

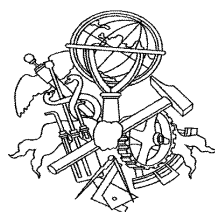
Desenvolvimento de programa em linguagem *Matlab*[®] para cálculo pelo Método de Elementos Finitos

Sandro Neves Campos

Dissertação submetida para a obtenção do grau de Mestre em
Engenharia Mecânica

Instituto Superior de Engenharia do Porto

Departamento de Engenharia Mecânica



16 de Novembro de 2014

Relatório da Unidade Curricular de Dissertação do 2º ano do Mestrado em Engenharia
Mecânica

Candidato: Sandro Neves Campos, 1091000@isep.ipp.pt

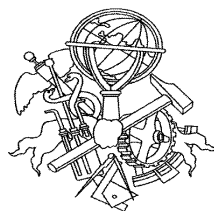
Orientação Científica: Raul Duarte Salgueiral Gomes Campilho, rds@isep.ipp.pt

Coorientação Científica: Jorge Manuel Costa Da Fonseca Justo, jfj@isep.ipp.pt

Mestrado em Engenharia Mecânica

Departamento de Engenharia Mecânica

Instituto Superior de Engenharia do Porto



16 de Novembro de 2014

Dedico este trabalho aos meus pais, pela paciência e atenção dedicada durante anos...

Agradecimentos

Quero agradecer a todos os que ao longo dos anos me ajudaram a dar mais um passo na minha jornada académica, que culmina com este projeto.

Primeiro, ao meu professor e orientador de projeto, o Professor Raul Duarte Campilho, pela formação sobre elementos finitos e pela ajuda durante o desenvolvimento do *software* que aqui é apresentado. Pelo tempo dispensado à procura em conjunto de soluções e abordagens. Pelas palavras alentadoras e pela celebração das pequenas vitórias. Ao meu também professor e coordenador, o Professor Jorge Fonseca Justo. Pelas suas observações e orientações para a melhoria da qualidade do programa.

Ao meu antigo Professor durante os meus anos na Faculdade de Engenharia da Universidade do Porto (FEUP), o Professor António Mendes Ferreira. Nunca imaginaria que depois de tantos anos o iria reencontrar no meu caminho. Devo a ele as minhas bases de Álgebra e Geometria Analítica, também necessárias durante a realização deste projeto. Também devo agradecer ao Professor António Mendes Ferreira, pelo seu trabalho publicado “*MATLAB® Codes for Finite Element Analysis*”, que se revelou a pedra angular onde assenta este trabalho.

À minha família. Que me proporcionou espaço e tempo necessário para concluir este projeto. Que compreenderam o meu mau humor, e me alentaram a dar um passo de cada vez. À minha mãe, que apesar de não ter conhecimentos de programação, ouviu-me divagar e por vezes me ajudou a descobrir falhas nos meus raciocínios. Ao meu pai, pela preocupação e pela paciência. Pela vontade de querer ajudar-me...

Aos meus colegas, amigos e alunos. Por compreenderem o porquê de ter estado ausente.

À minha namorada, o meu modelo a seguir durante a realização deste trabalho. Pois quis o destino que, à semelhança dela, também eu realizasse a minha tese desenvolvendo um programa em linguagem *Matlab®*.

E por último, devo agradecer a todos os programadores, que ao longo dos anos foram ajudando a responder a dúvidas e problemas nos fóruns *online*. Sem o contributo deles, este trabalho nunca estaria concluído com a qualidade aqui mostrada.

Resumo

Para o projeto de qualquer estrutura existente (edifícios, pontes, veículos, máquinas, etc.) é necessário conhecer as condições de carga, geometria e comportamento de todas as suas partes, assim como respeitar as normativas em vigor nos países nos quais a estrutura será aplicada. A primeira parte de qualquer projeto nesta área passa pela fase da análise estrutural, onde são calculadas todas as interações e efeitos de cargas sobre as estruturas físicas e os seus componentes de maneira a verificar a aptidão da estrutura para o seu uso.

Inicialmente parte-se de uma estrutura de geometria simplificada, pondo de parte os elementos físicos irrelevantes (elementos de fixação, revestimentos, etc.) de maneira a simplificar o cálculo de estruturas complexas e, em função dos resultados obtidos da análise estrutural, melhorar a estrutura se necessário.

A análise por elementos finitos é a ferramenta principal durante esta primeira fase do projeto. E atualmente, devido às exigências do mercado, é imprescindível o suporte computadorizado de maneira a agilizar esta fase do projeto. Existe para esta finalidade uma vasta gama de programas que permitem realizar tarefas que passam pelo desenho de estruturas, análise estática de cargas, análise dinâmica e vibrações, visualização do comportamento físico (deformações) em tempo real, que permitem a otimização da estrutura em análise. Porém, estes programas demonstram uma certa complexidade durante a introdução dos parâmetros, levando muitas vezes a resultados errados. Assim sendo, é essencial para o projetista ter uma ferramenta fiável e simples de usar que possa ser usada para fins de projeto de estruturas e otimização.

Sobre esta base nasce este projeto tese onde se elaborou um programa com interface gráfica no ambiente Matlab[®] para a análise de estruturas por elementos finitos, com elementos do tipo Barra e Viga, quer em 2D ou 3D. Este programa permite definir a estrutura por meio de coordenadas, introdução de forma rápida e clara, propriedades mecânicas dos elementos, condições fronteira e cargas a aplicar. Como resultados devolve ao utilizador as reações, deformações e distribuição de tensões nos elementos quer em forma tabular quer em representação gráfica sobre a estrutura em análise. Existe ainda a possibilidade de importação de dados e exportação dos resultados em ficheiros XLS e XLSX, de maneira a facilitar a gestão de informação.

Foram realizados diferentes testes e análises de estruturas de forma a validar os resultados do programa e a sua integridade. Os resultados foram todos satisfatórios e convergem para os resultados de outros programas, publicados em livros, e para cálculo a mão feitos pelo autor.

Palavras-Chave

Análise estrutural, Método de Elementos Finitos, Matlab[®], programa, interface gráfica.

Abstract

For the design of any existing structure (buildings, bridges, vehicles, machines, etc.) it is necessary to know the load conditions, geometry and behavior of all its parts, as well as to respect the standards in effect in the countries in which the structure will be implemented. The first part of any project in this area involves the structural analysis phase, in which all interactions and loads effects are calculated over the physical structures and their components in order to verify the suitability of the structure for its use.

Initially, a simplified geometry structure, setting aside the irrelevant physical components (fasteners, coatings, etc.) in order to simplify the calculation of complex structures and, depending on the results of the structural analysis, it is necessary to redefine the structure.

The finite element method is the main tool during this first phase of the project. Currently due to market requirements, computer support is mandatory to expedite this phase of the project. With this purpose, a wide range of programs exist that allow to perform tasks such as the design of structures, static loads analysis, dynamic vibration analysis, physical behavior (deformation) visualization in real time, enabling the optimization of the analysis results. However, these programs demonstrate a certain complexity during the introduction of parameters, often leading to erroneous results. Therefore, it is essential for designers to have a reliable and easy-to-use tool that can be used for structural design and optimization.

On this basis is born this thesis project where a complete program with graphical user interface on Matlab® environment was developed for the finite element analysis with bar and beam type elements in either 2D or 3D. This program enables defining the structure through fast and clear introduction of node coordinates, mechanical properties of the elements, boundary conditions and applied loads. The program returns as results, the reactions, strains and stresses distributions in the elements, either in tabular form or graphical representation over the structure under analysis. Still it allows the importation of data and exportation of results in XLS and XLSX files in order to ease the management of information.

Various tests and analyses of structures were performed to validate the results of the program and its integrity. The results were all satisfactory and converge to the results by other programs, published in books, and to manual calculations made by the author.

Keywords

Structural analysis, Finite Element Method, Matlab[®], program, graphical interface.

Índice

AGRADECIMENTOS.....	VII
RESUMO	IX
ABSTRACT	XI
ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABELAS	XIX
NOMENCLATURA.....	XXI
1 INTRODUÇÃO	1
1.1 CONTEXTUALIZAÇÃO	1
1.2 OBJETIVOS.....	1
1.3 CALENDARIZAÇÃO	2
1.4 ORGANIZAÇÃO DA DISSERTAÇÃO	3
2 REVISÃO BIBLIOGRÁFICA	5
2.1 O MÉTODO DE ELEMENTOS FINITOS.....	5
2.1.1 A evolução do Método de Elementos Finitos.....	5
2.1.2 Princípios gerais do Método de Elementos Finitos.....	9
2.1.3 Potencialidades	12
2.1.4 Aplicações do Método de Elementos Finitos.....	13
2.1.5 Tipos de Elementos Finitos para análise estrutural	17
2.1.6 Técnica do Método de Elementos Finitos.....	19
2.1.7 Softwares disponíveis para análise estrutural.....	22
2.1.7.1 Softwares comerciais de uso genérico	23
2.1.7.2 Códigos específicos.....	25
2.2 SOFTWARES DE COMPUTAÇÃO E VISUALIZAÇÃO GRÁFICA	28
2.2.1 Características gerais dos softwares existentes no mercado.....	29
2.2.2 Seleção do software a utilizar no desenvolvimento deste trabalho	31
2.2.3 Funcionalidades pretendidas para o software a desenvolver.....	31
3 FORMULAÇÃO DOS ELEMENTOS FINITOS A IMPLEMENTAR.....	33
3.1 ELEMENTOS DE BARRA 2D.....	33
3.1.1 Formulação linear	33
3.2 ELEMENTOS DE BARRA 3D.....	40
3.2.1 Formulação linear	40
3.3 ELEMENTOS DE ESTRUTURA 2D.....	44

3.3.1	<i>Formulação linear</i>	44
3.4	ELEMENTOS DE ESTRUTURA 3D	49
3.4.1	<i>Formulação linear</i>	49
4	DESENVOLVIMENTO DO SOFTWARE	57
4.1	ESTRUTURA	57
4.2	MÉTODOS DE PROGRAMAÇÃO	64
4.3	BASES DE DADOS	64
4.4	FUNCIONAMENTO DO SOFTWARE	66
4.4.1	<i>Arranque do programa</i>	66
4.4.2	<i>Seleção do tipo de estrutura a analisar</i>	67
4.4.3	<i>Propriedades dos elementos e geometria</i>	71
4.4.4	<i>Introdução de condições de análise, fronteira e carregamentos</i>	73
4.4.5	<i>Apresentação de resultados</i>	76
4.4.6	<i>Edição de propriedades dos elementos e condições de análise</i>	81
4.5	VALIDAÇÃO DO SOFTWARE DESENVOLVIDO	83
4.5.1	<i>Elementos de barra 2D</i>	83
4.5.2	<i>Elementos de barra 3D</i>	87
4.5.3	<i>Elementos de viga 2D</i>	91
4.5.4	<i>Elementos de viga 3D</i>	95
5	CONCLUSÕES	99
	REFERÊNCIAS	101
	BIBLIOGRAFIA CONSULTADA	107
	ANEXO A. GUIA E DICAS DE CÓDIGOS MATLAB®	109

Índice de Figuras

Figura 1 Exemplo da aplicação de Elementos finitos [7].....	9
Figura 2 Elementos Finitos para o estudo de avião A319 [8]	10
Figura 3 Elementos unidimensionais e nós	11
Figura 4 Elemento Triangular de 3 nós.....	11
Figura 5 Exemplo de pilar de estrutura [10].....	14
Figura 6 Chassi de autocarro [11]	14
Figura 7 Campo de tensões em um indutor axial usando campo de pressão de uma análise de CFD [12]	15
Figura 8 Análise de campo magnético usando MagView [13]	15
Figura 9 Transístor MOSFET [14].....	16
Figura 10 Simulação 3D do crescimento de um tumor cerebral [17].....	17
Figura 11 Elementos comumente utilizados na formulação MEF [6].....	18
Figura 12 Sistema local vs. sistema global do elemento de barra 2D	34
Figura 13 Projeção de deslocamentos	37
Figura 14 Sistema local vs. sistema global barra 3D	40
Figura 15 Elemento de barra 3D numa posição genérica no espaço, ilustrando o ângulo do elemento com os eixos globais	41
Figura 16 Sistema local vs. sistema global do elemento de viga 2D	45
Figura 17 Projeção dos deslocamentos para as coordenadas globais para um elemento de viga 2D	45
Figura 18 Carregamentos equivalentes [6].....	47
Figura 19 Exemplo de uma estrutura constituída por elementos de Viga 3D	49

Figura 20 Eixos locais de um elemento de Viga 3D	50
Figura 21 Convenções de Sinais para elemento de Viga 3D.....	50
Figura 22 Momento torsor aplicado ao elemento.....	51
Figura 23 Disposição dos deslocamentos relativamente às coordenadas globais	53
Figura 24 Plano de referência para determinar cossenos diretores	54
Figura 25 Diagrama base do programa	57
Figura 26 Diagrama programático expandido - parte 1 de 2.....	59
Figura 27 Diagrama programático expandido - parte 2 de 2.....	61
Figura 28 Mudança de diretoria de trabalho.....	66
Figura 29 Início do programa	66
Figura 30 Seleção do tipo de análise	67
Figura 31 Método de entrada de dados.....	68
Figura 32 Método de importação	68
Figura 33 Aviso para importação de CAD	69
Figura 34 Abertura de ficheiro tipo folha de cálculo	70
Figura 35 Abertura de ficheiro de sessão anterior.....	70
Figura 36 Propriedades do 1º elemento.....	71
Figura 37 Propriedades dos elementos.....	72
Figura 38 Nova coordenada	72
Figura 39 Novo ponto criado.....	73
Figura 40 Janela ações nos nós.....	74
Figura 41 Ações e restrições sobre a estrutura	75
Figura 42 Representação de apoio encastrado em análise de elementos viga em 3D	76

Figura 43 Janela de resultados	76
Figura 44 Resultados: Reações	78
Figura 45 Resultados: Deslocamentos	78
Figura 46 Tensões para elementos de barra	79
Figura 47 Aviso das soluções das ações nos elementos.....	79
Figura 48 Comparação de seletores de resultados para as ações nos elementos. À esquerda para elementos de vigas 2D, à direita para elementos de viga 3D.	80
Figura 49 Resultados: Tensão axial	79
Figura 50 Edição de dados na folha de cálculo.....	81
Figura 51 Janela edição, modo “propriedades”	81
Figura 52 Janela edição, modo “ações”.....	82
Figura 53 Problema para validação do estudo de elementos de barra 2D [2]	83
Figura 54 Estrutura para validação do estudo de elementos de barra 2D	84
Figura 55 Problema para validação do estudo de elementos barra 3D [6].....	87
Figura 56 Estrutura para validação do estudo de elementos de barra 3D	88
Figura 57 Problema para validação do estudo de elementos de viga 2D [2]	91
Figura 58 Estrutura para validação do estudo de elementos de viga 2D.....	91
Figura 59 Problema para validação do estudo de elementos viga 3D [6]	95
Figura 60 Estrutura para validação do estudo de elementos de viga 3D.....	95

Índice de Tabelas

Tabela 1 Calendário de Atividades	3
Tabela 2 Variáveis utilizadas na programação - parte 1 de 3.....	62
Tabela 3 Variáveis utilizadas na programação - parte 2 de 3.....	63
Tabela 4 Variáveis utilizadas na programação - parte 3 de 3.....	64
Tabela 5 Exemplo de gestão de variáveis e montagem da matriz de rigidez global.....	65
Tabela 6 Representação de reações em 2D	75
Tabela 7 Valor real vs. valor residual	76
Tabela 8 Tipos de estruturas disponíveis para visualização.....	77
Tabela 9 Coordenadas utilizadas na validação do estudo de elementos de barra 2D.....	84
Tabela 10 Ligações utilizadas na validação do estudo de elementos de barra 2D	84
Tabela 11 Propriedades dos elementos do problema de validação para estudo de elementos barra 2D.....	85
Tabela 12 Resultados: Deslocamentos para validação do estudo para elementos de barra 2D.....	85
Tabela 13 Resultados: Reações para validação do estudo para elementos de barra 2D.....	86
Tabela 14 Resultados: Tensões para validação do estudo para elementos de barra 2D.....	86
Tabela 15 Coordenadas utilizadas na validação do estudo de elementos de barra 3D.....	88
Tabela 16 Ligações utilizadas na validação do estudo de elementos de barra 3D	88
Tabela 17 Propriedades dos elementos do problema de validação para estudo de elementos barra 3D.....	88
Tabela 18 Resultados: Deslocamentos para validação do estudo para elementos de barra 3D.....	89
Tabela 19 Resultados: Reações para validação do estudo para elementos de barra 3D.....	90

Tabela 20 Resultados: Tensões para validação do estudo para elementos de barra 3D	90
Tabela 21 Coordenadas utilizadas na validação do estudo de elementos de viga 2D	92
Tabela 22 Ligações utilizadas na validação do estudo de elementos de viga 2D.....	92
Tabela 23 Propriedades dos elementos do problema de validação para estudo de elementos viga 2D	92
Tabela 24 Resultados: Deslocamentos para validação do estudo para elementos de viga 2D	93
Tabela 25 Resultados: Reações para validação do estudo para elementos de viga 2D	94
Tabela 26 Coordenadas utilizadas na validação do estudo de elementos de viga 3D	96
Tabela 27 Ligações utilizadas na validação do estudo de elementos de viga 3D.....	96
Tabela 28 Pontos de suporte para transformada de coordenadas	96
Tabela 29 Propriedades dos elementos do problema de validação para estudo de elementos viga 3D	96
Tabela 30 Resultados: Deslocamentos para validação do estudo para elementos de viga 3D	97
Tabela 31 Resultados: Reações para validação do estudo para elementos de viga 3D	98

Nomenclatura

Caracteres Romanos

X e xx – referem à direção X nas coordenadas locais.

