não-obtusos, é uma DT.
- Qualquer triangulação pode ser transformada em uma DT pela troca das diagonais de triângulos adjacentes.

É importante ressaltar a definição de um triângulo de Delaunay: um triângulo é de Delaunay se e somente se o círculo circunscrito não contiver nenhum outro ponto de  $\mathbf{V}$  em seu interior. Esse também é chamado de critério de Delaunay. Considerando que o círculo circunscrito possui centro  $Q_s$  e raio  $r_s$ , um dado ponto  $P$  não satisfaz o critério caso a distância entre  $P$  e o centro  $Q_s$  seja menor do que  $r_s$ :

$$\|P - Q_s\| < r_s \quad (4.2)$$

### 4.5.4 O contorno como restrição da triangulação de Delaunay

Destaca-se que o critério de Delaunay por si só não é um algoritmo de geração de malha - ele meramente provê uma regra para conectar um conjunto de pontos no espaço para criar uma triangulação. Conforme Lo (2015), apesar da formulação do critério de Delaunay ser de 1934, ela se tornou mais útil a partir do fim da década de 1970 com o desenvolvimento de algoritmos e com o rápido desenvolvimento do MEF, sendo que essas contribuições também são úteis para o MEC.

Mesmo nos casos em que o contorno do domínio é bem definido, faz-se necessário desenvolver uma técnica para determinar a quantidade e os locais de inserção dos pontos no domínio de interesse. Os pontos do contorno devem ser inseridos e conectados atendendo ao critério de Delaunay, fazendo com que seja criada uma triangulação dos nós do contorno. O próximo passo é inserir mais pontos na malha inicialmente grosseira do contorno, refinando, assim, os triângulos à medida que novos pontos são introduzidos, até que uma quantidade desejável de elementos seja formada, (BOROUCHAKI; GEORGE; LO, 1996) *apud* (LO, 2015).

É essencial que a triangulação contenha o contorno do domínio do problema, de modo a possibilitar a integridade do contorno apenas deletando os triângulos localizados fora do domínio. Nos processos de DT é comum que primeiro os nós (pontos) do contorno sejam particionados e triangularizados e só então sejam gerados os nós internos (do domínio). Esse processo, entretanto, pode fazer com que nem todos os nós do contorno sejam triangularizados. Por isso, é comum a divisão da triangulação em dois passos: o primeiro é particionar e triangularizar os pontos do contorno com um algoritmo comum de Delaunay de inserção de pontos, sem preocupação com a integridade do contorno; o segundo é recuperar a integridade do contorno (há algumas formas que serão explicadas na subseção 4.5.7). É possível recuperar uma fronteira perdida apenas fazendo uma troca de diagonais dos triângulos.

### 4.5.5 Cavidades

A partir da ideia de que nenhum ponto pode estar dentro da circunferência da triangulação, surge a definição de cavidade. Considere um conjunto  $V$  de pontos, um ponto  $P \notin V$  e  $B = \{S/S \in T, \|P - Q_s\| < r_s\}$ . Para esse caso,  $B$  é o conjunto dos simplexes para os quais o ponto  $P$  não satisfaz o critério de Delaunay. Ao se retirar a área  $B$  da área  $T$ , obtém-se uma região definida como cavidade, um conceito fundamental para a implementação de um algoritmo de inserção de novos pontos.

Visando a permitir que  $P$  seja integrado a  $T$ , é possível a implementação de um algoritmo de inserção de pontos. Para que isso ocorra, é fundamental saber se o ponto  $P$  é visível - pela teoria dos Grafos - em toda a cavidade. De acordo com essa teoria, para dois pontos serem mutuamente visíveis, é necessário que haja a possibilidade de uni-los sem a intercepção de qualquer lado existente. Infere-se que  $P$  será visível em toda a cavidade caso seja visível de todos os pontos do contorno da cavidade - Figura 4.8.

Seja  $T$  uma triangulação de Delaunay de um conjunto  $V$  de pontos no plano, a introdução de um ponto  $P \notin V$  e a remoção do conjunto  $B$  (triângulos que não satisfazem o critério de Delaunay em relação ao ponto  $P$ ) dá origem a uma cavidade com todos os pontos visíveis a partir de  $P$ . Além disso, a triangulação de  $T - B$  (formada pela união do ponto  $P$  a cada um dos pontos dos lados da fronteira da cavidade) é uma triangulação de Delaunay. As demonstrações dessas propriedades encontram-se em Oliveira (1995).

Segundo Oliveira (1995), criar uma triangulação estritamente de Delaunay é uma tarefa impossível em casos práticos. Algumas razões para isso são: a) A precisão finita dos cálculos computacionais, ocasionando na impossibilidade de determinar com exatidão o conjunto  $B$ ; b) A correção de eventuais problemas que surgem durante a geração da malha, como existência de elementos achatados, espaços vazios e interpenetração de elementos.

Em razão dos motivos elencados no parágrafo anterior, considere uma triangulação geral (não necessariamente de Delaunay) e introduza um novo ponto  $P \notin T$ , que gera o conjunto  $B$ . A partir da retirada dos elementos de  $B$  da triangulação  $T$ , a cavidade gerada pode ser

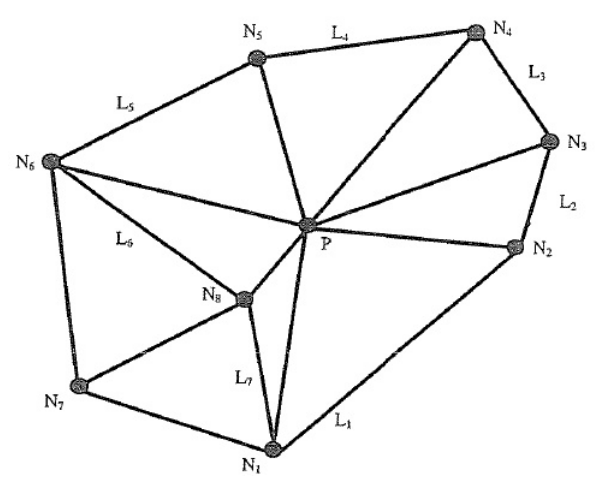
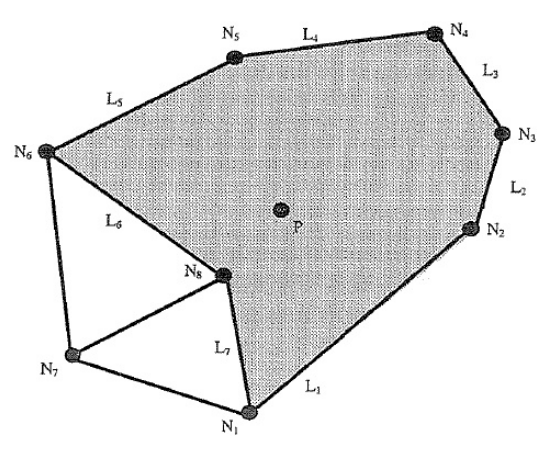
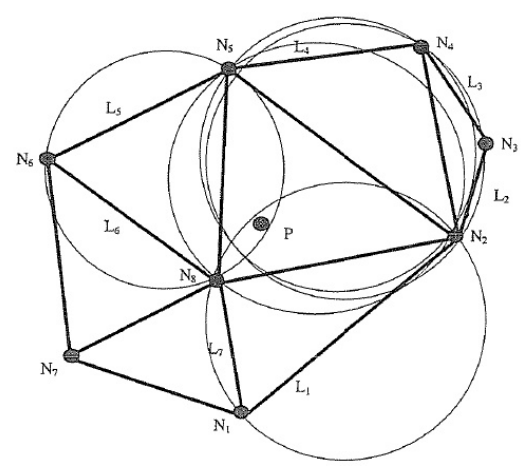
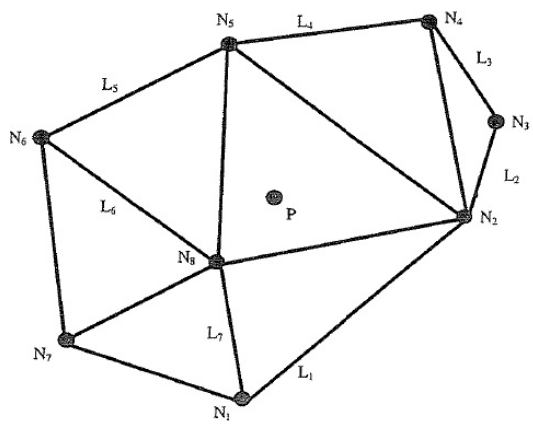


Figura 4.8: Inserção de ponto  $P$ . Fonte: Oliveira (1995).

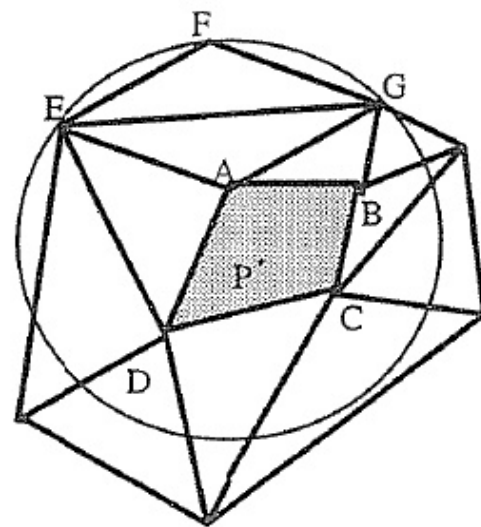
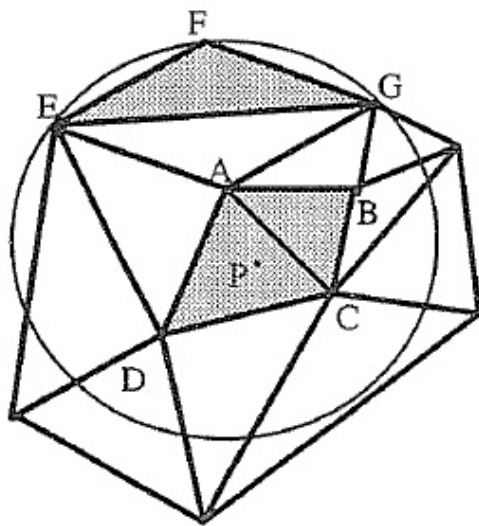


Figura 4.9: Cavityde generalizada (a) e componente principal (b). Fonte: Oliveira (1995).

desconexa. Para o caso onde há regiões desconexas da cavityde, há o componente principal da cavityde e também a cavityde generalizada. Esta é a união das regiões obtidas pela remoção de todas as áreas da cavityde, inclusive as desconexas. Já aquele pode ser definido como a maior região conexa que contém o ponto  $P$ . Exemplos de cavitydes conexas e desconexas podem ser vistas na Figura 4.9.