Y e yy – referem à direção Y nas coordenadas locais.

Z e zz – referem à direção Z nas coordenadas locais.

[m] – Metro. Unidade de medida distância.

[N] – Newton. Unidade de medida de força.

[N.m] – Newton $\times$ Metro. Unidade de medida do momento.

[Pa] – Pascal. Unidade de medida de pressão; força sobre a área; tensão.

[rad] – Radiano. Unidade de medida angular; razão do comprimento do arco e o seu raio.

Caracteres Gregos

ϕ – Phi. Refere à variável de campo dos elementos.

λ – Lambda. Refere à matriz dos cossenos diretores para os elementos de viga 3D.

σ – Sigma. Refere às tensões genéricas.

ε – Epsilon. Refe às deformações.

τ – Tau. Refere às tensões de corte.

θ – Theta. Refere à rotação sobre um eixo.

Operadores

i e j – referem aos índices de contadores genéricos.

N_i – referente às funções de forma dos elementos.

u_i - referente ao deslocamento genérico local, comumente na direção X.

v_i - referente ao deslocamento genérico local, comumente na direção Y.

w_i - referente ao deslocamento genérico local, comumente na direção Z.

U_i - referente ao deslocamento genérico global, comumente na direção X.

V_i - referente ao deslocamento genérico global, comumente na direção Y.

W_i - referente ao deslocamento genérico global, comumente na direção Z.

Rx_i - referente à reação do apoio, na direção X por nó.

Ry_i - referente à reação do apoio, na direção Y por nó.

Rz_i - referente à reação do apoio, na direção Z por nó.

RMx_i - referente à reação angular do apoio, na direção X por nó.

RMY_i - referente à reação angular do apoio, na direção Y por nó.

RMz_i - referente à reação angular do apoio, na direção Z por nó.

Mx_i – referente ao momento aplicado segundo X por nó.

A_e e a_e – referente aos vetores de variáveis nodais em coordenadas globais e locais, respectivamente.

F_e e f_e – referente aos vetores de forças nodais em coordenadas globais e locais, respectivamente.

R – referente ao vetor das reações.

K e k_e – referente às matrizes de rigidez globais e locais, respectivamente.

B – referente à matriz de deformação.

D – referente à matriz de elasticidade.

ϵ_e – referente ao vetor das deformações.

T – referente à matriz de transformação para a mudança de coordenadas.

L – referente ao comprimento do elemento.

l – referente ao cosseno diretor na direção X para mudança de coordenadas.

m – referente ao cosseno diretor na direção Y para mudança de coordenadas.

n – referente ao cosseno diretor na direção Z para mudança de coordenadas.

F_{x_i} – referente à carga aplicada segundo X por nó.

F_{y_i} – referente à carga aplicada segundo Y por nó.

F_{z_i} – referente à carga aplicada segundo Z por nó.

M_{y_i} – referente ao momento aplicado segundo Y por nó.

M_{z_i} – referente ao momento aplicado segundo Z por nó.

A – referente à secção do elemento.

E – referente ao módulo de elasticidade do elemento (módulo de Young).

G – referente ao módulo de elasticidade transversal do elemento.

I – referente ao momento de inércia.

I_y – referente ao momento de inércia segundo a direção Y.

I_z – referente ao momento de inércia segundo a direção Z.

J – referente ao momento de inércia angular.

y_1 – referente à geometria da secção do elemento na cota segundo Y desde o seu eixo neutro.

y_2 – referente à geometria da secção do elemento na cota segundo Z desde o seu eixo neutro.

r – referente à geometria da secção do elemento desde o seu centro geométrico até a aresta onde é aplicado o esforço.

Abreviaturas

2D – Duas Dimensões.

3D – Três Dimensões.

GUI – *Graphical User Interface*.

CAD – *Computer Aided Design*.

MEF – Método de (dos/por/com) Elementos Finitos.

FEM – *Finite Element Method*.

CFD – Cálculo Fluido Dinâmico.

CAE – *Computer Aided Engineering*.

1 Introdução

1.1 Contextualização

Para o projeto de qualquer estrutura existente (edifícios, pontes, veículos, máquinas, etc.) é necessário conhecer as condições de carga, geometria e comportamento de todas as suas partes, assim como respeitar as normativas em vigor nos países nos quais a estrutura será aplicada. A primeira parte de qualquer projeto nesta área passa pela fase da análise estrutural, onde são calculadas todas as interações e efeitos de cargas sobre as estruturas físicas e os seus componentes de maneira a verificar a aptidão da estrutura para o seu uso.

Como com qualquer tipo de projeto, é importante ter à disposição ferramentas que permitam a validação e apoio ao cálculo. E, com o passar dos anos, nenhuma outra ferramenta se mostrou tão fiável e multidisciplinar quanto o Método de Elementos Finitos. Os modelos a ser analisados pelo Método de Elementos Finitos são cada vez mais complexos e seria impensável realizar tais análises sem um apoio computadorizado. Existe para esta finalidade uma vasta gama de programas que permitem realizar tarefas que passam pelo desenho de estruturas, análise estática de cargas, análise dinâmica e vibrações, visualização do comportamento físico (deformações) em tempo real, que permitem a otimização da estrutura. Porém, estes programas demonstram uma certa complexidade durante a introdução dos parâmetros, levando muitas vezes a resultados errados. Assim sendo, é essencial para o projetista ter uma ferramenta fiável e simples de usar que possa ser usada para fins de projeto de estruturas e otimização.

1.2 Objetivos

Sobre bases mencionadas na secção anterior, nasce este projeto tese onde ir-se-á elaborar um programa com interface gráfica no ambiente Matlab® para a análise de estruturas por elementos finitos, com elementos do tipo Barra e Viga, quer em 2D ou 3D. Este programa permitirá definir a estrutura por meio de coordenadas, introdução de forma rápida e clara, propriedades mecânicas dos elementos, condições fronteira e cargas a aplicar. Como resultados devolverá ao utilizador as

reações, deformações e distribuição de tensões nos elementos quer em forma tabular quer em representação gráfica sobre a estrutura em análise. Existirá ainda a possibilidade de importação de dados e exportação dos resultados em ficheiros XLS e XLSX, de maneira a facilitar a gestão de informação.

Sendo assim, os objetivos da presente tese passam pelo desenvolvimento de um programa em *Matlab*[®] para análise de esforços e deformações de estruturas tipo barra e viga, utilizando o Método de Elementos Finitos como base de cálculo. Para tal utilizar-se-á o modulo *GUI-Builder* (*Graphical User Interface - GUI*) para desenhar e representar todas as janelas de interface que permitam, de forma dinâmica e intuitiva, facilitar o trabalho do utilizador. Este programa irá representar a estrutura durante todas as fases da sua criação (geometria, posicionamento de esforços, etc.), assim como mostrar nos resultados a estrutura deformada e a distribuição das tensões por elemento.

Sub-objetivos:

Como sub-objetivos referem-se os seguintes:

- Permitir a importação de dados para a criação da estrutura em análise;
- Gravar os resultados de uma análise efetuada;
- Permitir a edição de um estudo efetuado previamente.

1.3 Calendarização

De maneira a atingir o primeiro dos sub-objetivos apresentados previamente, existiu um período de dois meses de pesquisa (em horário pós-laboral). Inicialmente pretendeu-se importar modelos em 2 e 3 dimensões criados por interfaces de desenho computadorizado. No entanto, esta hipótese de importação foi abandonada em virtude de dificuldades como a diversidade de codificações e tipos de ficheiros.

Assim sendo e apesar de o período indicado não ter sido produtivo para o desenvolvimento do programa, na Tabela 1 esta incluído e indicado a vermelho.

Tabela 1 Calendário de atividades

	2014																																									
Atividade	Março					Abril					Maio					Junho					Julho					Agosto					Setembro					Outubro						
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5		
Importação através de CAD																																										
Pesquisa Bibliográfica																																										
Pesquisa de Código																																										
Ambiente gráfico GUI																																										
Código apoio a ambiente gráfico																																										
Criação de código de cálculo																																										
Criação de código para PLOT																																										
Exportação de resultados																																										
Importação de Dados																																										
Validação de resultados																																										
Redação da dissertação																																										

1.4 Organização da dissertação

O autor, no desenvolvimento da presente dissertação, optou pela divisão das suas secções seguindo a cronologia do desenvolvimento do trabalho. Com isto em mente seguiu-se a seguinte divisão:

O capítulo 1 trata das características relevantes desta dissertação, tais como o contexto no qual é inserido, os objetivos, o calendário seguido e o presente ponto, a organização da dissertação.

No capítulo 2 da presente dissertação, revisão bibliográfica, ir-se-á expor as potencialidades do Método de Elementos Finitos, assim como a sua cronologia e formulação. Abordam-se também os tipos de elementos e casos de aplicação do método. A segunda parte da revisão bibliográfica, visa compreender e seleccionar o melhor programa para desenvolvimento do *software* a ser criado. Foi inevitável a comparação dos diferentes programas existentes de maneira a perceber o que o utilizador de *software* de cálculo e simulação por elementos finitos procura num programa de este tipo.

No capítulo 3 da presente dissertação, o autor expõe as características dos elementos finitos a implementar nas análises que o programa irá realizar. Também é apresentada como a formulação a seguir.

Na última parte do trabalho mostrar-se-á o programa desenvolvido, assim como um pequeno exemplo de funcionamento do programa a desenvolver. Será incluída nesta secção a validação do programa utilizando cálculos paralelos, quer por parte do autor, quer por parte de outros programas do mesmo tipo.

2 Revisão Bibliográfica

2.1 O Método de Elementos Finitos

Atualmente, o método dos elementos finitos é uma ferramenta muito utilizada para a validação de cálculos em muitos ramos da engenharia. O método nasce na necessidade de estudar o comportamento de estruturas sólidas complexas mas, com o aumento da capacidade de processamento dos computadores, rapidamente tornou-se numa ferramenta não só de cálculo, mas de validação em diferentes áreas. O método baseia o seu cálculo na divisão do modelo em estudo em áreas ou volumes infinitesimais (tão pequeno quanto se queira) e subsequentemente, calcula as interações matemáticas (forças, deslocamentos, cargas térmicas, etc.) entre o elemento e a sua vizinhança. Ou como descrito por Yijun Liu, “...pensar nos modelos como LEGO[®]s...” [1].

O Método Dos Elementos Finitos (MEF) representa uma aproximação dum modelo matemático que representa o mais fielmente possível o problema físico. Registe-se que o método não pode fornecer mais informações do que as fornecidas pelo modelo matemático [2].

Atualmente a sua aplicabilidade não se encontra limitada ao cálculo de estruturas, podem-se ver exemplos da sua aplicação na mecânica de fluidos e na transferência de calor, entre muitas outras áreas.

2.1.1 A evolução do Método de Elementos Finitos

Antes do aparecimento do MEF, a análise dos meios contínuos era efetuada por resolução direta dos sistemas de equações de derivadas parciais que regem o fenómeno, tendo em consideração as necessárias condições fronteira. Para facilitar a aplicação desta técnica a problemas não elementares, era comum recorrer a séries de Fourier. Devido à sua complexidade, estes procedimentos só eram aplicáveis a meios contínuos homogêneos e de geometria simples. Para tentar ultrapassar algumas destas limitações, era frequente a substituição de derivadas exatas por derivadas parciais, calculadas com base em grelhas de pontos. Da aplicação desta técnica resulta o Método das Diferenças Finitas, que, antes do aparecimento dos computadores, apresentava o

inconveniente de requerer a resolução de grandes sistemas de equações lineares. Para evitar este inconveniente foram propostos diversos métodos de relaxação baseados na sucessiva diminuição de um conjunto de resíduos. Devido à morosidade associada à aplicação de qualquer um destes métodos, tornava-se muito atrativa a substituição do problema real por outro semelhante, recorrendo a resultados publicados em tabelas ou ábacos [3].

Segue-se uma breve compilação dos autores que fizeram do Método de Elementos Finitos aquilo que conhecemos hoje em dia. Soriano e Lima [4] escrevem a seguinte cronologia:

- Richard Courant utilizou em 1943 [...], de forma pioneira, o método Rayleigh-Ritz em subdomínio. Adotou, então, elementos triangulares no estudo da torção de Saint-Venant.
- J.H. Argyres e S. Kelsey, com considerações de ordem física, adotaram, de forma pioneira e independente do trabalho de Courant, em 1955 [...] interpolação do campo de deslocamentos no desenvolvimento de elemento bilinear retangular de estado plano de tensões. Utilizaram o elemento no estudo de distribuição de tensões em chapas de asa de avião.
- M.J. Turner, R.W. Clough, H.C. Martin e L.J. Topp desenvolveram em 1956 [...], para estado plano de tensão, elemento triangular de deformação constante e elemento retangular mais refinado que o de Argyris e Kelsey, sem ter conhecimento dos trabalhos precursores de Courant e de Argyris. Diferentemente ao trabalho de Argyris e Kelsey, adotaram campo de deformações em função de parâmetros generalizados.
- R.W. Clough utilizou, em 1960 [...], o termo *elemento finito* na aceção aqui adotada.
- A. Adini e R.W. Clough desenvolveram, com a teoria de Kirchhoff, o primeiro elemento de placa em 1961 [...].
- R.J. Melosh apresentou, em 1963 [...], critérios de convergência para o Método de Elementos Finitos e o conceito de força nodal equivalente (com nome de *força generalizada*) a ações aplicadas ao elemento. Além disso, reapresentou o método a partir do funcional de energia potencial total, sem ter conhecimento do trabalho pioneiro de Richard Courant de 1943.
- R.H. Gallagher e J. Padlog sugeriram, em 1963 [...], a adoção de campos de deslocamentos em vigas e placas para considerar o efeito de não linearidade geométrica e obter cargas críticas. Exemplificaram a formulação, no caso de vigas, utilizando o polinómio completo do terceiro grau. Contudo, foi J.H. Argytis o autor em 1965 [...], do termo *matriz de rigidez geométrica* que se refere à rigidez adicional devida à mudança de geometria.
- J.S. Archer apresentou de forma inovadora, em 1963 [...], a *matriz de massa consistente*, utilizando lei de distribuição de aceleração no elemento (em função de valores nodais)

obtida a partir da lei adotada para o campo de deslocamentos. Exemplificou a formulação em vigas, utilizando funções de interpolação de Hermite.

- B. Irons e J. Barlow estabeleceram, em 1964 [...] o critério de convergência dos estados de tensão constantes.
- R.E. Jones e T.H.H Pian apresentaram separadamente, em 1964 [...], os primeiros elementos híbridos.
- R.E. Jones generalizou, em 1964 [...], a formulação do Método de Elementos Finitos, utilizando multiplicadores de Lagrange para relaxar condições de continuidade nas interfaces entre elementos.
- L.R. Hermann apresentou, em 1965 [...], o primeiro elemento misto, utilizando a teoria Reissner de flexão de placa semi-espessa e apresentou, em 1965 [...], o primeiro elemento para sólidos incompressíveis e quase-incompressíveis.
- B. Fraeijns de Veubeke apresentou, em 1965 [...], o primeiro elemento do modelo de equilíbrio, assim como um amplo estudo dos fundamentos *variacionais* do Método de Elementos Finitos e o princípio da limitação que rege as formulações mistas.
- F.K. Bogner, R. Fox e L.A. Schmit apresentaram, em 1965 [...], os primeiros elementos conformes de placa e de casca de teoria clássica, repressivamente com 16 e 32 graus de liberdade por elemento.
- G.P. Baseley, Y.K. Cheung, B.M. Irons e O.C. Zienkiewicz apresentaram, em 1965 [...], critérios de convergência de elementos não conformes.
- Y.K. Cheung e O.C. Zienkiewicz apresentaram, em 1965 [...], a primeira aplicação do Método de Elementos Finitos em interação solo-estrutura. Utilizaram a hipótese de Winkler em semi-espaco infinito elástico e isotrópico.
- O.C. Zienkiewicz e Y.K. Cheung estenderam, em 1965 [...], o Método de Elementos Finitos a problemas de campo bidimensional não estruturais.
- Em 1965, 1968 e 1971 ocorreram três congressos de grande importância histórica para o Método de Elementos Finitos, denominados *Conference on Matrix Methods in Estrutural Mechanics*.
- B.M. Irons generalizou em 1966 [...] o conceito de elemento isoparamétrico, cuja terminologia foi criada por I. Ergatoudis, B.M. Irons, e O.C. Zienkiewicz em 1968 [...]; a conceção da distorção do elemento retangular para a forma quadrilateral é da autoria de I.C. Taig, em 1961. Naquele mesmo trabalho, Irons sugeriu a utilização de funções de interpolação hierárquicas, de integração numérica e do *patch test*, [...] denominado teste de malha de Irons.
- E.I. Wilson e R.E. Nickell fizeram, em 1966 [...] a primeira aplicação do método em problema de transferência de calor.