Vale destacar que é possível obter uma nova triangulação (não necessariamente de Delaunay) contendo  $P$  a partir do componente principal da cavityde. Isso deve-se ao fato de que a remoção de triângulos de uma cavityde para malhas em 2D particiona a cavityde em regiões disjuntas.

#### 4.5.6 Inserção de pontos

O processo de inserção de um novo ponto em uma malha preexistente faz com que ele faça parte da malha, depois de alguns procedimentos como eliminar e criar elementos. Há



um algoritmo para a inserção de novos pontos em Lo (2015). Seja  $S$  um conjunto de  $n$  pontos diferentes em um plano. Inicialmente constrói-se um domínio retangular composto por dois triângulos  $T_0$  grande o suficiente para conter todos os pontos de  $S$ . Em seguida, deve-se inserir todos os pontos, um por um, em  $S$ , e cada ciclo de inserção de pontos pode ser dividido em três etapas:

1. Para um ponto recentemente inserido, identificar todos os triângulos cujas circunferências contenham o ponto. A cavidade formada pela remoção desses triângulos forma um polígono com forma de estrela, que será denominado como núcleo.
2. Devido à precisão aritmética finita, as arestas do contorno do polígono devem ser verificadas pelo teste de visibilidade e corrigidas antes de serem conectadas com os pontos inseridos a fim de formar os triângulos.
3. A triangularização da cavidade deve ser trivial. Entretanto, a relação de adjacência entre os triângulos novos e antigos nas arestas do contorno deve ser estabelecida.

#### 4.5.7 Recuperação da fronteira do domínio

Conforme dito anteriormente, uma das desvantagens da DT é que ela não preserva a fronteira do domínio ou as interfaces entre subdomínios. Deve-se ao fato de que o algoritmo de geração de malha trabalha com os pontos que constituem os triângulos em vez dos triângulos propriamente ditos. O passo após a inserção de todos os pontos é, então, recuperar a fronteira perdida. Observa-se no exemplo (Figura 4.10) que a malha gerada cobre toda a envoltória, mas atravessa a fronteira definida pelas arestas  $AB$ ,  $BD$  e  $CD$ . Conforme Oliveira (1995), na solução do problema deve-se construir uma lista de todos os lados gerados e compará-los com a lista de lados especificados, de forma a determinar quais lados estão ausentes e, em seguida, proceder com a regeneração dos lados ausentes.

De acordo com Lo (2015), o procedimento para a geração de malha sobre um domínio bem definido pode ser dividido em três passos distintos: i) inserção de pontos do contorno; ii) inserção de pontos interiores; iii) recuperação da fronteira (do contorno). Esse problema é conhecido como triangulação de Delaunay restringida (*constrained*). A conformidade de todo o contorno, entretanto, só é garantida para os casos 2D, para as quais o contorno consiste em uma aresta.

Há mais de um processo de recuperação da fronteira (OLIVEIRA, 1995), mas o explicado em detalhes neste trabalho consiste em realizar modificações topológicas e geométricas da malha da envoltória, que fornece como resultado uma malha adaptada geometricamente de forma exata à fronteira do domínio. Esse processo permite que um domínio seja particionado em subdomínios, para os quais se geram malhas de forma independente. É importante que as interfaces entre os subdomínios sejam discretizadas pelas mesmas faces, o que permite que elas se unam, formando uma malha geral para todo o domínio.

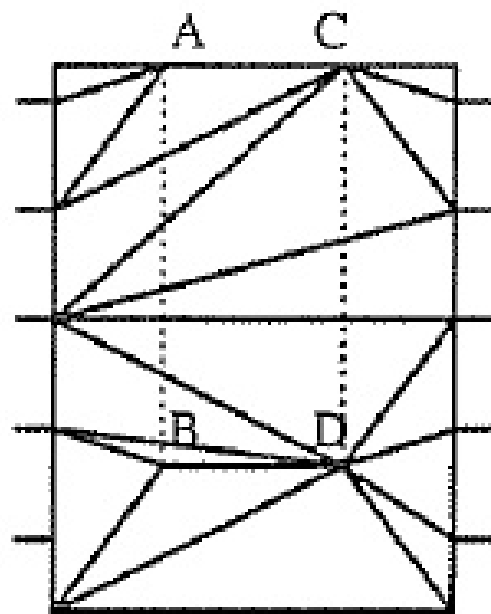
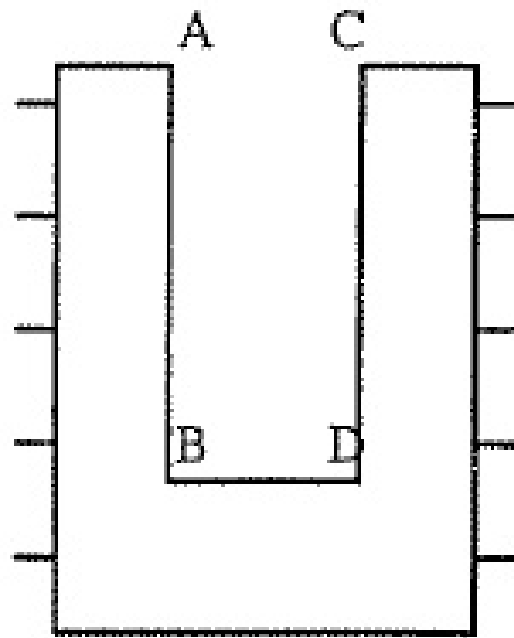


Figura 4.10: Domínio inicial (a) e perda da fronteira (b). Fonte: Oliveira (1995).

Ressalta-se que a recuperação de uma aresta ausente para o caso 2D pode ser feita por sucessivas transformações. Por exemplo, seja  $T$  a triangulação de Delaunay de um conjunto de pontos no plano, considere que  $AB$  seja um lado genérico ausente obtido após a elaboração da lista de todos os lados ausentes. É possível associar a  $AB$  uma malha  $T_{AB}$  formada por todos os triângulos interceptados por  $AB$  e, em seguida, gerar uma malha equivalente  $T'_{AB}$  que contém a aresta  $AB$ . A Figura 4.11 mostra a malha inicial em (a) e o passo-a-passo das trocas de diagonais, até chegar à recuperação de uma aresta ausente em (b) a (e). Quando o método é aplicado a todos os lados ausentes, está finalizada a recuperação da fronteira.

## 4.5.8 Geração de malha sobre superfícies curvas

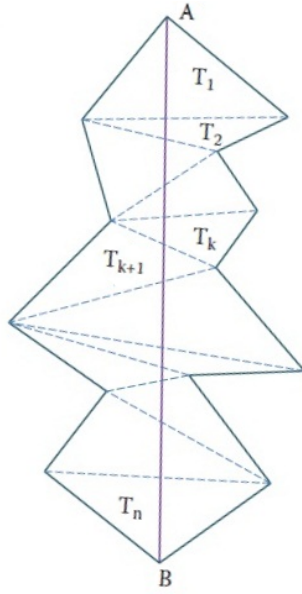
Há uma grande demanda por algoritmos que façam a geração de malha sobre superfícies curvas arbitrárias, com elementos de variados tamanhos e formas. É comum que os contornos de um objeto sejam representados por superfícies NURBS geradas em algum programa CAD. A discretização do sólido em ambiente CAD é geralmente rápida e prática e também pode ser utilizada para as análises de MEF e MEC, servindo como dado de entrada para a geração de malha. Há duas formas principais de representar uma superfície para fins de visualização e computação. Conforme Lo (2015), uma forma é utilizar superfícies analíticas com funções como *Coon*, *B-splines* ou superfícies NURBS - Frey e George (2000). O outro método é usar uma estrutura com dados discretos como facetas quadrilaterais e triangulares (LOHNER, 1996).

Para aplicações na engenharia, é comum utilizar superfícies com dados discretos, conforme Shostko, Löhner e Sandberg (1999). Ressalta-se que para cada método de representação da superfície, há variados tipos de processos de geração de malha. É usual fazer uma classificação para dividir os processos em mapeamento paramétrico ou geração de malha diretamente sobre a superfície 3D.

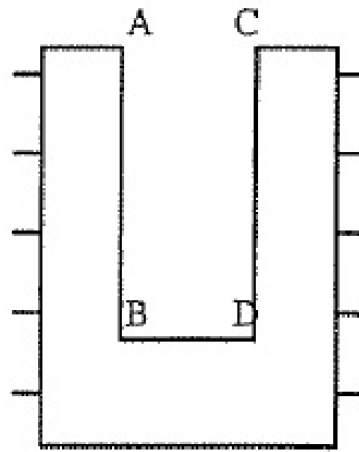
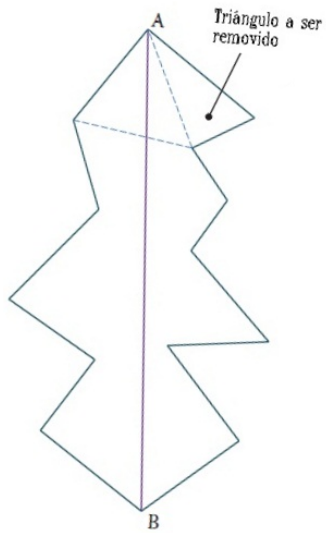
### 4.5.8.1 Geração de malha diretamente sobre a superfície 3D

Essa técnica consiste em formar os elementos diretamente na superfície sem a necessidade de uma representação paramétrica da geometria subjacente. Caso não seja possível utilizar a geração de malha pelo método da parametrização, usa-se a geração diretamente sobre a superfície 3D. Há um esquema, conforme Lan e Lo (1996) e Lo e Lau (1998), de avanço de fronteira (ADF) de malhas triangulares em superfícies curvas no espaço. Para esse método é fundamental que as normais e tangentes da superfície sejam computadas para determinar a direção da frente de geração. Faz-se necessário uma quantidade de projeções de superfície para trazer os novos nós gerados para a superfície, (LO, 2015).