- O.C. Zienkiewicz, P. Mayer e Y.K. Cheung, apresentaram em 1966 [...], a primeira aplicação do método em fluxo de meios porosos bidimensionais.
- O.C. Zienkiewicz e Y.K. Cheung foram, em 1967 [...], os autores do primeiro livro inteiramente dedicado ao Método de Elementos Finitos.
- O.C. Zienkiewicz, A.K. Bahrani e P.L. Arlett aplicaram, em 1967 [...], o método a problemas elétricos bidimensionais.
- A.G. Wempner, J.T. Oden e D.A. Kross foram, em 1968 [...], os primeiros a usar restrições discretas de Kirchhoff no desenvolvimento de elemento de casca, e Strincklin e coautores em 1969 [...] foram os primeiros a usá-las no desenvolvimento de elementos de placa. Contudo, foi B.J. Veubeke o primeiro a conceber restrições discretas, em 1965 [...], aplicando-as em elemento de viga.
- S. Ahmad, B.M. Irons e O.C. Zienkiewicz apresentaram pioneiramente em 1968 [...] o desenvolvimento de elementos curvos de casca por degeneração de elementos tridimensionais, utilizando as hipóteses de Reissner-Mindlin.
- J. Barlow, em 1968, foi o primeiro a observar que em certos pontos de integração de Gauss-Legendre podem ser obtidos melhores resultados de tensões do que nos demais pontos do elemento [...]. Por essa razão aqueles pontos receberam a denominação de pontos de Barlow.
- W.P. Daherty, E.L. Wilson e E.L. Taylor foram os primeiros a utilizar, em 1969 [...], integração reduzida, que foi modificada para a forma reduzida seletiva por S.F. Pawsey e R.W. Clough em 1971 [...].
- B.A. Szabo e G.C. Lee utilizaram pioneiramente em 1969, o método de Galerkin [...] na formulação do modelo de deslocamentos do Método de Elementos Finitos. O.C. Zienkiewicz e C.J. Parekh estenderam em 1970 [...], o uso daquele método aos campos transitórios.
- T. Belytschko, em 1976 estudou problemas associados a grandes deslocamentos em dinâmica não linear, e melhorou as técnicas numéricas para a resolução de sistemas de equações.
- J.F. Lyness, D.R.J. Owen e O.C. Zienkiewicz [...] aplicaram em 1977 o método dos resíduos ponderados na determinação de campos magnéticos.

Esta evolução foi acompanhada também com a evolução e implementação dos computadores como ferramenta de cálculo. Os primeiros cálculos em computador foram realizados através de linguagem programática do tipo FORTRAN, utilizando para este efeito computadores do tipo *Mainframe*. Nesta fase não estavam disponíveis capacidades gráficas complexas que permitissem visualização [5, 6]. O refinamento do Método de Elementos Finitos e o aumento da capacidade de processamento dos computadores levou inevitavelmente, a que este método seja, cada vez mais,

prática corrente para a análise de estruturas de geometria arbitrária, constituídas por múltiplos materiais e sujeitas a qualquer tipo de carregamento. Este avanço é tão significativo que os outros métodos iniciais deixaram praticamente de ser utilizados. Atualmente, o interesse dos métodos descritos anteriormente restringe-se ao de fornecer soluções teóricas de problemas simples para validar métodos aproximados [3].

2.1.2 Princípios gerais do Método de Elementos Finitos

O princípio base do Método de Elementos Finitos (MEF) é encontrar a solução para um problema complexo, utilizando um problema simplificado. Uma vez que foi substituído o problema original por um simplificado, ao encontrar a solução, só estamos a encontrar a solução aproximada e nunca a solução real. As ferramentas matemáticas atuais não são suficientes para encontrar a solução exata (e por vezes até a aproximada) da maioria dos problemas. Por isso, e na ausência de outro método mais conveniente para encontrar as ditas soluções, o MEF é a escolha mais assertiva para análise de sistemas complexos. No entanto, com o MEF sempre será possível refinar e aproximar a solução final das condições reais [7].

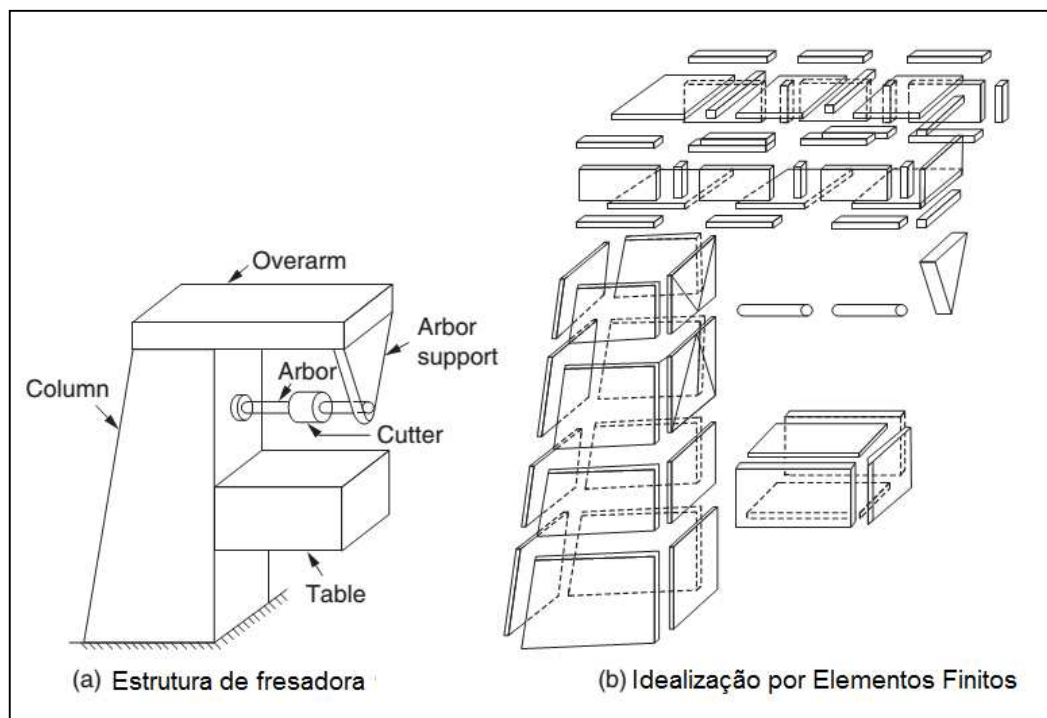


Figura 1 Exemplo da aplicação de Elementos finitos [7]

No MEF, a região de cálculo é considerada pela constituição de pequenas regiões e sub-regiões interligadas entre elas. Pode-se tomar como exemplo a Figura 1, onde é possível ver o exemplo de uma estrutura de uma fresadora. Partindo do modelo inicial é quase impossível determinar a resposta exata (tensões e deslocamentos) da máquina em condições de trabalho (cargas). Ao

analisar esta máquina como somatório das suas partes, será possível calcular cada parte e interação com a vizinhança em separado, obtendo-se assim uma solução para o problema principal [7].

Pode-se ir muito mais além do que a divisão apresentada na Figura 1, podendo não só separar-se seções da máquina e tratar as geometrias simples como elementos finitos, mas sim dividir completamente as estruturas como é mostrado na Figura 2. Deve-se sempre referir que existe um equilíbrio para esta divisão, quanto mais pequeno o elemento, mais aproximada é a solução, mas também maior será também o volume de cálculo do estudo pelo MEF.

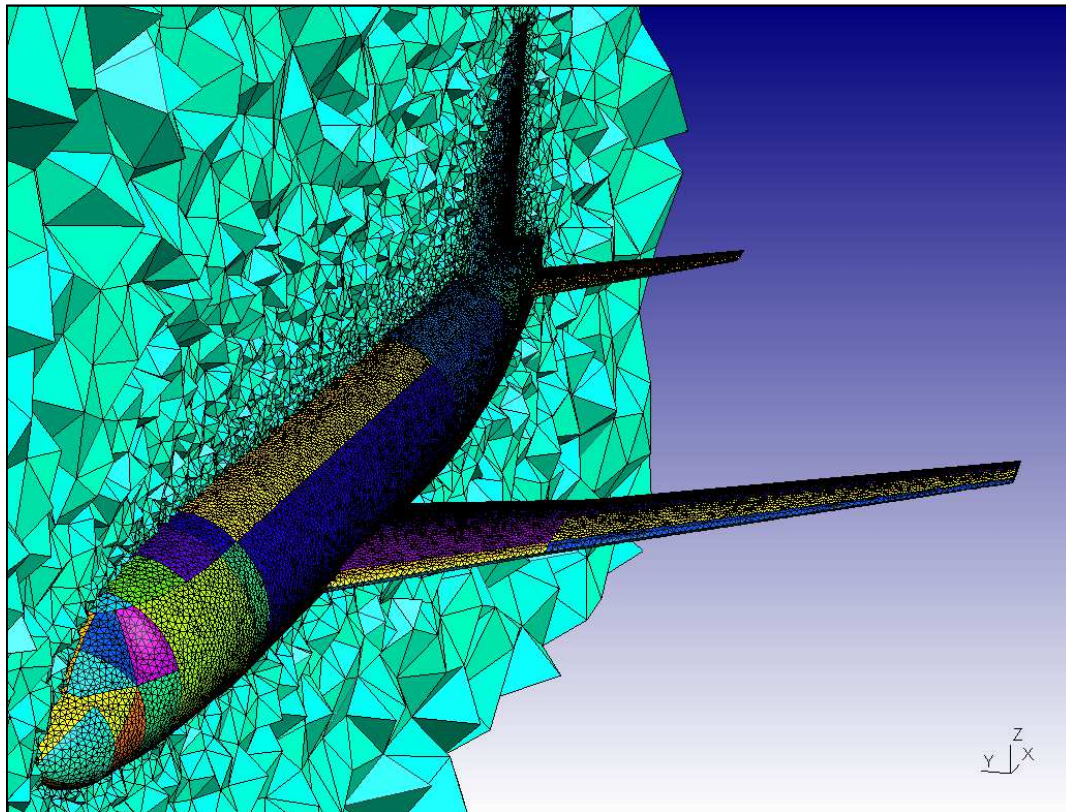


Figura 2 Elementos Finitos para o estudo de avião A319 [8]

Independentemente da complexidade que se pretenda adotar durante a análise, existirão sempre duas definições primordiais:

- Elemento: Porção do objeto de estudo limitada por uma vizinhança. Normalmente é-lhe atribuído propriedades físicas ou mecânicas;
- Nó: Intersecção, fronteira entre elementos. Ponto do domínio no qual a/as variáveis de campos vão ser calculadas diretamente pelo sistema de equações criado pelo MEF [6].

Pode-se ver um exemplo da aplicação das definições mencionadas na Figura 3.

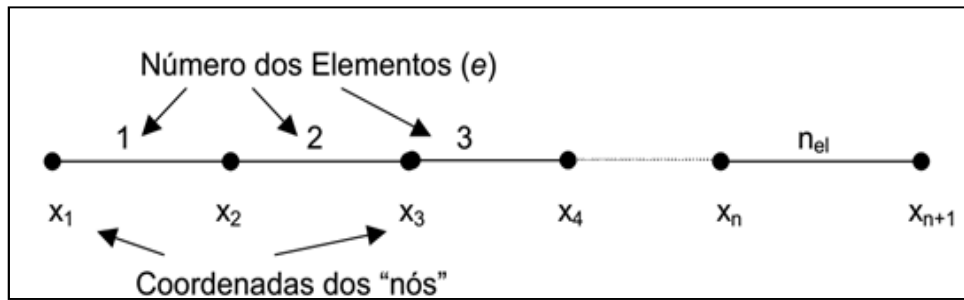


Figura 3 Elementos unidimensionais e nós

Um problema de engenharia pode ser formulado como um sistema contínuo ou discreto. Num modelo contínuo a resposta é baseada em equações diferenciais. Num modelo discreto, a resposta do sistema baseia-se numa solução com um número finito de variáveis [2].

Durante a aplicação do MEF numa determinada análise, e partido do princípio que o modelo inicial pode ser descrito por um número finito de variáveis, devem seguir-se as seguintes etapas:

1. Idealização do sistema enquanto contribuição de vários elementos;
2. Estabelecimento de equilíbrio em cada elemento em termos das variáveis de interesse;
3. Contabilização da contribuição de todos os elementos, tendo em conta as ligações entre elementos para as variáveis do problema;
4. Cálculo da resposta, tendo em conta a solução do sistema de equações, obtendo-se as variáveis do problema;
5. Podem obter-se outras informações para além destas variáveis, para cada elemento [2].

Tomemos como ponto de partida o elemento triangular de 3 nós (mais à frente serão enumerados os diferentes tipos de elementos). Cada nó encontra-se numerado e é sobre estes nós que serão aplicadas as ações exteriores e as restrições. O número total de equações algébricas necessárias para uma análise pelo MEF é o produto do grau de liberdade do elemento e da quantidade dos nós da estrutura. É sobre os nós que variáveis como os deslocamentos são calculadas pela análise do MEF.

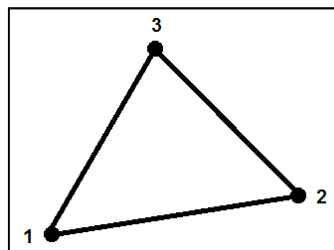


Figura 4 Elemento Triangular de 3 nós

O método de determinação das mesmas variáveis no interior do elemento representa um dos princípios básicos do MEF: as variáveis nodais. São usadas para aproximar as variáveis nos pontos não nodais por intermédio de funções de interpolação. Para o triângulo em causa temos:

$$\phi(x, y) = N_1(x, y)\phi_1 + N_2(x, y)\phi_2 + N_3(x, y)\phi_3 \quad (1)$$

ϕ_1, ϕ_2, ϕ_3 são valores da variável de campo nos três nós e N_1, N_2, N_3 são as funções de interpolação. No MEF o objetivo é determinar as variáveis de campo nos nós dos elementos (sistema de equações global), sendo as funções de interpolação usadas para determinar as variáveis nos restantes pontos do domínio. As funções de interpolação são usualmente polinomiais, construídas para satisfazer certas condições nos nós exteriores. Estas são pré-determinadas e conhecidas em função das variáveis de campo e descrevem a variação da variável de campo dentro dos elementos[6].

Se o domínio for um corpo fino em estado plano de tensão, a variável de campo ϕ é um vetor de deslocamentos e as variáveis nodais são os deslocamentos u_i e v_i ($i=1, 2, \dots$, numero de nós). O elemento finito triangular terá assim 6 graus de liberdade ($3 \text{ nós} \times 2 \text{ variáveis nodais}$). Nas ligações nodais, o valor das variáveis de campos é igual para os nós de cada elemento ligado a esse ponto (também válido para as fronteiras entre elementos), o que garante a continuidade das variáveis de campo no domínio. Esta característica evita vazios no domínio, que seriam fisicamente inaceitáveis (num problema estrutural representariam a separação de material). No entanto, não há usualmente continuidade nas derivadas das variáveis de campos. Por exemplo, em problemas estruturais, a deformação (definida em termos de primeira derivada da variável de campo deslocamento), não é contínua nas fronteiras entre elementos [9].

2.1.3 Potencialidades

Em função do anteriormente exposto, o MEF tem como potencialidades:

- Permitir a análise de estruturas de grau hiperestático;
- Não é necessário conhecer todos os pontos de um dado problema de estudo. Basta conhecer o comportamento dos pontos que limitam a fronteira dos sólidos em estudo e através da matriz de rigidez e a sua relação de distâncias obter o comportamento aproximado dos pontos desconhecidos;
- Aplicabilidade em simulação computadorizada de comportamentos estáticos e dinâmicos de estruturas e máquinas;
- Apesar de o MEF ter grande aplicabilidade nos campos da mecânica estrutural, também tem sido usado com sucesso em outros ramos da engenharia como: condutividade térmica,

dinâmica de fluidos, fluxos de infiltração (*seepage flow*), e determinação de campos elétricos e magnéticos;

- O MEF pode ser usado para obter o valor numérico de problemas de equações diferenciais [7];
- Utilizado para validar outros modelos matemáticos;
- Simular o processo de fabricação do produto, reduzindo os custos (ex: fundição);
- Utilizando um computador, podemos corrigir e testar um projeto, evitando assim a construção de um protótipo e testes despectivos, reduzindo assim os custos;
- A utilização de um sistema CAE (Engenharia Auxiliada por Computador ou *Computer Aided Engineering*) permite eliminar ou reduzir significativamente a quantidade de protótipos de teste a serem construídos, reduzido assim o tempo de concepção do produto;
- Resolução de sistemas complexos em termos de geometria, condições fronteira e carregamento.

2.1.4 Aplicações do Método de Elementos Finitos

O MEF é utilizado atualmente em diferentes áreas da engenharia. Podem-se enumerar alguns exemplos:

Análise estrutural (estática/dinâmica, linear/não linear)

Estruturas submetidas a carregamentos cíclicos, em frequências próximas aos modos naturais da estrutura, podem apresentar elevadas amplitudes de vibração. Os esforços dinâmicos provenientes dessas vibrações podem ser significativos, gerando solicitações à estrutura acima do previsto no cálculo estático. Estruturas de torres e prédios altos, submetidas a cargas provenientes da ação do vento, são exemplos de estruturas onde as cargas dinâmicas são significativas [10]. Na Figura 5 pode-se verificar as zonas críticas (a vermelho) sujeitas a maior tensão numa construção pilar-estrutura.

Simulação mecânica, aeroespacial e automóvel

A Figura 6 representa o modelo de um chassi gerado num *software* de CAD. Nesse caso o objetivo do engenheiro de cálculo é determinar a deformação do chassi. Para cumprir essa tarefa ele precisará de conhecer a rigidez do conjunto (chassi) e as cargas que atuam nele. A rigidez desse conjunto é determinada a partir da rigidez de cada um dos seus elementos. A diferença entre este exemplo e o exemplo da estrutura de vigas, é que na estrutura de vigas a subdivisão é óbvia: os elementos de viga são conectados naturalmente entre si nas suas extremidades, formando os nós [11].