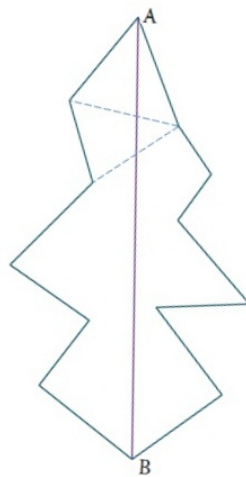
Além disso, é útil para os casos de regeneração da triangulação da malha em uma superfície existente para a qual não há parametrização disponível imediatamente ou para a geração de malhas híbridas com superfície 3D e elementos sólidos. A Figura 4.12 ilustra malhas de



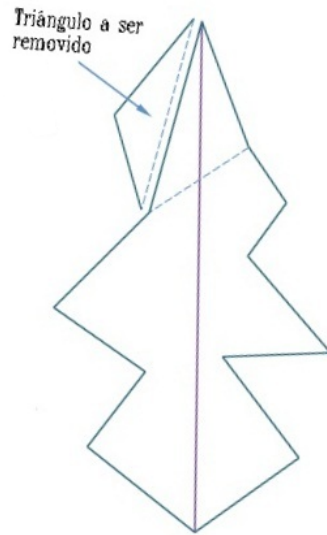
(a)



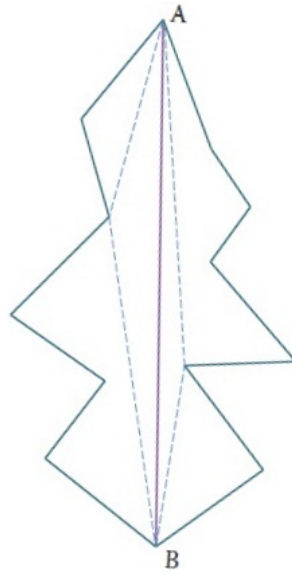
(b)



(c)



(d)



(e)

Figura 4.11: Recuperação de uma aresta ausente em 2D. Fonte: Lo (2015) (modificado).

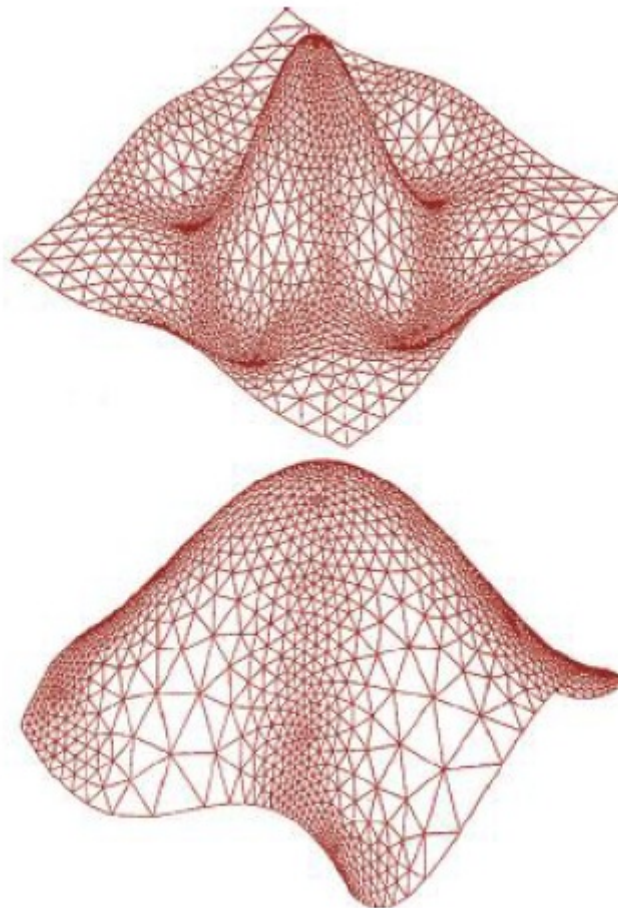


Figura 4.12: Malhas triangulares geradas diretamente no espaço. Fonte: Lo (2015).

superfície geradas diretamente no espaço 3D.

Outra possibilidade é a geração de malhas quadrilaterais diretamente no espaço 3D, conforme Figura 4.13. É fundamental que se atente para que os elementos não se sobreponham e nem se interseccionem. Destaca-se que as malhas triangulares são mais práticas de serem geradas em comparação às quadrilaterais. Estas podem ser geradas convenientemente a partir de malhas triangulares, formando-se os quadriláteros a partir de triângulos adjacentes.

#### 4.5.8.2 Geração de malha com mapeamento paramétrico

Essa abordagem utiliza uma função analítica com duas variáveis para discretizar a superfície, de tal forma que cada ponto na superfície 3D é mapeado em um espaço paramétrico 2D. Todo o processo de geração da malha é feito no plano por um gerador de malhas 2D. Em seguida, obtém-se a superfície final por meio de uma transformação da malha gerada no plano paramétrico para o espaço, conforme Figuras 4.15 e 4.16. Merece destaque esse tipo de abordagem, pois é a utilizada neste trabalho, como pode ser visto no capítulo de resultados. A Figura 4.14 ilustra o processo de transferência dos pontos de um plano com coordenadas  $(u, v)$  para  $(x, y, z)$ , formando uma superfície  $S$ .

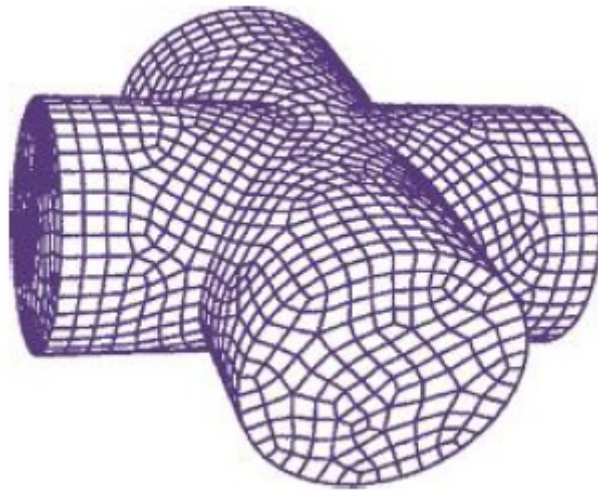


Figura 4.13: Malha quadrilateral gerada diretamente no espaço. Fonte: Lo (2015).

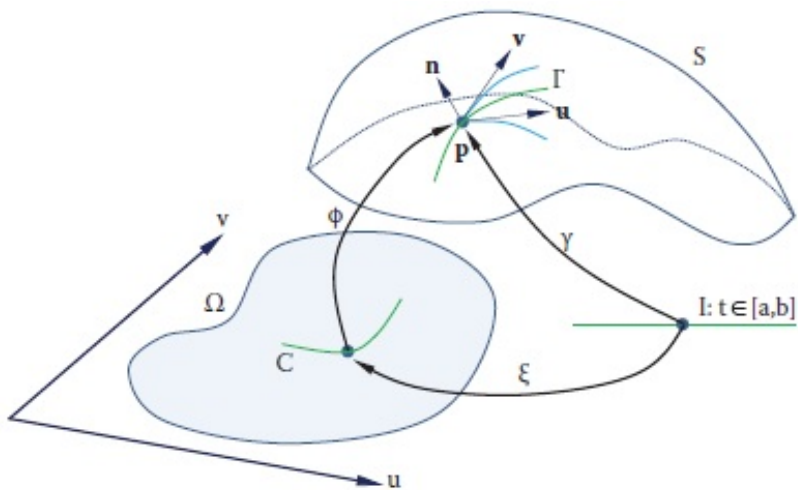


Figura 4.14: Superfície curva  $S$  produzida por um processo de mapeamento paramétrico. Fonte: Lo (2015).



Figura 4.15: Malha estruturada paramétrica. Fonte: Lo (2015).

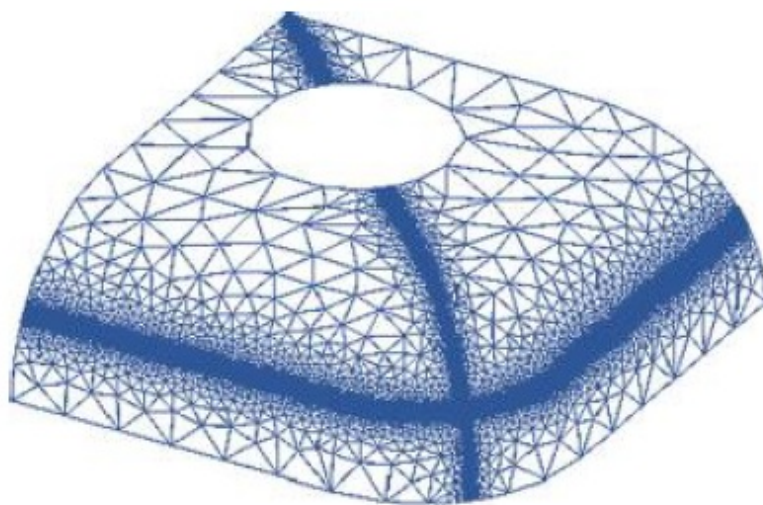


Figura 4.16: Malha não-estruturada paramétrica. Fonte: Lo (2015).



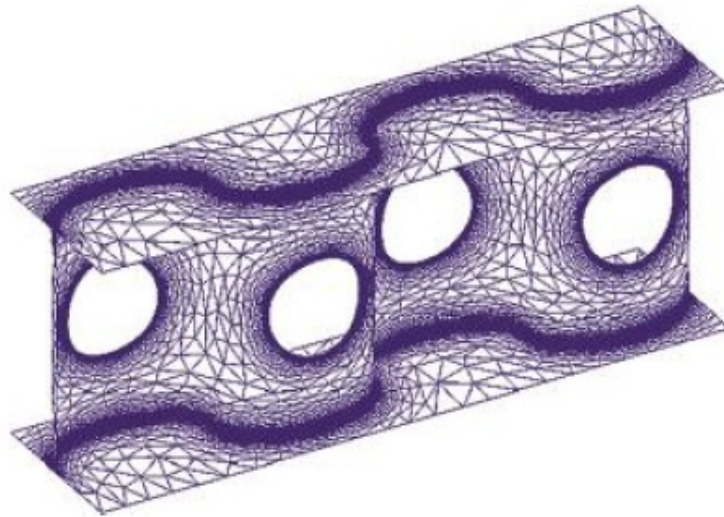


Figura 4.17: Malha anisotrópica paramétrica. Fonte: Lo (2015).

Conforme Lo (2015), esse método produz malhas de qualidade razoavelmente boa para superfícies simples que sejam suficientemente suaves com respeito a um domínio plano. Entretanto, para superfícies curvas mais complexas, é comum que as malhas sejam de qualidade baixa devido à distorções nos elementos, induzidas pelo mapeamento (BOROUCHAKI et al., 2005). Conhecendo essa limitação, há uma versão melhorada do método, que consiste em primeiro decompor as superfícies complexas em uma união de outras mais simples, e aí então gerar malhas em cada sub-superfície.