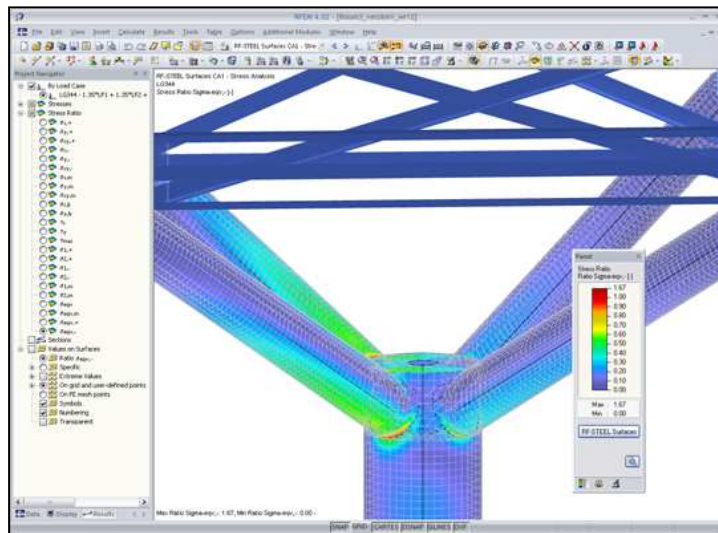


Figura 5 Exemplo de pilar de estrutura [10]

No caso do chassi, essa subdivisão deve ser proposta pelo engenheiro com base no conhecimento dos conceitos que ele tem do MEF. Ele cria uma malha de elementos finitos como mostra a Figura 6, de forma que a partir do conhecimento da rigidez de cada elemento do modelo, o *software* contabiliza a rigidez do conjunto. Assim, com o auxílio do computador, que resolve facilmente sistemas de equações algébricas com milhares e milhões de linhas, são calculadas, a deformação do conjunto, o nível de esforços internos em cada porção da estrutura e as tensões [11].

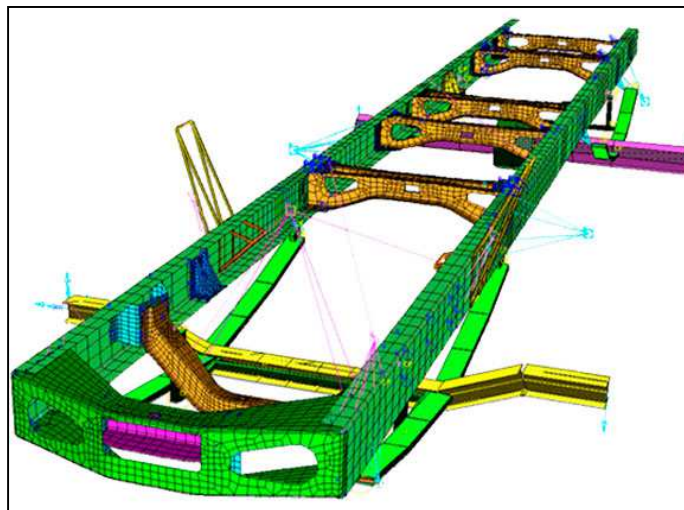


Figura 6 Chassi de autocarro [11]

Aplicações de escoamentos térmicos em sólidos e fluidos, comportamento de fluidos

Através da simulação numérica do fluxo de material é possível prever diversos fenómenos que podem ocasionar problemas ao processo tais como falhas estruturais, desgaste acentuado de superfícies ou fadiga da estrutura.

A previsão do fluxo de partículas e a colisão das mesmas entre si e com as superfícies existentes, como mostrado na Figura 7, possibilita um ajuste preciso de posição e ângulo de placas direcionadoras e demais acessórios dos elementos de modo a otimizar o fluxo conforme a necessidade, de forma a aumentar o rendimento e a vida útil dos equipamentos.

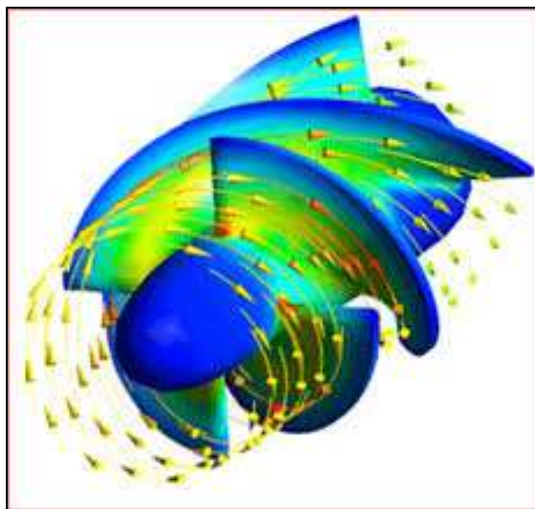


Figura 7 Campo de tensões em um indutor axial usando campo de pressão de uma análise de CFD [12]

Comportamentos eletromagnéticos

Todos os programas para campos eletromagnéticos obtêm a solução realizando cálculos em 3D, mas muitos modelos em análise têm na sua constituição planos de simetria que podem ser utilizados para reduzir o volume de trabalho. Por isso muitos programas de análise de campos eletromagnéticos representam o modelo em 2D ou em corte (ver na Figura 8 um exemplo de corte por simetria nos campos magnéticos) [13].

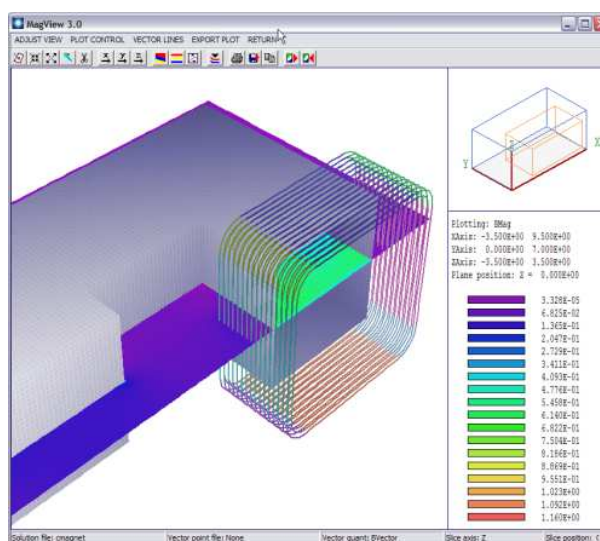


Figura 8 Análise de campo magnético usando MagView [13]

O MOSFET (*Metal Oxide Semiconductor Field-Effect Transistor* Figura 9) é o componente semiconductor mais implementado hoje em dia, é o primeiro bloco construtor para a maioria dos processadores, memórias e circuitos integrados digitais comercializados. Nas últimas décadas este equipamento experimentou grandes progressos e estudo, e hoje é fabricado com tamanhos de 90 nm e menor ainda. No modelo da Figura 9 é calcula as características DC no transístor usando a física dos semicondutores. Em operações normais, o sistema de simulação ativa o transístor aplicando a corrente no elétrodo. Quando a voltagem de entrada aumenta, a corrente de escoamento aumenta até atingir a saturação. A saturação da corrente depende da voltagem de entrada. O modelo contem dois varrimentos paramétricos dimensionais onde é computado o escoamento da corrente à diferentes voltagens de entrada. Com esta simulação consegue-se as características do transístor e o diagrama de escoamento elétrico [14].

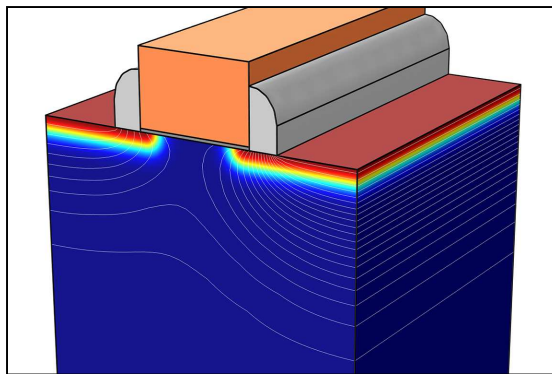


Figura 9 Transístor MOSFET [14]

Biomédica e Biomecânica

Atualmente, existe um conhecimento muito reduzido acerca dos efeitos associados à deformação corporal causada pelo crescimento de tumores cerebrais. Simulações computadorizadas pelo MEF têm o potencial para calcular as ditas deformações. Durante a simulação, são localizadas as grandes deformações dentro dos tecidos moles do cérebro. Este conhecimento será de grande significado na neurociência e neurologia, particularmente pela quantidade de casos de tumores agressivos, e permitirá o planeamento da terapia a ser aplicada, assim como a simulação de casos operatórios. A malha de Elementos Finitos usa na sua vizinhança um tumor em crescimento torna-se rapidamente obsoleta, o que requer uma constante atualização da malha [15]. Métodos livres de malhagem são capazes de tratar a grande expansão e deformação que ocorre, mas poucos métodos são tão confiáveis quanto o MEF. Nestas circunstâncias são definidas 3 zonas e abordagens:

- As zonas de pouca deformação utilizarão um modelo MEF;
- Nas zonas de grande deformação do volume cerebral, é necessário utilizar uma aproximação por malha mista;

- O método livre de Galerkin [16] é utilizado nas zonas de grandes deformações do volume do tumor.

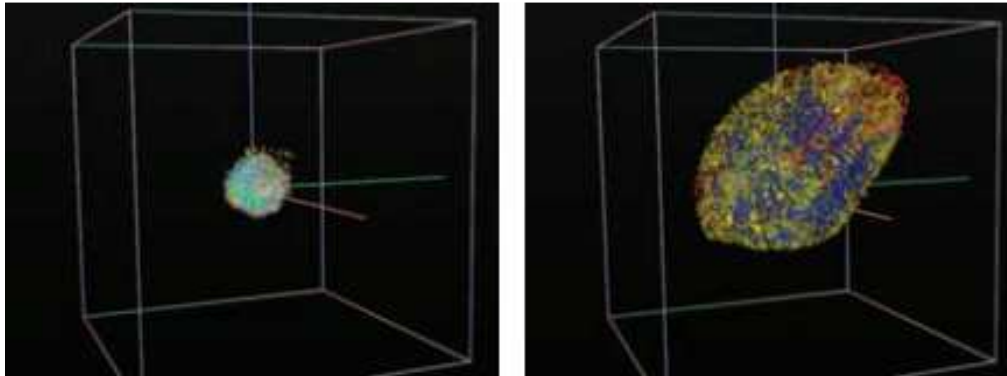


Figura 10 Simulação 3D do crescimento de um tumor cerebral [17]

2.1.5 Tipos de Elementos Finitos para análise estrutural

Uma análise de MEF é genericamente dividida em três fases: pré-processamento, processamento e pós-processamento. Na fase do pré-processamento o modelo é simplificado, escolhendo de antemão a divisão a ser utilizada e o tipo de Elemento Finito mais adequado que melhor traduza o comportamento do modelo real. Assim sendo, durante esta fase de pré-processamento, e dependendo do domínio em estudo, surgem duas abordagens: a escolha de uma estrutura simplificada (elementos unidimensionais), ou a criação de uma malha de elementos.

No caso das malhas de elementos, é comum a gradação do tamanho dos elementos, com elementos de tamanho inferior ou de grau superior em zonas de mudança abrupta das variáveis de campo (zonas de modificações bruscas na geometria ou próximo de cargas pontuais para problemas estruturais), enquanto se pode usar uma malha mais grosseira em zonas cujas variáveis de campo são aproximadamente constantes [18].

A escolha do tipo de elementos numa análise pelo MEF depende da disposição física do domínio a modelar e do grau de precisão pretendido para os resultados. Também a escolha entre análises a 1, 2 ou 3 dimensões deve ser considerada nesta altura [7].

De seguida são definidos de alguns dos elementos mais utilizados na formulação MEF para análise estrutural (Figura 11) [6]:

- Os **elementos unidimensionais** dividem-se em elementos de barra, viga e estrutura. Estes elementos têm área de secção bem definida e, apesar da possibilidade de formulação com área variável, é comum usar-se elementos de secção constante. São utilizados para a modelação de estruturas reticuladas articuladas ou contínuas. O elemento mais simples é o

elemento de barra para esforços axiais de dois nós, um em cada extremidade, e com funções de interpolação lineares. Elementos de ordem superior como o elemento de barra de três ou mais nós podem também ser formulados, sendo que já possuem funções de interpolação quadráticas, cúbicas ou superiores;

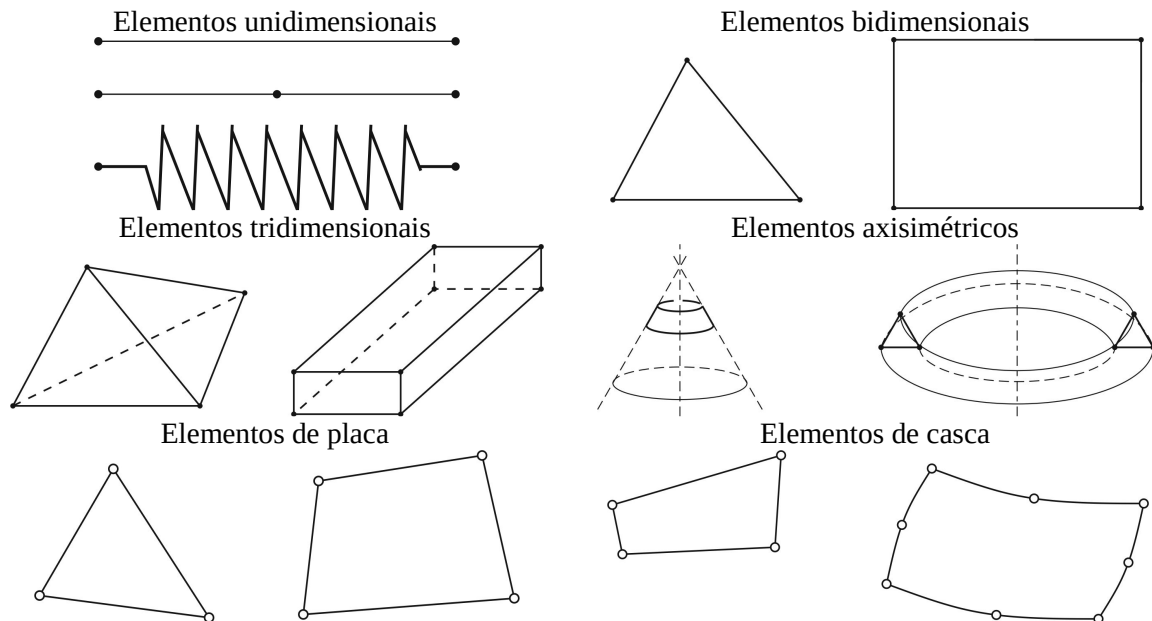


Figura 11 Elementos comumente utilizados na formulação MEF [6]

- Os **elementos bidimensionais** (ou planos) encontram-se limitados a carregamentos no seu plano (condições de estado plano de tensão ou de deformação) e apresentam forma triangular ou retangular. Os elementos mais simples apresentam nós em cada um dos seus vértices (funções de interpolação lineares para cada variável de campo) e arestas planas, embora estejam atualmente disponíveis formulações de elementos com os nós posicionados ao longo das arestas (funções de interpolação de ordem superior), com arestas curvas e espessura variável;
- Os **elementos tridimensionais** mais comuns apresentam 4 (tetraedro) 5 (pentágono) ou 6 faces (hexaedro) e são usados para análises tridimensionais de tensões. Na sua forma mais elementar, também estes elementos têm nós apenas nos seus vértices e arestas retilíneas, enquanto elementos de ordens superiores podem incluir nós nas arestas, nas faces ou no interior dos elementos, para além de faces curvas;
- Os **elementos axisimétricos** são formulados pela rotação de 360° de uma forma bidimensional em torno do eixo de revolução de um corpo, e permitem a modelação de estruturas em que a geometria, carregamentos e condições fronteira são simétricos relativamente ao seu eixo de rotação;

- Os **elementos de placa e casca** modelam corpos de espessura fina. Os elementos de placa destinam-se a modelar placas planas com carregamentos fora do plano, resultando na sua flexão e respetiva rotação segundo 2 eixos, e também na sua deflexão transversa. As cascas são definidas pela sua espessura e superfície média, representada por uma superfície curva no espaço. As cascas suportam principalmente esforços de membrana (no plano), mas também resistem a esforços transversos de flexão como os elementos de placa. Em termos de geometria, podem exibir curvatura num ou em dois eixos [6].

2.1.6 Técnica do Método de Elementos Finitos

De uma forma mais detalhada ao anteriormente exposto, as etapas a considerar numa análise MEF são as seguintes [6]:

Pré-processamento

- Definição do domínio geométrico do problema;
- Definição do tipo/tipos de elementos a utilizar > **formulação do elemento**;
- Definição das propriedades materiais dos elementos;
- Definição das propriedades geométricas dos elementos (comprimento, área);
- Definição das conectividades entre elementos (malha do modelo);
- Definição das restrições às variáveis de campo (condições fronteira);
- Definição dos carregamentos (forças, fluxos de calor, etc...).

As definições anteriores são de extrema importância na análise MEF, dado que qualquer consideração errónea pode levar à apresentação de resultados aparentemente corretos, mas na realidade a solução do problema em análise e poderá estar muito longe da realidade.

Como passo de grande importância no pré-processamento tem-se a formulação do elemento finito, dividida em 6 sub-etapas tal como descreve Campilho [6].

Etapa 0 – Definir as variáveis de campo relevantes para o problema a resolver. Em problemas estruturais as variáveis são deslocamentos (e nalguns casos também rotações), diferindo de acordo com o tipo de estrutura a estudar. Por exemplo, por cada nó do elemento tem-se:

- Estrutura reticulada articulada no plano: 2 deslocamentos;
- Estrutura reticulada contínua no plano: 1 deslocamento e uma rotação;
- Estruturas contínuas planas: 2 deslocamentos;
- Estruturas tridimensionais sólidas: 3 deslocamentos;

- Estruturas contínuas orientadas (placas e cascas): deslocamentos e rotações dependentes do tipo de elemento.

Etapa 1 – Definir o elemento na sua forma, localização dos nós e variáveis associadas a cada nó, pela criação dos vetores

$\mathbf{a}_e$ ou $\mathbf{A}_e \rightarrow$ vetor de variáveis nodais

$\mathbf{f}_e$ ou $\mathbf{F}_e \rightarrow$ vetor de forças nodais.

No presente texto, utiliza-se nomenclatura em minúsculas para vetores e matrizes respeitantes ao elemento, ou em maiúsculas caso seja efetuada a transformação das variáveis para coordenadas globais (nos elementos em que tal seja necessário).

Etapa 2 – Expressar o campo de deslocamentos no interior do elemento u_e em função das variáveis nodais $\mathbf{a}_e$ e das funções de interpolação $\mathbf{N}$. As funções de interpolação são definidas para cada nó, e são mais comuns polinómios lineares, quadráticos e cúbicos devido à maior simplicidade na manipulação, embora também possam ser usadas séries trigonométricas. O campo de deslocamentos é definido por um número de equações igual ao número de variáveis de campo em cada nó (por exemplo num problema tridimensional são utilizadas três funções, cada uma expressando o deslocamento segundo um eixo coordenado). Assim, para um deslocamento segundo x tem-se

$$u_e = \mathbf{N} \mathbf{a}_e. \quad (2)$$

Etapa 3 – Definir o vetor de deformações $\boldsymbol{\epsilon}$ em função da matriz de deformação $\mathbf{B}$ e de $\mathbf{a}_e$

$$\boldsymbol{\epsilon}_e = \mathbf{B} \mathbf{a}_e. \quad (3)$$

Etapa 4 – Definir o vetor de tensões $\boldsymbol{\sigma}$ em função da matriz de elasticidade $\mathbf{D}$ e de $\boldsymbol{\epsilon}$

$$\boldsymbol{\sigma}_e = \mathbf{D} \boldsymbol{\epsilon}_e. \quad (4)$$

Etapa 5 – Definir a matriz de rigidez do elemento, $\mathbf{k}_e$, e o vetor de forças nodais, $\mathbf{f}_e$ (ou $\mathbf{K}_e$ e $\mathbf{F}_e$ após conversão para coordenadas globais), através de

$$\mathbf{k}_e = \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV, \quad \mathbf{f}_e = \mathbf{k}_e \mathbf{a}_e. \quad (5)$$

A expressão apresentada para a matriz de rigidez do elemento, $\mathbf{k}_e$, relaciona as forças nodais, $\mathbf{f}_e$, e variáveis de campo, $\mathbf{a}_e$, de um elemento finito, e pode ser obtida de diferentes formas dependendo

do tipo de elemento: por *Métodos Variacionais*, *Métodos dos Resíduos Ponderados* ou por equilíbrio do sistema [19]. Neste último caso, $\mathbf{k}_e$ é obtida por expressão matricial das equações de equilíbrio num elemento. Este método é o mais viável para elementos unidimensionais como molas e barras [6].

Obtenção da Solução

Durante esta fase, o *software* de MEF (ou o utilizador) monta as equações algébricas em forma matricial ou forma o sistema de equações global a partir das matrizes de cada elemento, cuja resolução permite obter $\mathbf{a}_e$. O resultado a nível de um elemento é um sistema de equações do tipo [19]

$$\begin{bmatrix} k_{11} & k_{12} & \cdots & k_{1n} \\ k_{21} & k_{22} & \cdots & k_{2n} \\ \vdots & & & \vdots \\ k_{n1} & k_{n2} & \cdots & k_{nn} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{Bmatrix} = \begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{Bmatrix}, \quad (6)$$

ou na forma matricial

$$\mathbf{k}_e \mathbf{a}_e = \mathbf{f}_e \text{ ou } \mathbf{K}_e \mathbf{A}_e = \mathbf{F}_e \text{ (após transformação de coordenadas)}. \quad (7)$$

O sistema global de equações, relativo à totalidade dos elementos finitos da estrutura para sistemas estáticos, fica

$$\mathbf{K} \mathbf{A} = \mathbf{R} + \mathbf{F}, \quad (8)$$

onde $\mathbf{K}$ é a matriz de rigidez global da estrutura, $\mathbf{A}$ é o vetor global das variáveis de campo, $\mathbf{R}$ é o vetor global das reações na estrutura e $\mathbf{F}$ é o vetor global das forças nodais. É possível provar que $\mathbf{K}$ é singular, já que o seu determinante é nulo, o que resultaria no movimento de corpo rígido da estrutura. Para evitar este facto, as condições fronteira (deslocamentos ou rotações prescritas) são incluídas em $\mathbf{A}$ [20]. Como a expressão (8) tem incógnitas em $\mathbf{R}$ e $\mathbf{A}$, o sistema é inicialmente condensado por consideração apenas das variáveis desconhecidas em $\mathbf{A}$. Os valores de $\mathbf{A}$ são depois utilizados para extrair do mesmo sistema de equações as variáveis secundárias em $\mathbf{R}$. Como é comum estes sistemas de equações serem constituídos por dezenas de milhar de equações, e como $\mathbf{K}$ é simétrica e em banda, é possível utilizar métodos de resolução otimizados que minimizam o espaço de armazenamento e esforço computacional. Para problemas estáticos lineares é utilizado o método de resolução frontal (*wave front solver*) baseado no Método de Gauss [6].

Pós-Processamento

Esta fase consiste na análise e avaliação dos resultados da simulação pelo MEF. Os *softwares* comuns já possuem rotinas de seleção de dados a apresentar, impressão e visualização dos resultados obtidos. Alguns exemplos de operações incluem [21]:

- Ordenação das tensões dos elementos por magnitude;
- Verificação do equilíbrio estático;
- Cálculo de coeficientes de segurança;
- Visualização da estrutura deformada;
- Animação do modelo (carregamento progressivo);
- Visualização em escala de cor de tensões ou temperaturas.

É também de extrema importância a aplicação de juízo crítico para averiguar se os resultados são fisicamente aceitáveis [6] assim como a comparação dos resultados da análise por outros métodos.

2.1.7 Softwares disponíveis para análise estrutural

Como afirmado anteriormente, o Método de Elementos Finitos, apesar de ser uma aproximação bastante fiel aos casos reais, necessita da subdivisão da estrutura em análise num grande número de elementos de maneira que os resultados se assemelhem em maior grau à resposta real do objeto em análise. Isto acarreta um aumento no volume de cálculo necessário sendo indispensável o auxílio da informática e da capacidade de processamento dos computadores.

Ao longo dos anos, as empresas de *software* e programadores no geral têm apostado na criação de códigos e programas que auxiliam os projetistas no cálculo nas análises mais complexas. Estes códigos utilizam não só o MEF, mas também outros modelos matemáticos de maneira a agilizar o processamento de grandes estruturas.

Atualmente existe uma grande escolha de programas para cálculo e simulação, com licenças que variam entre o *open-source* (código aberto e grátis) até licenças que estão fora do alcance do utilizador pontual. Esta diferença na acessibilidade é principalmente por características tais como:

- Capacidades de desenho assistido por computador (CAD);
- Possibilidade de simulação;
- Comparação de resultados;
- Possibilidade de trabalho multiplataformas (trabalho em rede);

- Disponibilidade de mais que um código para cálculo;
- Facilidade de leitura e criação de vários tipos de ficheiros (compatibilidade);
- Tipo de interface com o usuário (linha de texto ou interfase gráfica);
- Certificação sobre testes efetuados.

2.1.7.1 *Softwares comerciais de uso genérico*

Dentro da diversa gama de *softwares* profissionais para o cálculo de estruturas por MEF destacam os seguintes:

NASTRAN

O NASTRAN (NASA STRUCTURAL ANALYSIS) é um programa aberto e gratuito, amplamente utilizado e reconhecido internacionalmente, que permite a solução de modelos por elementos finitos. A linguagem do seu código baseia-se no FORTRAN. Tem como principais vantagens [22]:

- Disponível para trabalho em diferentes sistemas operativos.
- Processamento de modelos geométricos e numéricos.
- Cálculo de modelos estáticos, dinâmicos e transientes, lineares e não lineares.
- Permite a adição de módulos programáticos.
- Criado para análises sensitivas e otimização.
- Prototipagem Virtual. Otimização de modelos, reduzindo volumes desnecessários.
- Cálculo de resposta acústica, fadiga e delaminação de placas.
- Boa aproximação a modelos em material compósito.
- Dispõe de biblioteca com características e propriedades de materiais.
- Criação de super elementos. Após obtenção da resposta de um modelo por MEF, este pode ser utilizado como um todo (um único elemento) numa nova análise na vizinhança de elementos com resposta desconhecida.

ANSYS®

Programa de modelação e simulação em vários ramos da física, muito aceitado em Portugal e no mundo no que toca à rapidez de cálculo e aproximação de resultados. Características [23]:

- Possibilidade de trabalho por diagrama de fluxo, permitindo aos resultados de um domínio em um dado modelo, servir como dados de entrada num outro domínio de um outro modelo (efeito dominó);
- Possibilidade de manipulação dos modelos de estudo através de *Drag-and-Drop* de condições de estudo (propriedades físicas externas/vizinhança);

- Possibilidade de baterias de estudos sequenciais a um mesmo modelo;
- Possibilidade de mudança das condições de análise ainda antes de ser concluída a simulação ou análise;
- Possibilidade de importação de modelos CAD de diferentes fontes;
- Cálculo e modelação paramétrica;
- Possibilidade de ajustar a malha de elementos dos modelos em análise segundo os critérios pretendidos (refinamento à medida).

ABAQUS®

O Abaqus® é um pacote de *software* comercial para análise por elementos finitos desenvolvido pela HKS Inc de Rhode Island, E.U.A. e agora comercializado sob a marca SIMULIA® marca da Dassault Systèmes S.A. As principais vantagens do *software* são [24]:

- Possibilidade de manipulação e simulação de modelos complexos com todas as ações em jogo;
- Cálculo de modelos estáticos, dinâmicos e transientes, lineares e não lineares;
- Amplamente adotado em ensaios de impacto (*Crash-Test*);
- Cálculo de resposta acústica, fadiga e delaminação de placas;
- Dispõe de biblioteca com características e propriedades de materiais (não disponível na versão standard) [25];
- Possibilidade de cálculo e simulação de fluxos de calor;
- Possibilidade de simulação de fluidos (CFD);
- Biblioteca extensa de elementos finitos nos mais diversos campos de aplicação;
- Possibilidade de programação de novos elementos finitos pelo utilizador;
- Variedade enorme de modelos de materiais.

SOLIDWORKS®

Software amplamente utilizado para desenho e simulação. Comercializado pela marca Dassault Systèmes S.A. e amplamente utilizado na indústria. Principais vantagens do *software* [26]

- Modelação em CAD;
- Possibilidade de cálculo e simulação de fluxos de calor;
- Possibilidade de simulação de fluidos;
- Dispõe de biblioteca com características e propriedades de materiais;
- Cálculo de modelos estáticos, dinâmicos e transientes, lineares e não lineares;
- Prototipagem virtual. Otimização de modelos, reduzindo volumes desnecessários;

- Cálculo de resposta acústica, fadiga e delaminação de placas (mas de difícil modelação de laminados).

INVENTOR®

Software da mesma produtora do Autocad® (Autodesk®). Cada vez com maior utilização na indústria. As características deste programa são [27]:

- Modelação em CAD;
- Dispõe de biblioteca com características e propriedades de materiais;
- Programação dos modelos de modo a agirem como as suas contrapartes reais (um modelo de motor programado com parâmetros reais, é simulado como um motor real);
- Disponibilidade de um grande número de elementos finitos para simulação (barra, placas, tetraedros, axisimétricos, etc.);
- Possibilidade de importação de modelos CAD de diferentes fontes;
- Vocacionado para modelos de grande porte;
- Cálculo de modelos estáticos, dinâmicos e transientes, lineares e não lineares;
- Ferramentas de otimização do modelo durante a simulação;
- Cálculo de resposta acústica e fadiga.

2.1.7.2 Códigos específicos

Para além dos *softwares* dedicados para cálculo e simulação profissional, existem também programas e códigos que podem ser implementados em diversas plataformas programáticas. É neste último ramo que o programa Maltab® atinge o seu expoente máximo devido às capacidades de programação modular, cálculo matricial e possibilidade de integração gráfica. Permite ainda a interação com outros programas de simulação e modelação, e aceita uma boa gama de tipos de ficheiros.

Com esta premissa em consideração, existe um leque de pequenos programas que trabalham sobre a plataforma Maltab® para cálculo por elementos finitos. A maioria destes programas está disponível para *download* no site da Maltab® Central [28]. Destacam-se os seguintes:

Matrix Structural Analysis (MSA) [29]

É um programa criado por Hossein Rahami da Universidade de Tehran-Iran em 2010. Este programa permite realizar análises elásticas de estruturas de ligações rígidas em vigas 2D e 3D. Trata-se de um script programático, no qual são introduzidas as variáveis no próprio ficheiro do tipo *.m. Após a entrada dos dados, o programa devolve como resultados as forças internas nos

elementos e momentos nas coordenadas globais, além de informar sobre o valor das reações e deslocamentos. Tem o inconveniente de requerer que o utilizador tenha conhecimentos de programação na linguagem do Matlab®, e de existir a possibilidade do usuário inserir em campos erróneos os dados de entrada no problema em análise. O programa carece de capacidades gráficas.

2D-Truss with GUI Program [30]

Um programa desenvolvido por Asare Stephen e Yan Lou, na China em 2009. Com este programa é possível realizar análise elástica e estática de barras em 2D, onde as cargas só podem ser aplicadas nos nós. Tem uma interface gráfica onde se pode realizar a entrada dos parâmetros da estrutura, com atualização instantânea a cada nova entrada. A possibilidade de erros na alimentação dos dados é mínima. Os resultados só podem ser visualizados na linha de comando do Matlab®. No entanto, o programa consegue fazer o desenho da estrutura deformada.

Beam and Truss Analysis utilities [31]

Programa desenvolvido por Roei Lahav em Israel em 2005. O programa permite realizar análise estrutural de barras e vigas em 2D. Os dados da estrutura são introduzidos num ficheiro *.m. Após este passo, o utilizador deverá escolher o tipo de análise pretendida selecionando a rotina (ficheiro) de maneira a obter as resultantes do cálculo. Os resultados são apresentados tanto em forma gráfica, como sobre o ambiente GUI. No entanto, a conceção do programa e da interface de utilizador (GUI) é pouco clara e desorganizada. O utilizador não possui meios para alterar nenhuma das propriedades da estrutura previamente definida.

Structure analysis [32]

Programa desenvolvido por Yasser Bigdeli em 2012. O programa permite realizar análise estática elástica a barras e vigas em 2D e 3D. É composto por um único ficheiro *.m e não possui capacidades gráficas, sendo toda a análise feita através da linha de comando do Matlab®. Carece de qualquer meio de alteração da estrutura uma vez o programa inicializado, sendo necessário iniciar novamente o programa para realizar qualquer alteração na análise em curso.

Matrix Analysis of Three Dimensional Bar Structures MABS3D [33]

Programa desenvolvido por Marcos Cessar Ruggeri em 2010. O programa permite a análise de elementos barra no espaço 3D. O programa é alimentado através de um formulário em ficheiro Excel onde são colocadas todas as características da estrutura em análise. Após o lançamento do programa o utilizador deve indicar o ficheiro *.XLS para análise. O programa devolve os resultados na linha de comando do Matlab® e desenha a estrutura com a distribuição de tensão axial. No entanto, carece da informação visual do local e intensidade das cargas aplicadas. A

estrutura deformada é pouco visível, uma vez que utiliza um fator de escala de 1. Uma vez que o utilizador não possui uma pré-visualização da estrutura a cada passo da criação (no ficheiro formulário), existe a possibilidade de introduzir erroneamente os dados de entrada.

Space Truss Systems as Linear Static Analysis [34]

Programa da autoria de Ali Ozgul criado em 2003. O principal foco do programa é análise de barras 3D. É composto por um único ficheiro onde deve ser colocados todos os dados de entrada para realizar a análise. O programa tem no seu código, 5 casos predefinidos de análise que o utilizador pode alterar dependendo das necessidades e semelhanças com o problema em estudo. Isto requer que o utilizador seja fluente na programação na linguagem Matlab[®], o que pode levar a erros na análise e na possibilidade de danificar o código principal. Não possui capacidades gráficas e os resultados são mostrados sobre a linha de comando do Matlab[®].

3D Truss Analysis\User Interface in FEM [35]

Programa criado por Balajee Ananthasayanam em 2005. O programa permite análise 2D e 3D de barras. O programa trabalha sobre a linha de comandos guiando o utilizador sobre os dados que devem ser introduzidos. No entanto, os passos para a introdução de parâmetros carecem de lógica, o que pode levar à introdução de dados incorretos. Como resultados reproduz graficamente a estrutura inicial e deformada, mas sem a informação das cargas aplicadas. Os resultados são apresentados na linha de comandos do Matlab[®].

MASTAN2 [36]

Programa desenvolvido por Ronald D. Ziemian (professor de Engenharia Civil na Universidade de Becknell) e William McGuire (professor de Engenharia Civil na Universidade de Emeritus Cornell) em 2010. O programa é capaz de analisar estruturas do tipo barra ou viga em 2D e 3D. Tem a capacidade de análise linear e não linear de 1ª e 2ª ordem. Toda a interface é realizada com recurso de GUIs, onde todos os dados são guardados em ficheiros do tipo *.mat. Este programa não requer que o utilizador tenha conceitos de Matlab[®], para além de realizar a iniciação do programa. A interface é estável e bem definida. O programa permite a visualização de gráficos de distribuição de forças e momentos, assim como a visualização da estrutura deformada. Os programadores continuam a fazer melhoramentos no programa, com perspetiva futura da realização de estudos dinâmicos de vibrações. O grande inconveniente deste programa é que apesar de ser um programa para Matlab[®], o seu código não é aberto, o que impossibilita a sua leitura e modificação.