Apesar de ser adequada para alguns casos, essa versão melhorada também possui suas limitações, pois decompor a superfície em sub-superfícies mais simples e suaves pode ser tão complexo quanto gerar a malha diretamente na superfície original. Além disso, cria-se uma nova preocupação - cuidado especial deve ser tomado com os contornos entre as sub-superfícies para garantir a compatibilidade e o comportamento igual ao original.

Outro cuidado que deve ser tomado, conforme Remacle et al. (2010), é que se deve controlar tanto o formato quanto o tamanho dos elementos no plano paramétrico, a fim de garantir uma malha que satisfaça as especificações do usuário. É importante ter em mente que para gerar uma malha em uma superfície curva que atenda os espaçamentos nodais, deve-se modificar o algoritmo de geração de malha com o objetivo de que os elementos alongados e/ou anisotrópicos gerados no plano 2D paramétrico sejam convertidos em elementos com bom formato no espaço 3D.

# Capítulo 5

## Descrição dos pacotes usados

### 5.1 Introdução

Este capítulo faz uma descrição do programa implementado ao longo deste trabalho, detalhando o fluxograma do algoritmo e também os pacotes usados na implementação da formulação do método dos elementos de contorno aplicado à análise de problemas de condução calor em sólidos.

A análise de um problema tridimensional, seja ele térmico, elástico, eletromagnético ou de qualquer outro tipo, começa com o desenho do sólido. Embora problemas com geometrias simples possam ser modelados facilmente utilizando as NURBS, na construção de sólidos com geometria mais complexas, o uso de um programa CAD é fortemente recomendado. O programa implementado nesse trabalho pode ser melhor entendido ao se consultar o fluxograma na Figura 5.1.

O primeiro passo consiste em modelar certas geometrias em um software CAD que exporte em formato IGES, como o SolidWorks (SYSTEMES, 2017b) ou o FreeCad (HAVRE, 2016; COLLETTE; FALCK, 2012). Em seguida, por meio do pacote *IGES toolbox* (BERGSTROM, 2017), o Matlab (ou o Octave) é capaz de ler os parâmetros do arquivo IGES. Com isso, é possível utilizar o pacote *MESH2D* (ENGWIRDA, 2017), responsável por gerar a malha de superfície no modelo sólido. Conforme descrito no capítulo 2, no pacote IGES, cada superfície do sólido é representado por uma superfície aparada, ou seja, a superfície NURBS sem apartamento e as respectivas curvas de apartamento. Tanto as superfícies NURBS quanto as curvas de apartamento são descritas no plano paramétrico  $(u, v)$ . O pacote *MESH2D* é usado para gerar a malha no plano  $(u, v)$ . Então, usando novamente o pacote *IGES toolbox*, a malha gerada no plano é mapeada na superfície que encontra-se no espaço  $(x_1, x_2, x_3)$ .

Após o passo anterior, calcula-se os vetores normais unitários em cada elemento (uma vez que os elementos são triângulos planos, o vetor normal unitário é constante ao longo de todo o elemento). Entretanto, os programas CAD não se preocupam em ter todas as normais apontando para fora do domínio do modelo. Ou seja, uma verificação de quais superfícies

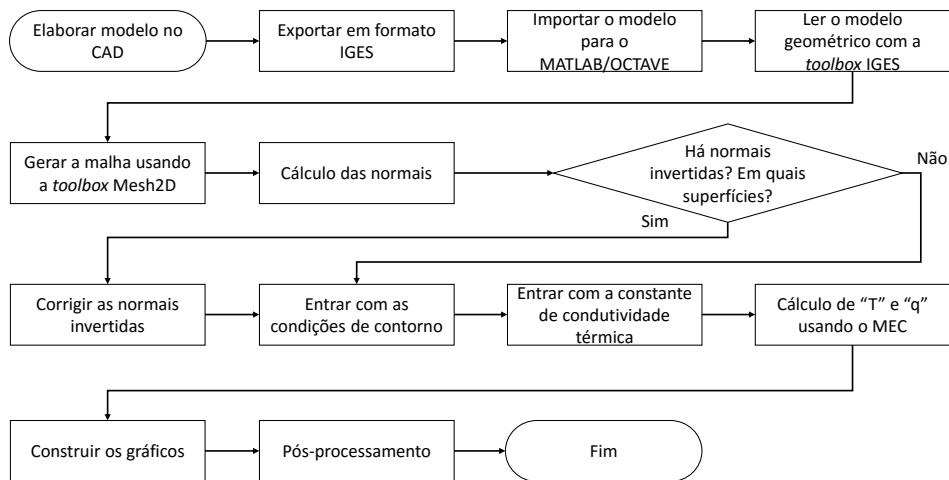


Figura 5.1: Fluxograma de processos do trabalho.

Tabela 5.1: Dados de entrada

Dado	
Arquivo IGES	O arquivo gerado pelo <i>software</i> CAD.
<i>showlines</i>	Opção que permite plotar apenas as linhas associadas a superfícies ou todas as linhas do arquivo.

têm normais invertidas faz-se necessária. Para contornar este problema, foi implementada uma função que mostra a direção que os vetores normais unitários apontam. As superfícies que tem os vetores normais invertidos tem seus nós reordenados para que a direção dos vetores normais sejam corrigidas. Então, as matrizes de influência do método dos elementos de contorno são montadas, as condições de contorno são aplicadas, as variáveis desconhecidas são calculadas e, por fim, é feito o pós-processamento, onde um mapa de temperatura é mostrado nas superfícies do sólido.

## 5.2 IGES Toolbox

Este pacote é utilizado para a análise de arquivos IGES no Matlab/Octave, que extrai os dados de um arquivo *.igs* por meio da função *iges2matlab*. Ressalta-se que a *toolbox* não lê todas as entidades do arquivo IGES, mas as mais usuais. Além disso, é prática e de simples utilização.

Os dados de entrada e de saída são explicados nas Tabelas 5.1 e 5.2, respectivamente.

Há várias funções dentro da *toolbox*, como:

- *iges2matlab*: Extrai os parâmetros de dados do arquivo IGES para uma *cell array*.
- *plotIGES*: Plota o modelo IGES vindo do CAD.

Tabela 5.2: Dados de saída

Dado	
<i>ParameterData</i>	Matriz com os dados do arquivo IGES e algumas informações adicionais.
<i>EntityType</i>	Vetor com entidades no arquivo IGES convertido para Matlab.
<i>numEntityType</i>	Vetor com o número de entidades pertencentes ao <i>EntityType</i> .
<i>unknownEntityType</i>	Vetor com entidades desconhecidas pelo iges2matlab.
<i>numunknownEntityType</i>	Vetor com o número de entidades pertencentes ao <i>unknownEntityType</i> .

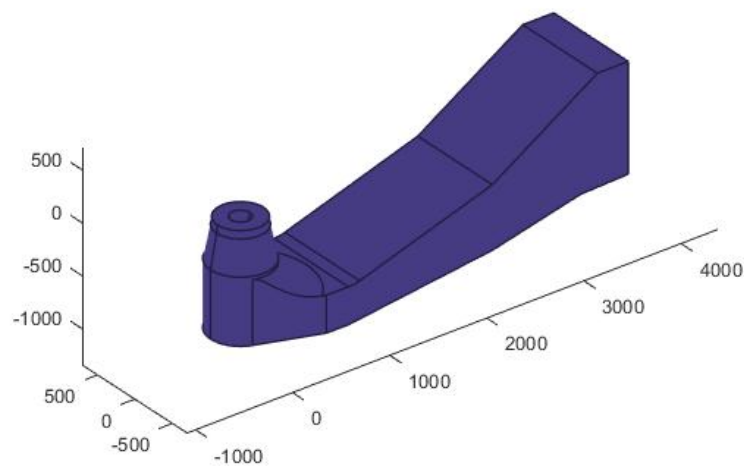


Figura 5.2: Exemplo de arquivo IGES lido no Matlab pelo pacote.

- `plotIGESentity`: Plota dada entidade do modelo IGES vindo do CAD.
- `plotNURBS`: Plota uma entidade NURBS (como superfície ou curva).

Essas são apenas exemplos do que contém o pacote e mais informações podem ser obtidas lendo-se a documentação disponível em Bergström (2017), Gonçalves (2016). A Figura 5.2 demonstra um exemplo de utilização que está disponível com o pacote.

### 5.3 Mesh2D Toolbox

O *Mesh2D*, por sua vez, é um pacote que proporciona a geração automática de malhas triangulares não-estruturadas para geometrias 2D arbitrárias. Ele tem uma característica interessante que utiliza algoritmos iterativos para otimizar a topologia da malha e a posição dos nós, com isso atingindo triangulações de alta qualidade. Cabe dizer que o polígono

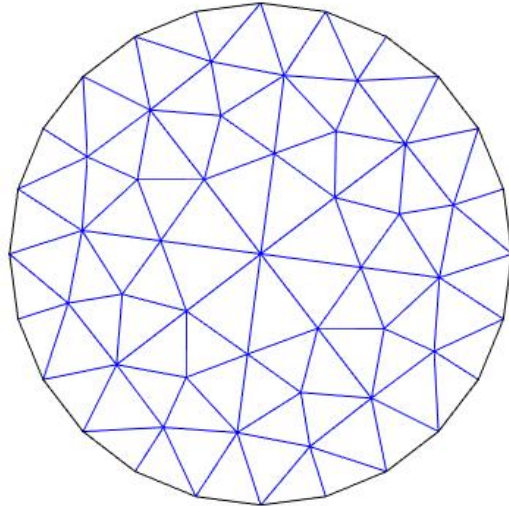


Figura 5.3: Exemplo 1 do pacote Mesh2D.

utilizado por este pacote pode conter uma quantidade arbitrária de furos.

Apesar de ser automatizado, este pacote permite ao usuário a configuração de dimensão dos elementos, garantindo diferentes resoluções de malha no domínio. As Figuras 5.3 e 5.4 mostram exemplos de malhas geradas automaticamente pelo pacote, ou seja, sem definição de função pelo usuário.

Conforme dito anteriormente, o pacote Mesh 2D permite que o usuário determine arbitrariamente uma função para o refinamento da malha. A Figura 5.5 mostra, respectivamente, a malha gerada e a função utilizada para o refinamento.

$$H = 0,01 + 0,1\sqrt{(x - 0,25)^2 + (y - 0,75)^2}$$

Percebe-se que as curvas de nível produzidas para o refinamento variam de acordo com o valor da função. Quanto menor o valor da função, mais próximas as curvas de nível e maior o refinamento da malha, ou seja, maior o número de elementos. À medida que o valor da função cresce, as curvas de nível ficam com espaçamento maior e a malha, mais grosseira.

#### **Os dados de entrada são:**

*node*: A geometria dos nós especificados em ordem consecutiva, a fim de que o nó 2 seja conectado ao nó 1 e assim sucessivamente.

*edge*: Define a conectividade entre os nós do contorno como uma lista de arestas.

*hdata*: Com esses dados, há a possibilidade de o usuário definir informações acerca do tamanho do elemento.

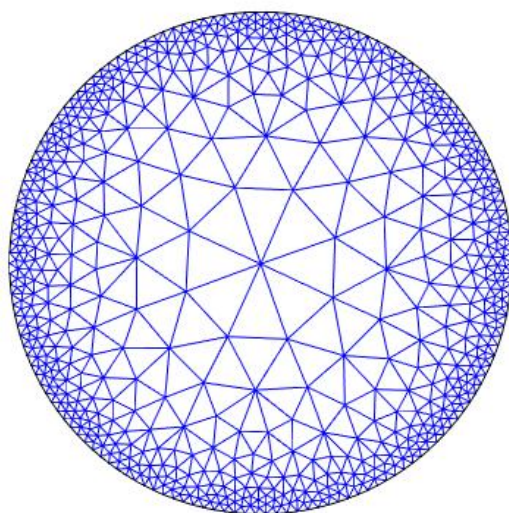


Figura 5.4: Exemplo 2 do pacote Mesh2D.

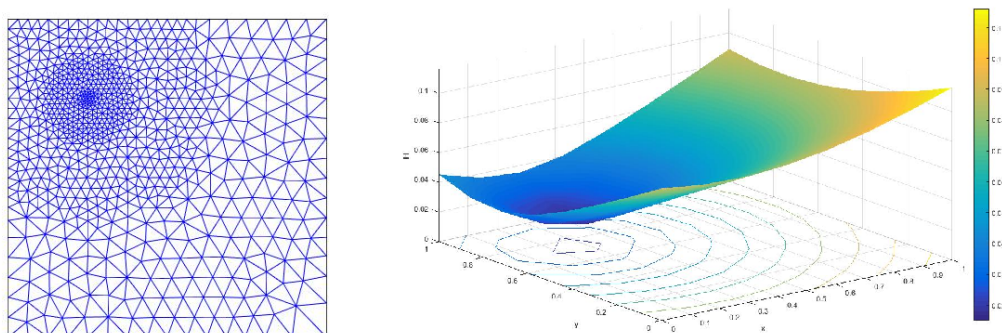


Figura 5.5: Malha gerada (a) pela função 5.3 (b).

*hdata.hmax*: Tamanho máximo global permitido aos elementos.

*hdata.edgeh*: Tamanho do elemento em arestas especificadas.

*hdata.fun*: Função de tamanho de elemento definida pelo usuário.

*hdata.args*: Argumentos adicionais para *hdata.fun*.

*options*: Permite ajustes em alguns parâmetros utilizados no *solver*.

*options.mlim*: A tolerância da convergência, o erro aceitável, em percentual. A diferença percentual máxima no tamanho de uma mesma aresta entre as sucessivas iterações.

*options.maxit*: A quantidade máxima de iterações.

*options.dhmax*: O gradiente máximo aceitável na função de tamanho de elemento.

*options.output*: Exibe a malha e as estatísticas após o término.

É possível o usuário utilizar sua própria função para o tamanho dos elementos, desde que ela aceite entradas vetorizadas de forma  $H = FUN(X, Y, ARGS\{:\})$ , onde  $X$  e  $Y$  são as coordenadas nas quais o tamanho do elemento será avaliado e  $ARGS$  são argumentos adicionais como descritos em *hdata.args*. Mesmo que haja uma função definida pelo usuário (ela é opcional), uma função automática de tamanho de elemento sempre é gerada para garantir que a geometria seja adequadamente resolvida. A função média para o tamanho do elemento é o mínimo entre as funções definida pelo usuário e a automática.

**Os dados de saída são:**

- As coordenadas  $(x_1, x_2)$  nodais dos pontos (vértices dos triângulos gerados).
- Os triângulos que é uma matriz de 3 colunas contendo a conectividade com uma ordenação nodal em sentido anti-horário.

Ressalta-se que um tamanho de elemento é gerado automaticamente baseado na complexidade da geometria e geralmente produz malhas com a menor quantidade possível de triângulos.

### 5.3.1 Transferindo os pontos da malha gerada no plano 2D para o espaço 3D

Após a geração da malha no plano com a *toolbox* Mesh2D, é necessário transferi-la para o espaço, já que se pretende trabalhar com sólidos. O processo é feito por meio da *function* Discretiza 3D (disponível em apêndice). A ideia é utilizar uma mudança de sistema de coordenadas, passando de  $(u, v)$  para  $(x_1, x_2, x_3)$ . No processo são parametrizadas as coordenadas em 2D para gerar uma malha em 3D. É possível obter uma superfície NURBS a partir do produto tensorial entre curvas NURBS e os parâmetros independentes  $(u, v)$ .

$$S = \sum_{i=1}^k \sum_{j=1}^l R_{i,j}(u, v) B_{i,j} \tag{5.1}$$

# Capítulo 6

## Resultados

### 6.1 Mancal

Um dos exemplos gerados nesse trabalho foi um mancal mostrado na Figura 6.1 e desenhado usando o programa comercial Solidworks (SYSTEMES, 2017b) e exportado em formato IGES. Mancais são peças amplamente utilizadas em aplicações de engenharia mecânica e servem para apoiar um eixo. Por causa do atrito entre as partes móveis, há geração de calor nas peças (tanto no mancal quanto no eixo). Esse exemplo utilizou um mancal modelado por 16 superfícies. Ressalta-se que a parte curva do furo do mancal é formada por duas superfícies. Algumas informações sobre o sólido podem ser vistas nas Tabelas 6.1 e 6.2.

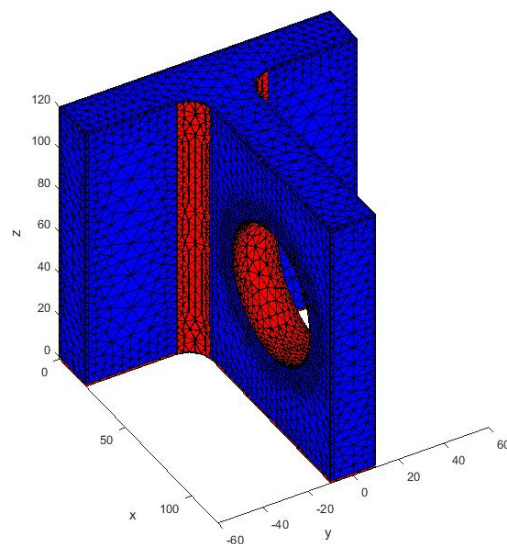


Figura 6.1: Mancal e superfícies.



Tabela 6.1: Entidades no mancal.

#	Número da entidade	Nome da entidade	Quantidade
1	128	B-NURBS SRF	16
2	126	B-NURBS CRV	104
3	110	LINE	80
4	102	COMPOSITE CRV	44
5	142	CRV ON A PARAMETRIC SURFACE	20
6	144	TRIMMED SURFACE	16
7	124	TRANSFORMATION MATRIX	18
8	314	COLOR	1

Tabela 6.2: Iterações e convergência do mancal.

Iteração	Convergência (%)
1	2,0
2	1,0
3	32,3
4	15,9
5	63,1
6	83,6
7	100,0

Vale discorrer brevemente sobre os *inputs* utilizados nas condições de contorno (CDC) do sólido. Para a resolução do problema, as CDC devem ser inseridas no formato de uma matriz, da seguinte forma:

$cdcsup = [\text{Número da superfície, tipo da CDC, valor da CDC}]$

Os tipos possíveis de CDC são:

- 1: Temperatura conhecida.
- 2: Fluxo conhecido.

Todas as superfícies que forem omitidas serão assumidas como tipo 2 (fluxo conhecido) e valor zero. Foram utilizados os seguintes dados para as condições de contorno do mancal:

$$cdcsup = \begin{bmatrix} 1 & 1 & 10 \\ 4 & 1 & 0 \\ 16 & 1 & 10 \end{bmatrix}$$

ou seja, nas superfícies 1 e 16 a temperatura é conhecida e tem valor igual a 10 e na superfície 4 a temperatura também é conhecida e tem valor igual a zero. Nas demais superfícies o fluxo é conhecido e tem valor zero (condição de contorno quando nada é especificado).

A Figura 6.1 mostra o sólido após a geração da malha. As superfícies em azul estão com as normais invertidas, enquanto as em vermelho estão com as normais no sentido correto. Após a correção das direções das normais, o resultado pode ser visto na Figura 6.5 (não há mais superfícies em azul).

Ao longo do processo, a malha é gerada conforme os seguintes passos, para cada superfície - apenas a superfície 2 é mostrada como exemplo. Primeiramente, geram-se os pontos do contorno no espaço paramétrico  $(u, v)$ , como na Figura 6.2. Em seguida, é a vez dos pontos internos serem gerados - Figura 6.3 - e triangularizados usando o método da triangulação de Delaunay restrita (mantendo as arestas que formam o contorno). Este processo faz primeiro a triangulação do contorno, sendo então inseridos pontos internos para melhorar a qualidade da malha. A Tabela 6.2 mostra o percentual de elementos que atingiram a qualidade desejada em cada um dos passos. Por fim, a malha gerada no plano é transferida para o espaço 3D - Figura 6.4.

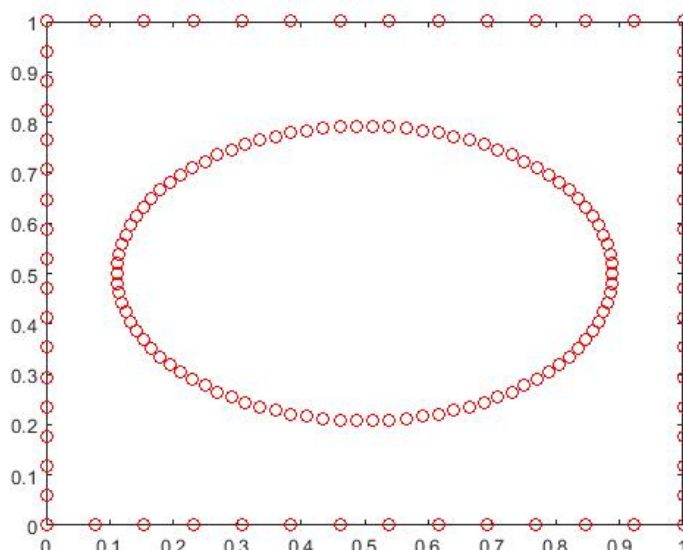


Figura 6.2: Geração de pontos do contorno da superfície 2.