Código de cálculo barra 3D – Campilho [37]

Programa desenvolvido por Raul Campilho (professor de Engenharia Mecânica no Instituto Superior de Engenharia do Porto). Criado em 2006. O programa permite a análise de estruturas do tipo barra 3D com ligações articuladas utilizando o MEF como base de cálculo. O utilizador é guiado na linha de comando, passo a passo durante a introdução dos parâmetros da estrutura. No final, os resultados aparecem de forma organizada na mesma linha de comando, dispostos na forma matricial ou tabular. No final a estrutura inicial e deformada é desenhada na janela de *plot* do Matlab®. Este programa consegue agilizar o processo de cálculo utilizando condensação matricial. Os resultados são perceptíveis apesar da limitação da janela da linha de comando. No entanto, carece de interface gráfica para além da visualização da comparação estrutura deformada com a inicial.

Structural Analysis Program by Stiffness Method (SABSM) [38]

Programa desenvolvido por Umar Draz Ahmad em 2013, para a obtenção do título de mestre em Estruturas de Engenharia Civil na Universidade de Londres. O programa permite a análise de estruturas tipo barra e viga, quer em 2D e 3D. O ambiente de interface com o utilizador é através de janelas programáticas. Cada dos um passos é claramente explicitado e encadeado numa sequência lógica. No final, o programa cria a estrutura inicial e a estrutura resultante (deformada), assim como os resultados da análise em forma tabular na linha de comandos, sendo estes as reações nos apoios, as forças locais nos elementos e os deslocamentos nodais.

2.2 Softwares de computação e visualização gráfica

Existem no mercado *softwares* que permitem ao utilizador pontual a criação dos seus próprios códigos programáticos para realizar as tarefas de cálculo durante as análises por MEF. O que é importante reter é que para além de fornecer uma linguagem programática de sintaxe simples, o programa deve oferecer outras mais-valias ao utilizador. São muito procuradas as seguintes características em programas de computação e visualização gráfica:

- Capacidade de cálculo vetorial, matricial, polinomial, etc;
- Rapidez de cálculo;
- Possibilidade de importação exportação de diferentes tipos de ficheiros;
- Capacidade de criação de gráficos (*plots*) de diferentes tipos;
- Possibilidade de interação com outros programas;
- Ser capaz de funcionar em diferentes máquinas e sistemas operativos;
- Possibilidade de criação de interfaces gráficas para os programas a ser criados.

2.2.1 Características gerais dos *softwares* existentes no mercado

De seguida serão enumerados alguns dos *softwares* existentes no mercado que permitem a criação de códigos e programas específicos pelo utilizador, sendo apresentada uma breve descrição sobre os mesmos.

Scilab [39]

Scilab é um programa gratuito em *Open-Source* para computação numérica que permite uma potente interface gráfica ao utilizador de diferentes áreas científicas. Está disponível para diferentes plataformas (Windows[®], Linux, Mac OS X[®]). As características principais desta plataforma programática são:

- Perto de 1700 funções matemáticas para o uso na engenharia e matemática;
- Visualização de gráficos em 2D e 3D;
- Biblioteca de algoritmos de cálculo para funções contínuas e descontínuas;
- Ferramentas de suporte à estatística;
- Algoritmos de visualização e filtragem de sinais;
- Possibilidade de simulação de sistemas mecânicos, hidráulicos e elétricos;
- Cálculo diferencial.

Octave [40]

Melhor chamado de GNU Octave. Criado em 1992 por John Eaton na Universidade de Wisconsin-Madison. É um programa gratuito em *open-source*, que possui capacidades de cálculo numérico para problemas lineares e não lineares. É o concorrente direto do Matlab[®] em termos de poder de cálculo e desenvolvimento. As suas características principais são as seguintes:

- Possibilidade de instalar os módulos requeridos pelo utilizador, ajustando o programa à sua medida;
- Possibilidade de instalação em Windows[®], MacOS[®] e Linux;
- Grande número de funções matemáticas disponível na biblioteca;
- Possibilidade de criação de gráficos 2D e 3D, e configurar o refinamento de malha no caso de superfícies;
- Possibilidade de importação de ficheiros de diferentes plataformas programáticas;
- Cálculo diferencial;
- Funciona melhor na plataforma Windows[®].

Matlab® [41]

Programa grandemente apreciado e com maior comunidade de utilizadores. Disponível na vertente comercial. Linguagem programática semelhante à do Fortran. Algumas das características do programa são:

- Grande número de fontes, tutoriais e guias;
- Referência no que toca a capacidades programáticas e de cálculo;
- Visualização de gráficos em 2D e 3D;
- Grande número de funções matemáticas disponíveis na biblioteca;
- Possibilidade de importação de ficheiros de diferentes tipos;
- Vocacionado para a computação matemática;
- Atualmente existem módulos como o ToolBox NuPAD, que permitem cálculo simbólico;
- Capacidade de criação de janelas programáticas para interface com o utilizador GUI;
- Simulação gráfica de multi-domínios;
- Capacidade de cálculo vetorial, matricial, polinomial, paramétrica, etc.;
- Ferramentas de suporte à estatística.

FlexPro® [42]

Software criado pela Weisang GmbH, disponível na vertente comercial. As características desta plataforma programática são:

- Linguagem de programação FPScript;
- Possibilidade de criar e visualizar *plots* e GUIs;
- Possui assistentes para cada área de trabalho;
- Guarda todas as variáveis sob um mesmo ficheiro (não dá para modificar);
- O ponto anterior permite a criação de relatórios *multi-data*;
- Cálculo matricial;
- Permite a importação e exportação de vários tipos de ficheiros;
- Permite visualização de vídeos;
- Grande leque de opções programática.

Sage [43]

Criado a partir de mais de 100 *softwares open-source*, não pretende copiar o Matlab®, mas melhorou a linguagem programática. Algumas características desta plataforma são:

- *Shell* interativa e permite manipulação;

- Cálculos básicos de Álgebra;
- *Plot* 2D e 3D;
- Possibilidade de criação de GUIs;
- Cálculo de grupos finitos (não confundir com elementos finitos);
- Permite a instalação de módulos programáticos.

2.2.2 Seleção do *software* a utilizar no desenvolvimento deste trabalho

Com as características dos *softwares* enumerados na secção anterior, a seleção do *software* programático recai sobre o Matlab[®]. Os principais motivos para esta escolha recaem sobre as ferramentas de apoio à criação de janelas programáticas para interface com o usuário, à grande comunidade existente em fóruns, a acessibilidade da linguagem programática e a capacidade gráfica para o desenho das estruturas em análise.

2.2.3 Funcionalidades pretendidas para o *software* a desenvolver

Uma vez selecionado o *software* Matlab[®] como plataforma base na programação e tendo em conta as características exigidas para uma análise por Método de Elementos Finitos (MEF), as características no software a desenvolver serão as seguintes:

- Capacidade de cálculo por elementos finitos;
- Interface gráfica para interação com o utilizador;
- Pré-visualização da estrutura em análise;
- Capacidade de análise de elementos barra e viga quer em 2D ou 3D;
- Possibilidade de introduzir propriedades mecânicas diferentes a cada elemento dentro de uma mesma estrutura em análise;
- Possibilidade de manipulação das pré-visualizações da estrutura;
- Obter gráfico da estrutura deformada;
- Obter gráfico da distribuição de tensões na estrutura em análise por elementos;
- Visualização da posição das cargas e das reações na pré-visualização da estrutura em análise;
- Apresentar os resultados sobre forma tabular;
- Capacidade de alterar as condições de análise de uma estrutura após obtenção de resultados sem necessidade de reiniciar a análise;
- Exportar os resultados, criando um relatório da análise num ficheiro em Excel;
- Possibilidade de gravar o estado da análise e continuar noutra sessão;

- Possibilidade de importação dos dados de entrada.

No capítulo 4 será explicado o modo de operação do programa criado com as características descritas, assim como a validação do programa.

3 Formulação dos elementos finitos a implementar

Segue-se neste capítulo a definição e formulação dos elementos finitos que serão disponíveis no programa desenvolvido. Na formulação foi seguida a abordagem tradicional tal como explica Campilho [6].

3.1 Elementos de barra 2D

O elemento barra 2D é caracterizado pela ligação de dois elementos com rótula sem fricção. Quando sujeito a ações exteriores, estas são sempre aplicadas nas suas extremidades. Este elemento está sujeito a esforços de tração/compressão pura. No caso de uma análise 2D em coordenadas globais, este tipo de elemento possui 4 graus de liberdade; dois deslocamentos ortogonais em cada nó.

3.1.1 Formulação linear

Etapa 0 – O elemento de barra de 2 nós destina-se a resolver problemas estruturais de barras articuladas, caracterizadas pela não existência de momentos fletores nas extremidades nem ao longo do comprimento. Este facto faz com que este elemento, em cada nó, apenas apresente em coordenadas locais (x,y) o deslocamento u na direção da barra. Em coordenadas globais (X,Y) apresenta dois deslocamentos (U e V).

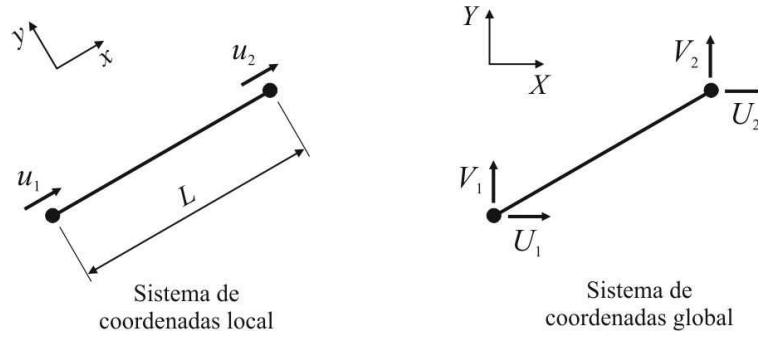


Figura 12 Sistema local vs. sistema global do elemento de barra 2D

Etapa 1 – O elemento apresenta a forma de barra e tem 1 nó em cada extremidade. Os vetores $\mathbf{a}_e$ e $\mathbf{f}_e$ tomam a forma

$$\mathbf{a}_e = \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}, \quad \mathbf{f}_e = \begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix}. \quad (9)$$

Etapa 2 – Obtenção do campo de deslocamentos em função de $\mathbf{N}$ e $\mathbf{a}_e$

$$u_e = \mathbf{N} \mathbf{a}_e. \quad (10)$$

Os polinómios a partir dos quais são obtidas as funções de interpolação para um elemento de uma dimensão, como é o caso dos elementos de barra e de viga, são do tipo

$$P(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3 \dots \quad (11)$$

O número de coeficientes do polinómio deve ser igual ao número de variáveis nodais disponíveis para efetuar a interpolação. No caso de elementos que apenas possuam deslocamentos como variáveis nodais, deve ser contabilizado um deslocamento por nó. Caso o elemento possua deslocamentos e rotações como variáveis nodais, então pode ser utilizado por cada nó um deslocamento e todas as rotações disponíveis [7]. No presente caso considera-se um deslocamento por nó (2 variáveis nodais)

$$P(x) = \alpha_1 + \alpha_2 x. \quad (12)$$

A expressão geral para o deslocamento u é igual a

$$u = \alpha_1 + \alpha_2 x. \quad (13)$$

De acordo com a expressão (13) definem-se os deslocamentos das extremidades da barra

$$\begin{aligned} u_1 &= u(x=0) = \alpha_1 \\ u_2 &= u(x=L) = \alpha_1 + \alpha_2 L, \end{aligned} \quad (14)$$

o que permite obter $\mathbf{a}_e$ em função da matriz $\mathbf{C}$ e do vetor $\boldsymbol{\alpha}$ dos coeficientes do polinómio de deslocamentos

$$\begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & L \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \end{Bmatrix} \quad \text{ou} \quad \mathbf{a}_e = \mathbf{C}\boldsymbol{\alpha}. \quad (15)$$

As funções de interpolação são obtidas por intermédio da expressão

$$\mathbf{N} = \mathbf{p}^T \mathbf{C}^{-1}, \quad \text{onde} \quad \mathbf{p}^T = \{1 \quad x\}. \quad (16)$$

Como tal

$$\mathbf{N} = [N_1 \quad N_2] = \{1 \quad x\} \begin{bmatrix} 1 & 0 \\ -1/L & 1/L \end{bmatrix} = \left[1 - \frac{x}{L} \quad \frac{x}{L} \right]. \quad (17)$$

Conforme se pode comprovar, as funções de interpolação satisfazem a condição geral de serem iguais à unidade no nó a que se referem, e serem nulas nos restantes nós. Recuperando a expressão (10)

$$u = \begin{bmatrix} 1 - \frac{x}{L} & \frac{x}{L} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix} = \left(1 - \frac{x}{L} \right) u_1 + \left(\frac{x}{L} \right) u_2. \quad (18)$$

A expressão (18) permite determinar o deslocamento u em qualquer ponto da barra (para $0 < x < L$) em função das funções de interpolação e dos deslocamentos das extremidades do elemento.

Etapa 3 – Definir o vetor de deformações $\boldsymbol{\varepsilon}$ em função da matriz de deformação $\mathbf{B}$ e de $\mathbf{a}_e$

$$\boldsymbol{\varepsilon} = \varepsilon_x = \frac{du}{dx} \quad (19)$$

$$\boldsymbol{\varepsilon} = \frac{d}{dx} \mathbf{N} \mathbf{a}_e = \frac{d}{dx} [N_1 \quad N_2] \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}. \quad (20)$$

$$\text{Como } \boldsymbol{\varepsilon} = \mathbf{B}\mathbf{a}_e \rightarrow \mathbf{B} = \frac{d}{dx} \begin{bmatrix} N_1 & N_2 \end{bmatrix} = \begin{bmatrix} -\frac{1}{L} & \frac{1}{L} \end{bmatrix}. \quad (21)$$

Etapa 4 – Definir o vetor de tensões $\boldsymbol{\sigma}$ em função do vetor de deformações $\boldsymbol{\varepsilon}$

$$\sigma_x = E \varepsilon_x \quad (22)$$

$$\text{Como } \boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} \rightarrow \mathbf{D} = E. \quad (23)$$

Etapa 5 – Definir a matriz de rigidez do elemento $\mathbf{k}_e$ através de

$$\mathbf{k}_e = \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dv. \quad (24)$$

No caso particular em questão [44]

$$\mathbf{k}_e = \int_A \int_0^L \mathbf{B}^T \mathbf{D} \mathbf{B} dx = AE \int_0^L \mathbf{B}^T \mathbf{B} dx \quad (25)$$

$$\mathbf{k}_e = AE \int_0^L \begin{bmatrix} 1/L^2 & -1/L^2 \\ -1/L^2 & 1/L^2 \end{bmatrix} dx \Leftrightarrow \mathbf{k}_e = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad (26)$$

em que $\mathbf{k}_e$ é dada por:

$$\frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = \begin{Bmatrix} f_i \\ f_j \end{Bmatrix} \quad \text{ou} \quad \mathbf{k}_e \mathbf{a}_e = \mathbf{f}_e \quad (27)$$

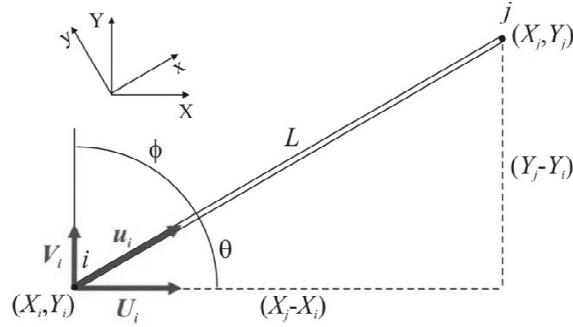
Transformação de coordenadas

A matriz de rigidez foi definida tendo em consideração o elemento no seu sistema de coordenadas locais (em que o eixo x apresenta a direção do eixo da barra). Para o elemento poder ser utilizado em qualquer orientação é introduzido o sistema de coordenadas global, válido para todos os elementos da estrutura independentemente da sua orientação [19].