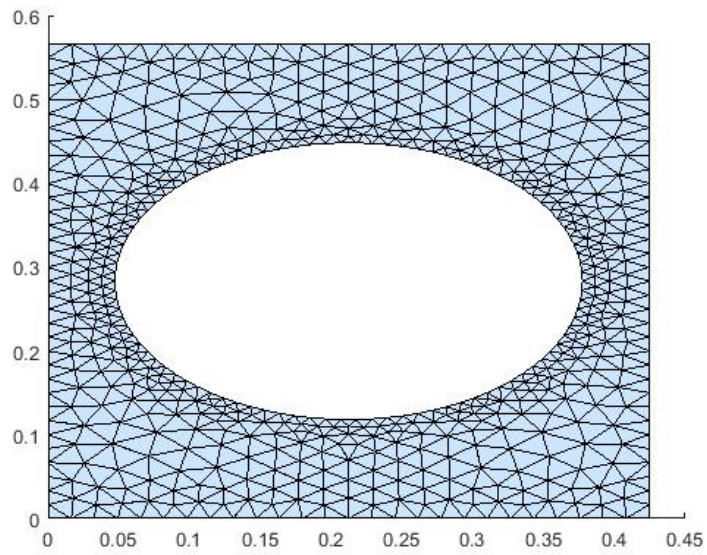


Figura 6.3: Geração de pontos internos da superfície 2.

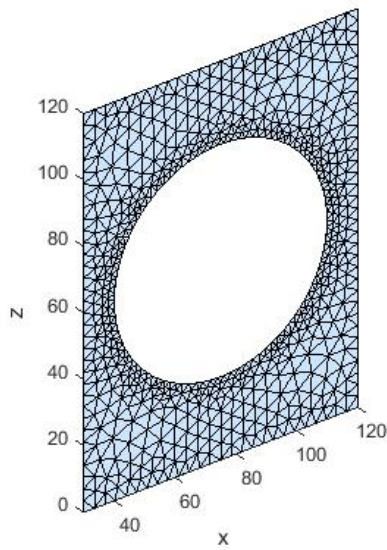


Figura 6.4: Transferência da malha do plano 2D para o espaço 3D da superfície 2.

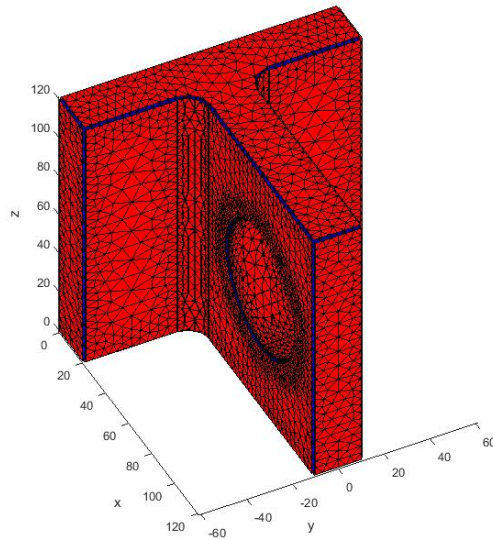


Figura 6.5: Manca com as normais corrigidas.

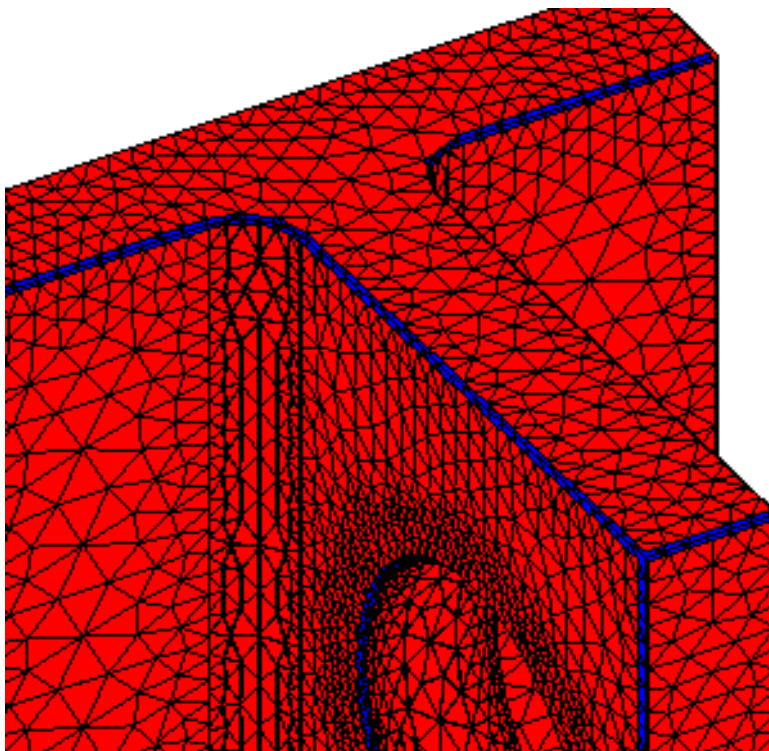


Figura 6.6: Detalhe para visualização da malha.

A Figura 6.6 mostra um detalhe para melhor visualização da malha.

Finalmente, o programa entra no módulo do MEC, onde as matrizes de influência  $H$  e  $G$  são montadas, as condições de contorno são aplicadas, as variáveis desconhecidas são calculadas e o fluxo e a temperatura são separados. A Figura 6.7 mostra o mapa de temperatura para o manca.

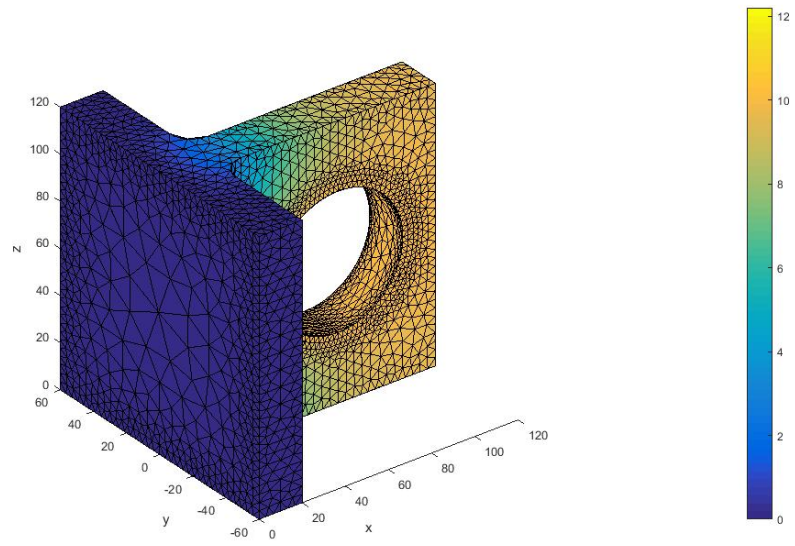


Figura 6.7: Temperaturas no mancal - mapa de cores.

## 6.2 Placa com Furo

O segundo exemplo gerado foi uma placa com um furo central - Figura 6.8. Essa geometria foi gerada pelo FreeCad (HAVRE, 2016) e exportado em formato IGES. Nesse exemplo, a placa possui um total de 8 superfícies, que na Figura 6.9 estão com as normais todas corrigidas - em cor vermelha. As informações sobre as entidades da placa estão nas Tabelas 6.3 e 6.4.

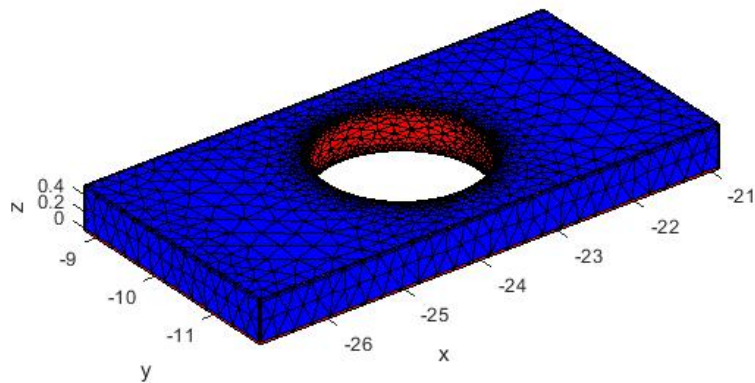


Figura 6.8: Placa com furo e superfícies.

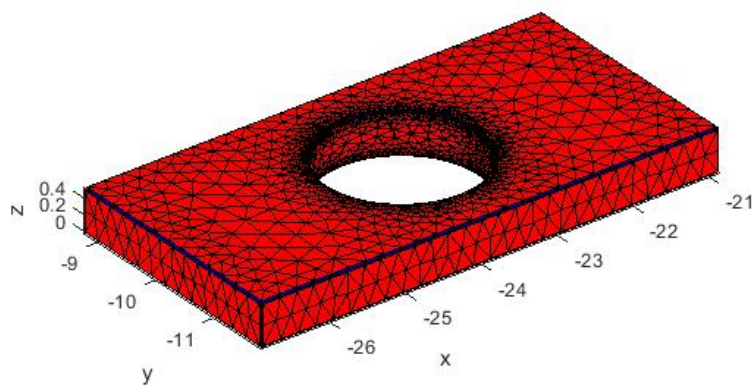


Figura 6.9: Placa com as normais corrigidas.

Tabela 6.3: Entidades no placa com furo.

#	Número da entidade	Nome da entidade	Quantidade
1	128	B-NURBS SRF	8
2	126	B-NURBS CRV	44
3	110	LINE	40
4	102	COMPOSITE CRV	22
5	142	CRV ON A PARAMETRIC SURFACE	10
6	144	TRIMMED SURFACE	8
7	124	TRANSFORMATION MATRIX	2
8	314	COLOR	1

Tabela 6.4: Iterações e convergência da placa com furo.

Iteração	Convergência (%)
1	2,0
2	0,2
3	7,1
4	33,2
5	68,1
6	82,0
7	100,0

A forma de entrada de dados é a mesma para o exemplo anterior. As condições de contorno foram as seguintes:

$$cdcsup = \begin{bmatrix} 2 & 1 & 30 \\ 4 & 1 & 10 \end{bmatrix}$$

O processo de geração de malha é análogo ao do mancal e demonstrado nas Figuras 6.10, 6.11 e 6.12 para a superfície 6.

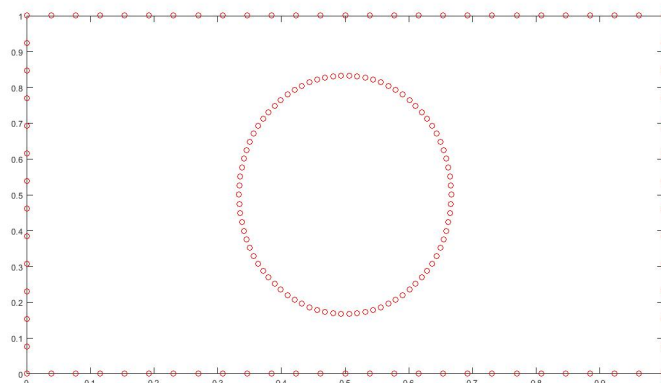


Figura 6.10: Geração de pontos do contorno da superfície 6

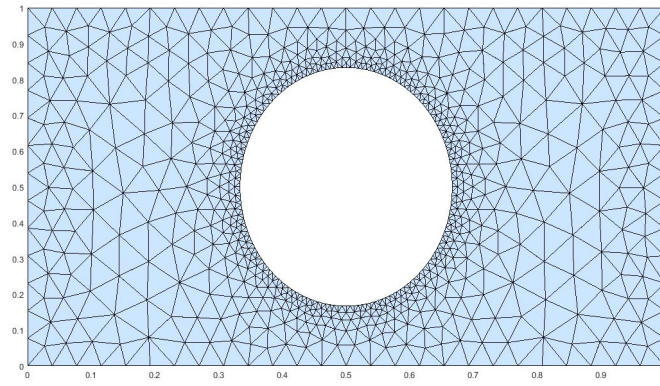


Figura 6.11: Geração de pontos internos da superfície 6

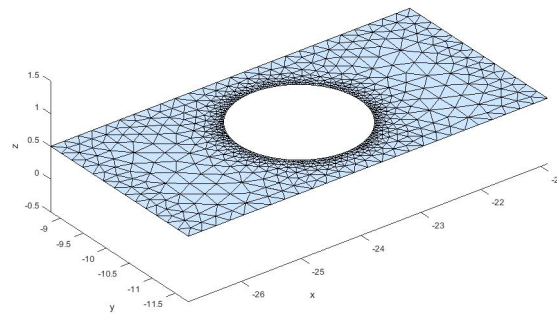


Figura 6.12: Transferência da malha do plano 2D para o espaço 3D da superfície 6.

Após passar pelo processo de geração de malha e de aplicação das condições de contorno, é gerado o mapa de temperatura da Figura 6.13. A Tabela 6.5 mostra a quantidade de elementos em ambos os sólidos, tanto o mancal quanto a placa com furo. Como são utilizados elementos constantes, há um nó por elemento e o número de nós e elementos é o mesmo.



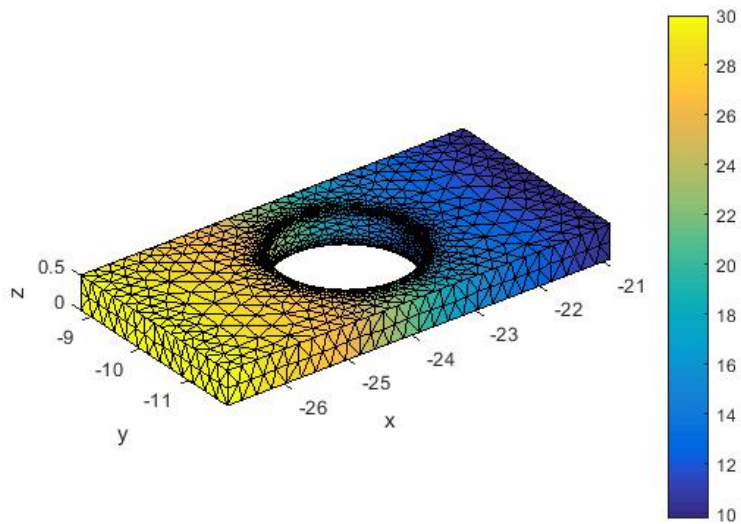


Figura 6.13: Temperaturas na placa com furo.

Tabela 6.5: Quantidades de elementos dos exemplos.

	Placa	Mancal
Número de elementos	4632	8422

# Capítulo 7

## Conclusão e proposta para trabalhos futuros

### 7.1 Conclusão

Este trabalho apresentou uma abordagem na qual o método dos elementos foi utilizado na análise de problemas de condução de calor em sólidos. O foco principal do trabalho foi na leitura de arquivos de troca em formato IGES (arquivos com a extensão .igs ou .iges) e na geração da malha de superfície. Foram utilizados pacotes com códigos abertos tanto para a leitura dos arquivos de troca quanto na geração da malha. Todos estes pacotes foram integrados em um único programa em conjunto com códigos referentes ao método dos elementos de contorno. A maioria das funções foram implementadas em linguagem Matlab/Octave. Entretanto, com o objetivo de se melhorar o desempenho do código, diminuindo o custo de execução, as rotinas que apresentavam um maior custo computacional foram implementados em linguagem C e compilados na forma de arquivos mex e também octfile.

Embora o pacote IGES toolbox não tenha todas as entidades presentes no formato IGES, ele tem a maioria delas. Além disso, como se trata de um código aberto, qualquer pessoa poderá implementar as entidades que porventura precisar e não se encontrar implementada no pacote. Foram analisados dois modelos sólidos, sendo um gerado no programa CAD comercial Solidworks e outro no programa CAD de código aberto Freecad. Ambos os sólidos foram lidos com sucesso pelo pacote IGES toolbox. Cada uma das superfícies dos sólidos analisados é representada por uma superfície aparada no pacote IGES toolbox. Estas superfícies aparadas são representadas como um retângulo no plano paramétrico junto com as curvas de apartamento que também estão neste plano. A discretização pelo pacote MESH2D é feita neste plano paramétrico em cada uma das superfícies. Embora não tenham sido implementadas funções que corrijam a malha considerando a curvatura e a inclinação das superfícies no espaço, as implementações obtiveram bons resultados para os problemas analisados. Foi necessário a implementação de rotinas para inverter o sentido dos vetores normais pois em algumas superfícies, os elementos triangulares foram gerados com o sentido do vetor normal

invertido. Uma vez gerada a malha, as matrizes de influência do método dos elementos de contorno são montadas, as condições de contorno são aplicadas, as variáveis desconhecidas são calculadas, temperatura e fluxo são separados em dois vetores e, por fim, um mapa de temperatura é gerado na superfície dos sólidos.

O código implementado se mostrou robusto, rápido e fácil de usar. Embora a formulação tenha sido desenvolvida para problemas de condução de calor, ela é facilmente extensível para outras formulações do método dos elementos de contorno como, por exemplo, problemas elásticos, acústicos, eletromagnéticos, dentre outros. Caso se opte por usar o Freecad ao invés do Solidworks e o Octave, ao invés do Matlab, tem-se toda a formulação em código aberto, sem qualquer custo para o usuário.

## 7.2 Proposta para trabalhos futuros

Apesar dos bons resultados na geração da malha superficial e na análise de problemas tridimensionais, este projeto encontra-se em um estágio inicial que ainda demanda muitas melhorias para tornar o código amigável ao usuário. Como trabalhos futuros, é sugerido os seguintes tópicos:

1. Implementar a verificação da qualidade e otimização da malha superficial. Atualmente apenas a qualidade da malha no plano paramétrico é verificada e, caso não esteja adequada, a malha é modificada. Depois do mapeamento na superfície, devido a curvatura ou inclinação da superfície, estes triângulos podem se deformar excessivamente. Entretanto, em superfícies com curvaturas constantes como as analisadas neste trabalho, a qualidade da malha é mantida depois do mapeamento no espaço.

2. A reordenação dos nós dos elementos das superfícies com vetores normais com sentidos invertidos é, atualmente, um processo pouco iterativo. O uso de processos gráficos mais iterativos deve ser um dos pontos presentes em trabalhos futuros.

3. A aplicação das condições de contorno também deve ser melhorada e um processo mais iterativo deve ser implementado.

4. A formulação deve ser estendida para outras análises como problemas elásticos e acústicos, dentre outros.

# Referências Bibliográficas

- AGUIRRE, J. 2017. <<http://aguirreeng.blogspot.com.br/2010/09/malhas-estruturadas-e-nao-estruturadas.html>>. [Online; acessado em 12-Janeiro-2017].
- ALBUQUERQUE, E. L. *Introdução ao Método dos Elementos de Contorno*. [S.l.], 2012.
- AZEVEDO, A. F. M. *Método dos Elementos Finitos*. [S.l.], 2003.
- AZEVEDO, C. A. C. *Formulação alternativa para análise de domínios não homogêneos e inclusões anisotrópicas via MEC*. Dissertação (Mestrado) — Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2007.
- BALAN, M. *Tyre and Rim Model*. 2016. <<https://grabcad.com/library/tyre-rim-model-1>>. [Online; acessado em 03-Novembro-2016].
- BATISTA, V. H. F. *Geração de Malhas Não-Estruturadas Tetraédricas Utilizando um Método de Avanço de Fronteira*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, 2005.
- BEER, G. *Advanced Numerical Simulation Methods - From CAD Data Directly to Simulation Results*. [S.l.]: CRC Press, 2015.
- BERGSTRÖM, P. *IGES Toolbox*. 2017. <[www.mathworks.com/matlabcentral/fileexchange/13253-iges-toolbox](http://www.mathworks.com/matlabcentral/fileexchange/13253-iges-toolbox)>. [Online; acessado em 29-Janeiro-2017].
- BOROUCHAKI, H.; GEORGE, P.; LO, S. Optimal delaunay point insertion. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 39, n. 20, p. 3407–3437, 1996.
- BOROUCHAKI, H. et al. Surface mesh enhancement with geometric singularities identification. *Computer methods in applied mechanics and engineering*, Elsevier, v. 194, n. 48, p. 4885–4894, 2005.
- BOWYER, A. Computing dirichlet tessellations. *The computer journal*, Br Computer Soc, v. 24, n. 2, p. 162–166, 1981.
- CAMPOS, L. S. *Método dos Elementos de Contorno Isogeométricos Rápido*. Tese (Doutorado) — Universidade de Brasília, Brasília, 2016.
- CHENG, A. H. D.; CHENG, D. T. Heritage and early history of the boundary element method. *Engineering Analysis with Boundary Element*, v. 29, n. 3, p. 268–302, 2005.
- CHENG, S.; DEY, T. K.; SHEWCHUK, J. R. *Delaunay Mesh Generation*. [S.l.]: CRC Press, 2013.