Para o sistema de coordenadas local (Figura 12) tem-se $\mathbf{a}_e$ e para o sistema de coordenadas globais $\mathbf{A}_e$

$$\mathbf{a}_e = \{u_1 \quad u_2\}^T, \quad \mathbf{A}_e = \{U_1 \quad V_1 \quad U_2 \quad V_2\}^T. \quad (28)$$

A relação entre $\mathbf{a}_e$ e $\mathbf{A}_e$ é descrita com auxílio à figura seguinte. Observa-se que o deslocamento u_i é igual à soma das projeções de U_i e V_i no eixo x . Como tal



$$\begin{aligned} u_i &= U_i \cos \theta + V_i \sin \theta \\ u_j &= U_j \cos \theta + V_j \sin \theta. \end{aligned} \quad (29)$$

Figura 13 Projeção de deslocamentos

Definem-se $\cos \theta$, $\sin \theta$ e L como

$$l = \cos \theta = \frac{X_j - X_i}{L}, \quad m = \sin \theta = \frac{Y_j - Y_i}{L} \quad (30)$$

$$L = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2}. \quad (31)$$

Os valores de l e m são denominados de cossenos diretores, por representarem os cossenos dos ângulos que o eixo local x faz com os eixos globais X e Y . As expressões (29) podem ser escritas na forma matricial

$$\begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = \begin{bmatrix} l & m & 0 & 0 \\ 0 & 0 & l & m \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix} \quad \text{ou} \quad \mathbf{a}_e = \mathbf{T} \mathbf{A}_e, \quad (32)$$

em que a matriz de transformação, $\mathbf{T}$, é igual a

$$\mathbf{T} = \begin{bmatrix} l & m & 0 & 0 \\ 0 & 0 & l & m \end{bmatrix}. \quad (33)$$

A obtenção da matriz de rigidez do elemento em coordenadas globais, $\mathbf{K}_e$, é conseguida pela igualdade entre a energia de deformação de um elemento nas coordenadas locais, u_e , e globais, U_e [45]

$$u_e = \frac{1}{2} \mathbf{a}_e^T \mathbf{k}_e \mathbf{a}_e \quad (34)$$

$$U_e = \frac{1}{2} \mathbf{A}_e^T \mathbf{K}_e \mathbf{A}_e. \quad (35)$$

Substituindo a expressão (32) na (34) obtém-se

$$\mathbf{a}_e^T = (\mathbf{T} \mathbf{A}_e)^T = \mathbf{A}_e^T \mathbf{T}^T \quad (36)$$

$$u_e = \frac{1}{2} \mathbf{A}_e^T \mathbf{T}^T \mathbf{k}_e \mathbf{T} \mathbf{A}_e. \quad (37)$$

Como u_e e U_e devem ser idênticas, por igualdade das expressões (35) e (37) tem-se que

$$\mathbf{K}_e = \mathbf{T}^T \mathbf{k}_e \mathbf{T} \quad (38)$$

$$\mathbf{K}_e = \begin{bmatrix} l & 0 \\ m & 0 \\ 0 & l \\ 0 & m \end{bmatrix} \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} l & m & 0 & 0 \\ 0 & 0 & l & m \end{bmatrix} \quad (39)$$

$$\mathbf{K}_e = \frac{AE}{L} \begin{bmatrix} l^2 & lm & -l^2 & -lm \\ lm & m^2 & -lm & -m^2 \\ -l^2 & -lm & l^2 & lm \\ -lm & -m^2 & lm & m^2 \end{bmatrix}. \quad (40)$$

O sistema de equações para a determinação das reações nos apoios e deslocamentos é o apresentado em (8).

Estado de tensão nos elementos

As expressões para cálculo das forças e tensões neste tipo de elementos podem ser obtidas notando que o elemento de barra em coordenadas locais é um elemento sujeito apenas a esforço axial. Como tal, a tensão axial, σ , e força axial, F , são iguais a [2]

$$\sigma = E \varepsilon \quad (41)$$

$$\sigma = E \frac{u_j - u_i}{L} = \frac{E}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = \frac{E}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \mathbf{a}_e \quad (42)$$

$$\sigma = \frac{E}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \mathbf{TA}_e = \frac{E}{L} \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} l & m & 0 & 0 \\ 0 & 0 & l & m \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix} \quad (43)$$

$$\sigma = \frac{E}{L} \begin{bmatrix} -l & -m & l & m \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix}, \quad \text{logo} \quad F = \frac{AE}{L} \begin{bmatrix} -l & -m & l & m \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ U_j \\ V_j \end{Bmatrix}. \quad (44)$$

De notar que valores positivos indicam tração e negativos compressão para σ e F .

3.2 Elementos de barra 3D

O elemento barra 3D segue a mesma definição que o elemento anteriormente estudado, caracterizando-se pela ligação de rótulas sem ficção entre elementos. No entanto, com o incremento de mais uma dimensão este novo elemento terá 6 graus de liberdade nas coordenadas globais; três deslocamentos por cada nó, nas direções dos eixos coordenados.

3.2.1 Formulação linear

Estruturas reticuladas articuladas tridimensionais também podem ser modeladas utilizando elementos de barra de 2 nós. É para tal necessária a transformação de $\mathbf{k}_e$ (expressão (26)) para o sistema de coordenadas globais (X,Y,Z) , idêntico para a totalidade dos elementos da estrutura, para possibilitar a montagem de $\mathbf{K}$ [7]. Com esta mudança de coordenadas, os deslocamentos axiais locais, passam a ser definidos pelos deslocamentos ortogonais nas coordenadas globais como mostra a Figura 14.

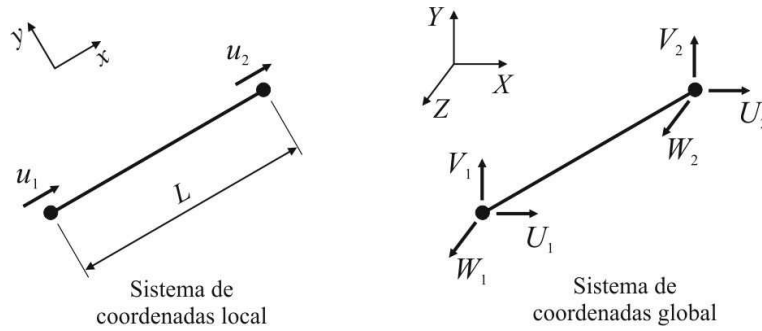


Figura 14 Sistema local vs. sistema global barra 3D

Transformação de coordenadas

Os vetores de variáveis nodais $\mathbf{a}_e$ e $\mathbf{A}_e$ podem ser escritos da seguinte forma

$$\mathbf{a}_e = \{u_1 \quad u_2\}^T, \quad \mathbf{A}_e = \{U_1 \quad V_1 \quad W_1 \quad U_2 \quad V_2 \quad W_2\}^T. \quad (45)$$

A Figura 15 representa um elemento de barra unidimensional ligada aos nós i e j de uma estrutura tridimensional, disposta de forma arbitrária no espaço de tal forma que a barra faz ângulos de θ_x , θ_y e θ_z com os eixos coordenados globais X , Y e Z , respectivamente. O versor com a direção da barra representada, λ_e , é dado por

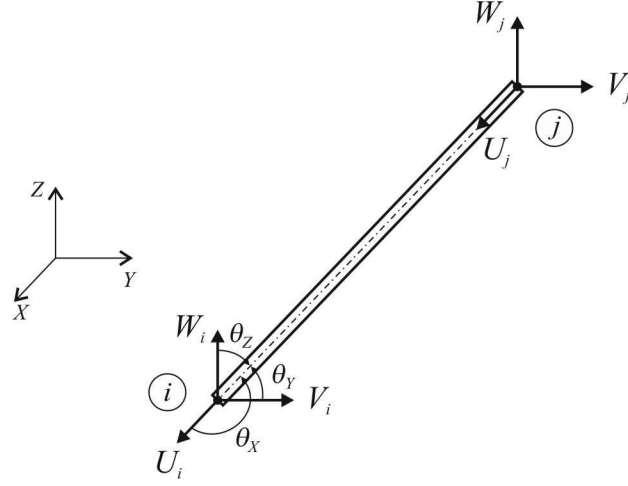


Figura 15 Elemento de barra 3D numa posição genérica no espaço, ilustrando o ângulo do elemento com os eixos globais

$$\lambda_e = \frac{1}{L} \left[(X_j - X_i) \mathbf{i} + (Y_j - Y_i) \mathbf{j} + (Z_j - Z_i) \mathbf{k} \right] \text{ ou} \quad (46)$$

$$\lambda_e = \cos \theta_x \mathbf{i} + \cos \theta_y \mathbf{j} + \cos \theta_z \mathbf{k} \text{ , onde} \quad (47)$$

$$l = \cos \theta_x = \frac{X_j - X_i}{L} \text{ , } m = \cos \theta_y = \frac{Y_j - Y_i}{L} \text{ , } n = \cos \theta_z = \frac{Z_j - Z_i}{L} \quad (48)$$

$$L = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2} . \quad (49)$$

Desta forma, os deslocamentos do elemento em coordenadas locais podem ser expressos em função das coordenadas globais da seguinte forma

$$\begin{aligned} u_i &= U_i \cos \theta_x + V_i \cos \theta_y + W_i \cos \theta_z \\ u_j &= U_j \cos \theta_x + V_j \cos \theta_y + W_j \cos \theta_z . \end{aligned} \quad (50)$$

Tal como no caso bidimensional, os cossenos diretores l , m e n representam os cossenos dos ângulos que o eixo local x faz com os eixos globais X , Y e Z . Escrevendo na forma matricial segundo (32) (i.e., $\mathbf{a}_e = \mathbf{T}\mathbf{A}_e$)

$$\begin{Bmatrix} u_i \\ u_j \end{Bmatrix} = \begin{bmatrix} l & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & n \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ W_i \\ U_j \\ V_j \\ W_j \end{Bmatrix} , \quad (51)$$

onde a matriz de transformação, $\mathbf{T}$, é dada por

$$\mathbf{T} = \begin{bmatrix} l & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & n \end{bmatrix}. \quad (52)$$

Pelo procedimento descrito no capítulo 3.1.1, considerando que a energia de deformação do elemento deve ser idêntica em coordenadas locais e globais, calcula-se a matriz de rigidez do elemento em coordenadas globais por (38) (i.e., $\mathbf{K}_e = \mathbf{T}^T \mathbf{k}_e \mathbf{T}$)

$$\mathbf{K}_e = \begin{bmatrix} l & 0 \\ m & 0 \\ n & 0 \\ 0 & l \\ 0 & m \\ 0 & n \end{bmatrix} \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} l & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & n \end{bmatrix} \quad (53)$$

$$\mathbf{K}_e = \frac{AE}{L} \begin{bmatrix} l^2 & lm & ln & -l^2 & -lm & -ln \\ & m^2 & mn & -lm & -m^2 & -mn \\ & & n^2 & -ln & -mn & -n^2 \\ & & & l^2 & lm & ln \\ & & & & m^2 & mn \\ \text{sim} & & & & & n^2 \end{bmatrix}. \quad (54)$$

O sistema de equações que permite a determinação das reações nos apoios é idêntico ao apresentado em (8).

Estado de tensão nos elementos

O procedimento de cálculo das forças e tensões neste tipo de elementos é idêntico ao apresentado no capítulo 3.1.1. Assim tem-se que

$$\sigma = \frac{E}{L} [-1 \quad 1] \mathbf{T} \mathbf{A}_e = \frac{E}{L} [-1 \quad 1] \begin{bmatrix} l & m & n & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & n \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ W_i \\ U_j \\ V_j \\ W_j \end{Bmatrix} \quad (55)$$

$$\sigma = \frac{E}{L} \begin{bmatrix} -l & -m & -n & l & m & n \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ W_i \\ U_j \\ V_j \\ W_j \end{Bmatrix}, \quad \text{logo} \quad F = \frac{AE}{L} \begin{bmatrix} -l & -m & -n & l & m & n \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ W_i \\ U_j \\ V_j \\ W_j \end{Bmatrix}. \quad (56)$$

3.3 Elementos de estrutura 2D

Também chamado de elemento viga 2D. Este elemento é caracterizado pela ligação rígida entre elementos. Quando sujeito a ações exteriores, estas devem ser sempre aplicadas nas suas extremidades. Este elemento está sujeito a esforços de tração/compressão, corte e momento fletor. No caso de uma análise 2D em coordenadas globais, este tipo de elemento possui 6 graus de liberdade; 1 deslocamento axial, 1 deslocamentos de corte e 1 rotação por cada um dos seus nós.

3.3.1 Formulação linear

A matriz de rigidez do elemento em coordenadas locais é obtida por sobreposição de $\mathbf{k}_e$ dos elementos de barra e de viga, por correspondência de graus de liberdade [2]

$$\mathbf{k}_e = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix}. \quad (57)$$

O sistema de equações do elemento (em coordenadas locais) é

$$\begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \begin{Bmatrix} u_1 \\ v_1 \\ \theta_1 \\ u_2 \\ v_2 \\ \theta_2 \end{Bmatrix} = \begin{Bmatrix} f_1 \\ F_1 \\ M_1 \\ f_2 \\ F_2 \\ M_2 \end{Bmatrix}. \quad (58)$$

Transformação de coordenadas

A expressão (57), relativa a $\mathbf{k}_e$, é referente ao sistema de coordenadas local do elemento (em que o eixo x tem a direção do seu eixo). Para utilização com uma orientação arbitrária no plano, é necessária a transformação de coordenadas [45, 46]. A figura seguinte compara as variáveis nodais em coordenadas locais (x,y) e globais (X,Y).

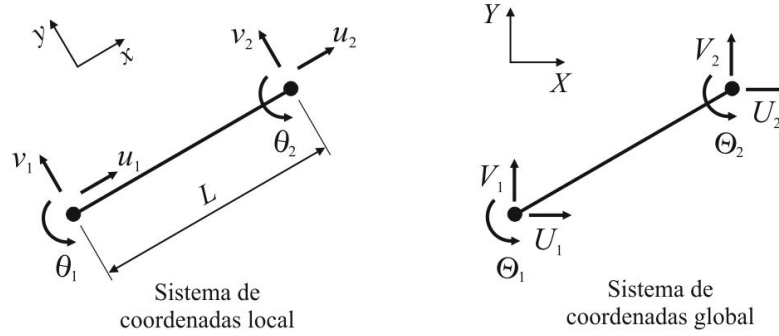
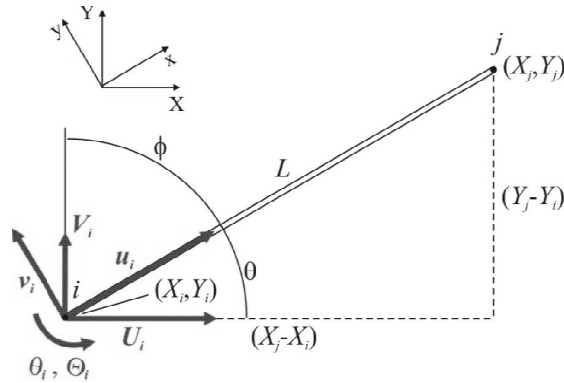


Figura 16 Sistema local vs. sistema global do elemento de viga 2D

Para o sistema de coordenadas local é definido $\mathbf{a}_e$ e para o sistema de coordenadas globais $\mathbf{A}_e$

$$\mathbf{a}_e = \{u_1 \quad v_1 \quad \theta_1 \quad u_2 \quad v_2 \quad \theta_2\}^T, \quad \mathbf{A}_e = \{U_1 \quad V_1 \quad \Theta_1 \quad U_2 \quad V_2 \quad \Theta_2\}^T. \quad (59)$$

A relação entre $\mathbf{a}_e$ e $\mathbf{A}_e$ é descrita com auxílio à figura seguinte.



$$\begin{aligned} u_i &= U_i \cos \theta + V_i \sin \theta & u_j &= U_j \cos \theta + V_j \sin \theta \\ v_i &= -U_i \sin \theta + V_i \cos \theta & v_j &= -U_j \sin \theta + V_j \cos \theta \\ \theta_i &= \Theta_i & \theta_j &= \Theta_j. \end{aligned} \quad (60)$$

Figura 17 Projeção dos deslocamentos para as coordenadas globais para um elemento de viga 2D

Os valores de $\cos \theta$, $\sin \theta$ e L são definidos como já descrito em (30) e (31). As expressões (60) podem ser escritas na forma matricial, lembrando que $l = \cos \theta$ e $m = \sin \theta$.

$$\begin{Bmatrix} u_i \\ v_i \\ \theta_i \\ u_j \\ v_j \\ \theta_j \end{Bmatrix} = \begin{bmatrix} l & m & 0 & 0 & 0 & 0 \\ -m & l & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & 0 \\ 0 & 0 & 0 & -m & l & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} U_i \\ V_i \\ \Theta_i \\ U_j \\ V_j \\ \Theta_j \end{Bmatrix} \quad \text{ou} \quad \mathbf{a}_e = \mathbf{T} \mathbf{A}_e. \quad (61)$$

A matriz de transformação, $\mathbf{T}$, é igual a

$$\mathbf{T} = \begin{bmatrix} l & m & 0 & 0 & 0 & 0 \\ -m & l & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & l & m & 0 \\ 0 & 0 & 0 & -m & l & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (62)$$

A obtenção de $\mathbf{k}_e$ é conseguida utilizando a expressão (38)

$$\mathbf{K}_e = \frac{E}{L} \begin{bmatrix} Al^2 + \frac{12I}{L^2} m^2 & \left(A - \frac{12I}{L^2}\right) lm & -\frac{6I}{L} m & -\left(Al^2 + \frac{12I}{L^2} m^2\right) & -\left(A - \frac{12I}{L^2}\right) lm & -\frac{6I}{L} m \\ \left(A - \frac{12I}{L^2}\right) lm & Am^2 + \frac{12I}{L^2} l^2 & \frac{6I}{L} l & -\left(A - \frac{12I}{L^2}\right) lm & -\left(Am^2 + \frac{12I}{L^2} l^2\right) & \frac{6I}{L} l \\ -\frac{6I}{L} m & \frac{6I}{L} l & 4I & \frac{6I}{L} m & -\frac{6I}{L} l & 2I \\ -\left(Al^2 + \frac{12I}{L^2} m^2\right) & -\left(A - \frac{12I}{L^2}\right) lm & \frac{6I}{L} m & Al^2 + \frac{12I}{L^2} m^2 & \left(A - \frac{12I}{L^2}\right) lm & \frac{6I}{L} m \\ -\left(A - \frac{12I}{L^2}\right) lm & -\left(Am^2 + \frac{12I}{L^2} l^2\right) & -\frac{6I}{L} l & \left(A - \frac{12I}{L^2}\right) lm & Am^2 + \frac{12I}{L^2} l^2 & -\frac{6I}{L} l \\ -\frac{6I}{L} m & \frac{6I}{L} l & 2I & \frac{6I}{L} m & -\frac{6I}{L} l & 4I \end{bmatrix}. \quad (63)$$

Carregamentos equivalentes

É comum em estruturas contínuas a aplicação de esforços distribuídos. Para resolução do sistema de equações estes esforços devem ser convertidos em cargas nodais estaticamente equivalentes (Figura 18), da mesma forma à descrita para os elementos de viga na referência [6], página 53 da Secção 3.4.2.