- COLLETTE, B.; FALCK, D. *FreeCad [How to]: Solid Modeling with the power of Python*. [S.l.]: Packt Publishing, 2012. 69 p.
- DEUS, L. R. Mdf: Conceitos básicos e algumas aplicações na engenharia estrutural. In: *Nono Simpósio de Mecânica Computacional*. Universidade Federal de São João Del-Rei: [s.n.], 2010. p. 24–16.
- DIRICHLET, G. L. Über die reduction der positiven quadratischen formen mit drei unbestimmten ganzen zahlen. *Journal für die reine und angewandte Mathematik*, v. 40, p. 209–227, 1850.
- ENGWIRDA, D. *MESH2D - Delaunay-based unstructured mesh generation*. 2017. <<http://www.mathworks.com/matlabcentral/fileexchange/25555-mesh2d-delaunay-based-unstructured-mesh-generation>>. [Online; acessado em 29-Janeiro-2017].
- FARIN, G. E. *NURB curves and surfaces: from projective geometry to practical use*. [S.l.]: AK Peters, Ltd., 1995.
- FARIN, G. E.; HOSCHEK, J.; KIM, M.-S. *Handbook of computer aided geometric design*. [S.l.]: Elsevier, 2002.
- FILHO, G. C. *Geração de Malha Tridimensional para o Método dos Elementos de Contorno*. Campinas: [s.n.], 2004.
- FREY, P. J.; GEORGE, P. L. *Mesh Generation: Application to Finite Elements*. [S.l.]: HERMES Science Publishing, 2000.
- GAUL, L.; KÖGL, M.; WAGNER, M. *Boundary Element Methods for Engineers and Scientists - An Introductory Course with Advanced Topics*. [S.l.]: Springer, 2003.
- GONÇALVES, J. E. S. B.S. Thesis, *Leitura e interpretação de arquivos IGES para uso em programas de elementos de contorno*. Brasília: [s.n.], 2016.
- GONÇALVES, N. D. F. et al. *Método dos volumes finitos em malhas não-estruturadas*. *Portugal: Faculdade de Ciências da Universidade do Porto*, 2007.
- HAVRE, Y. van. *FreeCad*. 2016. <<http://freecadweb.org/>>.
- ITO, Y.; NAKAHASHI, K. Direct surface triangulation using stereolithography data. *AIAA journal*, v. 40, n. 3, p. 490–496, 2002.
- JÚNIOR, A. B. D. *O Método dos Elementos de Contorno Rápido com Expansão em Multipólos Aplicado a Problemas de Elasticidade Plana*. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 2014.
- KANE, J. H. *Boundary Element Analysis in Engineering Continuum Mechanics*. [S.l.]: Prentice Hall, 1994.
- LAN, T.; LO, S. Finite element mesh generation over analytical curved surfaces. *Computers & Structures*, Elsevier, v. 59, n. 2, p. 301–309, 1996.
- LIU, Y. *Fast multipole boundary element method - Theory and Applications in Engineering*. [S.l.]: Cambridge, 2009.

- LO, D. S. H. *Finite Element Mesh Generation*. [S.l.]: CRC Press, 2015.
- LO, S.; LAU, T. Mesh generation over curved surfaces with explicit control on discretization error. *Engineering Computations*, MCB UP Ltd, v. 15, n. 3, p. 357–373, 1998.
- LOHNER, R. Regridding surface triangulations. *Journal of Computational Physics*, v. 126, p. 1–10, 1996.
- MACKERLE, J. 2d and 3d finite element meshing and remeshing: A bibliography (1990-2001). *Engineering computations*, MCB UP Ltd, v. 18, n. 8, p. 1108–1197, 2001.
- MALISKA, C. *Transferência de Calor e Mecânica dos Fluidos Computacional*. [S.l.]: Wiley, 2004.
- OLIVEIRA, T. A. S. *Geração Automática de Malha de Elementos Finitos Tridimensionais para Sólidos Convexos*. Belo Horizonte: [s.n.], 1995.
- PARK, J.; SHONTZ, S. M. An alternative mesh quality metric scheme for efficient mesh quality improvement. In: *International Conference on Computational Science, ICSS*. [S.l.: s.n.], 2011.
- PARK, S.; WASHAM, C. Drag method as a finite element mesh generation scheme. *Computers & Structures*, Elsevier, v. 10, n. 1-2, p. 343–346, 1979.
- PIEGL, L.; TILLER, W. *The NURBS book*. [S.l.]: Springer Science & Business Media, 2012.
- REBOREDO, R. R. *Refinamento Adaptativo Tipo H aplicado à Formulação de Elasticidade Plana do Método dos Elementos de Contorno Dual*. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 2015.
- REMACLE, J. et al. High-quality surface remeshing using harmonic maps. *International Journal for Numerical Methods in Engineering*, Wiley-Blackwell, 111 River Street Hoboken NJ 07030-5774 USA, v. 83, n. 4, p. 403–425, 2010.
- ROGERS, D. F. *An introduction to NURBS: with historical perspective*. [S.l.]: Elsevier, 2000.
- SANCHES, I. J.; FURLAN, D. C. *Métodos Numéricos*. [S.l.], 2007.
- SHOSTKO, A. A.; LÖHNER, R.; SANDBERG, W. C. Surface triangulation over intersecting geometries. *International journal for numerical methods in engineering*, Wiley Online Library, v. 44, n. 9, p. 1359–1376, 1999.
- SMITH, B.; WELLINGTON, J. *Initial graphics exchange specification (IGES); version 3.0*. [S.l.], 1986.
- SMITH, B. M. Iges: A key to cad/cam systems integration. *IEEE COMP. GRAPHICS APPLIC.*, v. 3, n. 8, p. 78–83, 1983.
- SYSTEMES, D. 2017. <[http://help.solidworks.com/2012/Portuguese-brazilian/SolidWorks/sldworks/HIDD\\_FEAT\\_TRIM\\_REF\\_SURFACE.htm?id=860705719ad14a0d800e0f1fa4b2147a](http://help.solidworks.com/2012/Portuguese-brazilian/SolidWorks/sldworks/HIDD_FEAT_TRIM_REF_SURFACE.htm?id=860705719ad14a0d800e0f1fa4b2147a)>. [Online; acessado em 10-Fevereiro-2017].
- SYSTEMES, D. *Solidworks*. 2017. <<https://www.solidworks.com/>>. [Online; acessado em 29-Janeiro-2017].

THACKER, W. A brief review of techniques for generating irregular computational grids. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 15, n. 9, p. 1335–1341, 1980.

USPRO. *Initial Graphics Exchange Specification IGES 5.3*. [S.l.: s.n.], 1996. <[https://web.archive.org/web/20120821190122/http://www.uspro.org/documents/IGES5-3\\_forDownload.pdf](https://web.archive.org/web/20120821190122/http://www.uspro.org/documents/IGES5-3_forDownload.pdf)>. [Online; acessado em 08-Fevereiro-2017].

WATSON, D. F. Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *The computer journal*, Br Computer Soc, v. 24, n. 2, p. 167–172, 1981.

ZIENKIEWICZ, O.; PHILLIPS, D. An automatic mesh generation scheme for plane and curved surfaces by ‘isoparametric’ co-ordinates. *International Journal for Numerical Methods in Engineering*, Wiley Online Library, v. 3, n. 4, p. 519–528, 1971.

## Apêndice A

# ROTINA PARA TRANSFERÊNCIA DA MALHA GERADA NO PLANO 2D PARA O ESPAÇO 3D

Rotina desenvolvida em Matlab para a transferência da malha gerada no plano paramétrico para o espaço.

```
function [MALHA,nossup]=Discretiza3D(ParameterData,scl,minNumP,indtrimsup)
contaelem=0;
contasup=0;
nossup=zeros(length(indtrimsup),1);
for ind=indtrimsup
contasup=contasup+1;
close all
srfind=ParameterDataind.pts;
n2=ParameterDataind.n2;
NN=ones(1,n2+1);
numEntScale=min(200/ParameterDataind.nument,1.5);
comprimentocitar curva=ParameterDataParameterDataind.pto.length;
comprimentodiagonal=ParameterDataParameterDataind.pto.gdiagonal;
NO=max(ceil(numEntScale*scl*(comprimentocurva/comprimentodiagonal)),minNumP);
NN(1)=NO;
for j=1:n2
NN(j+1)=max(ceil(numEntScale*(ParameterDataParameterDataind.pti(j).length/ParameterDataParameterDataind.pto.gdiagonal)),minNumP);
end
NN=cumsum(NN);
numP=NN(end);
UV=zeros(2,numP);
UV(:,1:NO)=ret2Crv2(ParameterData,ParameterDataind.pto,NO);
```



```

for j=1:n2
UV(:,(NN(j)+1):NN(j+1))=ret2Crv2(ParameterData,ParameterDataind.pti(j),(NN(j+1)-
NN(j)));
end

    uscale=ParameterDatsrfind.ratio(1)/(ParameterDatsrfind.u(2)-
ParameterDatsrfind.u(1));
    vscale=ParameterDatsrfind.ratio(2)/(ParameterDatsrfind.v(2)-
ParameterDatsrfind.v(1));
    UV(1,:)=UV(1,)*uscale;
    UV(2,:)=UV(2,)*vscale;
    numC=size(NN,2);
    numP=NN(end);

    Constraints=zeros(numP,2);
    Constraints(:,1)=(1:numP)';
    Constraints(:,2)=(2:(numP+1))';

    Constraints(NN(1),2)=1;
    for i=2:numC
    Constraints(NN(i),2)=NN(i-1)+1;
    end

    [UV,TRI] = mesh2d(UV',Constraints);
    figure
    patch('faces',TRI,'vertices',UV,'facecol',[.8,.9,1],'edgecol','k')

    UV(:,1)=UV(:,1)/uscale;
    UV(:,2)=UV(:,2)/vscale;
    figure
    patch('faces',TRI,'vertices',UV,'facecol',[.8,.9,1],'edgecol','k')

    P=nrbevalIGES(ParameterDatsrfind.nurbs,UV');
    figure
    patch('faces',TRI,'vertices',P,'facecol',[.8,.9,1],'edgecol','k')
    view(3)
    axis image
    xlabel('x')
    ylabel('y')
    zlabel('z')
    ntri=size(TRI,1);

```

```
for i=1:ntri
no1=TRI(i,1);
no2=TRI(i,2);
no3=TRI(i,3);
x1=P(1,no1);
y1=P(2,no1);
z1=P(3,no1);
x2=P(1,no2);
y2=P(2,no2);
z2=P(3,no2);
x3=P(1,no3);
y3=P(2,no3);
z3=P(3,no3);
MALHA(i+contaelem,1:13)=[x1,y1,z1,x2,y2,z2,x3,y3,z3,2,0,contasup,i+contaelem];
end
nossup(contasup)=contaelem+1;
contaelem=contaelem+ntri;
end
return
```