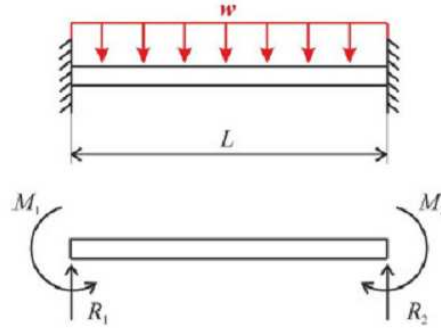


Figura 18 Carregamentos equivalentes [6]

Estado de tensão nos elementos

Para o cálculo dos esforços em cada elemento considera-se a equação de equilíbrio do elemento em causa

$$\mathbf{f}_e = \mathbf{k}_e \mathbf{a}_e \quad \text{ou} \quad \mathbf{F}_e = \mathbf{K}_e \mathbf{A}_e, \quad (64)$$

sendo necessárias as variáveis nodais em coordenadas do próprio elemento ($\mathbf{a}_e$) ou em coordenadas globais ($\mathbf{A}_e$), obtidas diretamente por aplicação da expressão geral (8) à totalidade da estrutura. Caso o elemento esteja sujeito a um esforço distribuído utiliza-se

$$\mathbf{f}_e = \mathbf{k}_e \mathbf{a}_e + \mathbf{r}_e \quad \text{ou} \quad \mathbf{F}_e = \mathbf{K}_e \mathbf{A}_e + \mathbf{R}_e, \quad (65)$$

onde $\mathbf{r}_e$ e $\mathbf{R}_e$ representam as reações ao esforço distribuído no elemento em coordenadas locais e globais, respetivamente. As expressões para cálculo da tensão axial devido ao momento fletor e esforço cortante nos elementos são dadas por

$$\sigma = \pm \frac{M}{I} y, \quad \tau = \frac{F}{A}. \quad (66)$$

Existe no entanto uma componente axial adicional devido ao esforço axial, dada por

$$\sigma_{axial} = \frac{f}{A}. \quad (67)$$

A tensão equivalente pode ser obtida pelo método de von Mises

$$\sigma_{eq} = \sqrt{(\sigma + \sigma_{axial})^2 + 3\tau^2}. \quad (68)$$

Refere-se ainda que o sinal de $(\sigma + \sigma_{\text{axial}})$ deve ser escolhido para que a soma das duas tensões axiais apresente o seu valor máximo em módulo, correspondente aos pontos críticos da secção da viga.

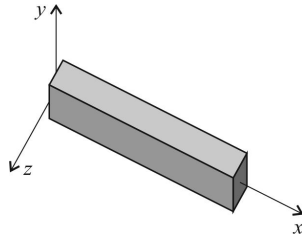


Figura 20 Eixos locais de um elemento de Viga 3D

Matriz de combinação do elemento – combinação de vários elementos

A componente de esforço axial pode ser introduzida na matriz de rigidez deste elemento de acordo com a expressão (26). Exemplificando novamente para as variáveis nodais da expressão (69)

$$\mathbf{k}_e = \frac{AE}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{matrix} u_1 \\ u_2 \end{matrix}. \quad (70)$$

Para flexão em torno do eixo z (plano de flexão xy) e em torno do eixo y (plano de flexão xz), as matrizes de rigidez do elemento em coordenadas locais são dadas por

$$\mathbf{k}_e = \frac{EI_z}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \begin{matrix} v_1 \\ \theta_{z1} \\ v_2 \\ \theta_{z2} \end{matrix}, \quad \mathbf{k}_e = \frac{EI_y}{L^3} \begin{bmatrix} 12 & -6L & -12 & -6L \\ -6L & 4L^2 & 6L & 2L^2 \\ -12 & 6L & 12 & 6L \\ -6L & 2L^2 & 6L & 4L^2 \end{bmatrix} \begin{matrix} w_1 \\ \theta_{y1} \\ w_2 \\ \theta_{y2} \end{matrix}. \quad (71)$$

As diferenças entre flexão em torno de z ou y decorrem do sentido definido como positivo para θ_i e θ_j , pela regra da mão direita no eixo respectivo [21]. Para o caso já deduzido de flexão em xy (esquerda), estes têm o sentido anti-horário. Para flexão em xz (direita), a mesma convenção estipula que o sentido de θ_i e θ_j é o oposto como apresentado pela Figura 21.

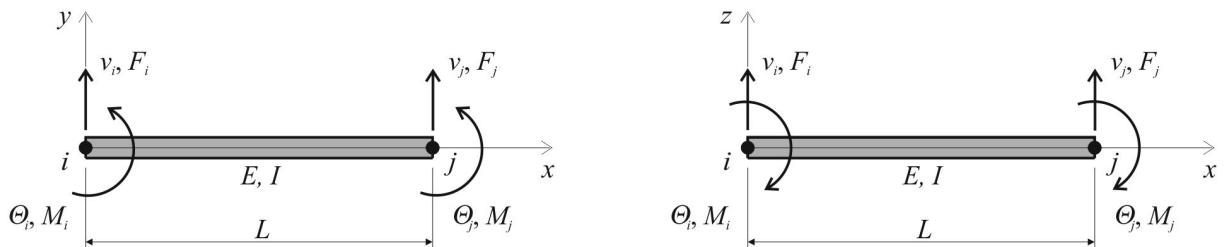


Figura 21 Convenções de Sinais para elemento de Viga 3D

A introdução da torção é apoiada na figura seguinte, que representa uma barra circular sujeita a momentos torsores nas suas extremidades. O elemento finito de torção correspondente, constituído

pelos nós 1 e 2, está igualmente representado. Este encontra-se orientado segundo o eixo x , e apresenta momentos nos nós com sentido definido pela regra da mão direita.

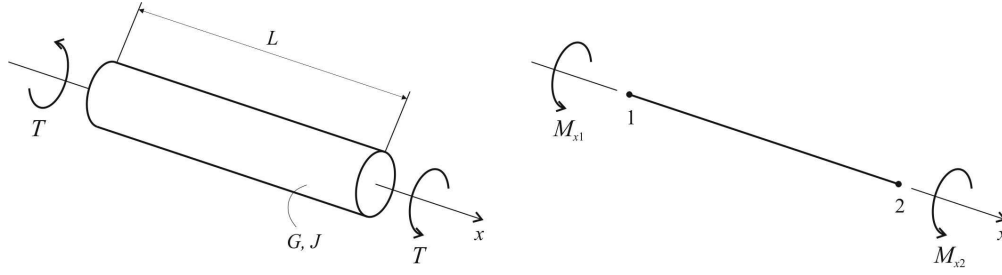


Figura 22 Momento torsor aplicado ao elemento

Da Resistência dos Materiais sabe-se que o ângulo de rotação (ϕ) de uma barra circular de secção constante sujeita ao momento torsor T é dado por

$$\phi = \frac{TL}{JG}, \quad (72)$$

onde L representa o comprimento do elemento, J o momento polar de inércia da secção reta e G o módulo de elasticidade ao corte do material. Aplicando as variáveis do elemento finito

$$\theta_{x2} - \theta_{x1} = \frac{TL}{JG} \Leftrightarrow T = \frac{JG}{L}(\theta_{x2} - \theta_{x1}). \quad (73)$$

Sabendo que o elemento se encontra em equilíbrio, pode-se escrever

$$M_{x1} = -T = -\frac{JG}{L}(\theta_{x2} - \theta_{x1}) \quad \text{e} \quad M_{x2} = T = \frac{JG}{L}(\theta_{x2} - \theta_{x1}), \quad (74)$$

ou em forma matricial

$$\frac{JG}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} \theta_{x1} \\ \theta_{x2} \end{Bmatrix} = \begin{Bmatrix} M_{x1} \\ M_{x2} \end{Bmatrix}. \quad (75)$$

Logo pode-se dizer que

$$\mathbf{k}_e = \frac{JG}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{matrix} \theta_{x1} \\ \theta_{x2} \end{matrix}. \quad (76)$$

Refere-se que o valor de J diz tipicamente respeito a secções circulares. Para secções distintas como secções quadradas, retangulares ou os mais diversos perfis, pode ser utilizado J_{eq} , obtido de tabelas disponíveis em literatura da Resistência dos Materiais [47]. A matriz de rigidez do elemento

em coordenadas locais é montada como se apresenta de seguida, por correspondência de graus de liberdade.

$$\mathbf{k}_e = \begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 & 0 & 0 & -\frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{JG}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{JG}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} \\ 0 & 0 & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & -\frac{JG}{L} & 0 & 0 & 0 & 0 & 0 & \frac{JG}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix} \begin{matrix} u_1 \\ v_1 \\ w_1 \\ \theta_{x1} \\ \theta_{y1} \\ \theta_{z1} \\ u_2 \\ v_2 \\ w_2 \\ \theta_{x2} \\ \theta_{y2} \\ \theta_{z2} \end{matrix} \quad (77)$$

A equação de equilíbrio do elemento (em coordenadas locais) é

$$\begin{bmatrix} \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 & 0 & 0 & -\frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{JG}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{JG}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ -\frac{AE}{L} & 0 & 0 & 0 & 0 & 0 & \frac{AE}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} \\ 0 & 0 & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & -\frac{JG}{L} & 0 & 0 & 0 & 0 & 0 & \frac{JG}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix} \begin{matrix} u_1 \\ v_1 \\ w_1 \\ \theta_{x1} \\ \theta_{y1} \\ \theta_{z1} \\ u_2 \\ v_2 \\ w_2 \\ \theta_{x2} \\ \theta_{y2} \\ \theta_{z2} \end{matrix} = \begin{matrix} f_1 \\ F_{y1} \\ F_{z1} \\ M_{x1} \\ M_{y1} \\ M_{z1} \\ f_2 \\ F_{y2} \\ F_{z2} \\ M_{x2} \\ M_{y2} \\ M_{z2} \end{matrix} \quad (78)$$

Transformação de coordenadas

É descrita a transformação de $\mathbf{k}_e$ para coordenadas globais [7]. A Figura 23 representa um elemento de estrutura unidimensional ligado aos nós i e j de uma estrutura tridimensional, disposto de forma arbitrária no espaço de tal forma que faz ângulos de θ_x , θ_y e θ_z com os eixos coordenados globais X , Y e Z , respetivamente.

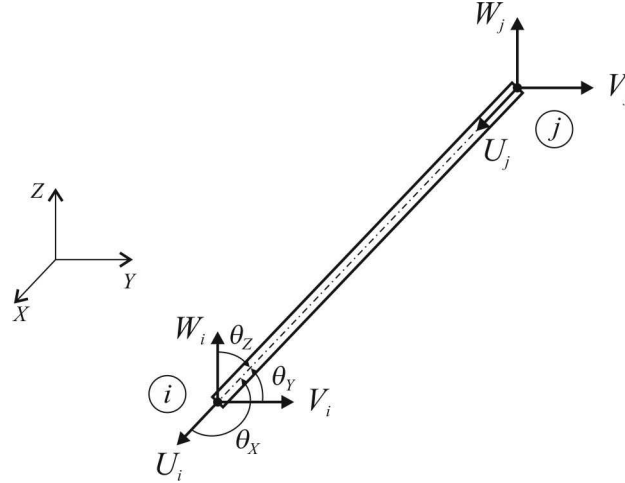


Figura 23 Disposição dos deslocamentos relativamente às coordenadas globais

É inicialmente feita a distinção entre os vetores de variáveis nodais locais $\mathbf{a}_e$ e globais $\mathbf{A}_e$

$$\begin{aligned} \mathbf{a}_e &= \{u_1 \quad v_1 \quad w_1 \quad \theta_{x1} \quad \theta_{y1} \quad \theta_{z1} \quad u_2 \quad v_2 \quad w_2 \quad \theta_{x2} \quad \theta_{y2} \quad \theta_{z2}\}^T \\ \mathbf{A}_e &= \{U_1 \quad V_1 \quad W_1 \quad \Theta_{x1} \quad \Theta_{y1} \quad \Theta_{z1} \quad U_2 \quad V_2 \quad W_2 \quad \Theta_{x2} \quad \Theta_{y2} \quad \Theta_{z2}\}^T. \end{aligned} \quad (79)$$

A transformação pode ser expressa da forma

$$\mathbf{a}_e = \mathbf{T} \mathbf{A}_e \quad \text{ou} \quad \mathbf{A}_e = \mathbf{T}^T \mathbf{a}_e. \quad (80)$$

A matriz de transformação, $\mathbf{T}$, de ordem 12×12 é definida pela matriz $\boldsymbol{\lambda}$ (3×3) dos cossenos diretores de cada eixo

$$\mathbf{T} = \begin{bmatrix} \lambda & & 0 \\ & \lambda & \\ & & \lambda \\ 0 & & \lambda \end{bmatrix}, \quad \text{onde} \quad \boldsymbol{\lambda} = \begin{bmatrix} l_1 & m_1 & n_1 \\ l_2 & m_2 & n_2 \\ l_3 & m_3 & n_3 \end{bmatrix}. \quad (81)$$

Na matriz $\boldsymbol{\lambda}$, os valores de l_1 , m_1 e n_1 são os cossenos dos ângulos que o eixo x local da viga faz com os eixos X , Y e Z globais, respetivamente. Da mesma forma, l_2 , m_2 e n_2 são os cossenos dos ângulos entre o eixo y local e os eixos coordenados do referencial global, e l_3 , m_3 e n_3 são os

cossenos dos ângulos entre o eixo z local e os mesmos eixos. Os cossenos diretores de cada eixo são definidos com recurso à figura que se segue, onde o ponto 3 está contido no plano xy .

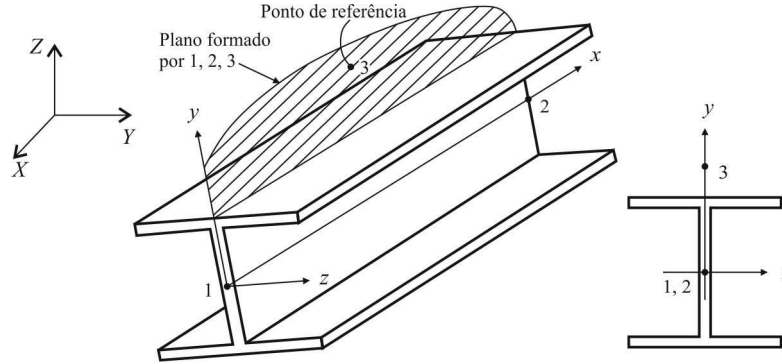


Figura 24 Plano de referência para determinar cossenos diretores

Sendo assim, seguindo um procedimento idêntico ao capítulo 3.1.1

$$l_1 = \frac{X_2 - X_1}{L}, \quad m_1 = \frac{Y_2 - Y_1}{L}, \quad n_1 = \frac{Z_2 - Z_1}{L} \quad (82)$$

$$L = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}. \quad (83)$$

$\mathbf{V}_x$ é assim um vetor unitário com o sentido positivo do eixo x local

$$\mathbf{V}_x = [l_1 \quad m_1 \quad n_1]^T. \quad (84)$$

O vetor unitário desde o ponto 1 ao ponto 3 da viga é dado por (L_{13} é a distância entre os pontos 1 e 3)

$$\mathbf{V}_{13} = \left[\frac{X_3 - X_1}{L_{13}} \quad \frac{Y_3 - Y_1}{L_{13}} \quad \frac{Z_3 - Z_1}{L_{13}} \right]. \quad (85)$$

O vetor unitário segundo o eixo z local ($\mathbf{V}_z$) e contendo os cossenos diretores do eixo z é dado por

$$\mathbf{V}_z = [l_3 \quad m_3 \quad n_3]^T = \frac{\mathbf{V}_x \times \mathbf{V}_{13}}{|\mathbf{V}_x \times \mathbf{V}_{13}|}, \quad (86)$$

onde o produto vetorial entre quaisquer vetores $\mathbf{u}$ e $\mathbf{v}$ é dado por

$$\mathbf{u} \times \mathbf{v} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ u_x & u_y & u_z \\ v_x & v_y & v_z \end{vmatrix} = \begin{bmatrix} u_y v_z - v_y u_z \\ v_x u_z - u_x v_z \\ u_x v_y - v_x u_y \end{bmatrix}. \quad (87)$$

Finalmente, os cossenos diretores do eixo y local são dados por

$$\mathbf{V}_y = [l_2 \quad m_2 \quad n_2]^T = \mathbf{V}_z \times \mathbf{V}_x. \quad (88)$$

O cálculo de $\mathbf{K}_e$ é obtido pela expressão (38). Para a resolução do sistema de equações (8) deve-se introduzir as reações e esforços aplicados à estrutura no sistema de eixos global

$$\begin{aligned} \mathbf{R} + \mathbf{F} = & \{R_{x1} \quad R_{y1} \quad R_{z1} \quad MR_{x1} \quad MR_{y1} \quad MR_{z1} \quad R_{x2} \quad R_{y2} \quad R_{z2} \quad MR_{x2} \quad MR_{y2} \quad MR_{z2}\}^T + \\ & + \{F_{x1} \quad F_{y1} \quad F_{z1} \quad M_{x1} \quad M_{y1} \quad M_{z1} \quad F_{x2} \quad F_{y2} \quad F_{z2} \quad M_{x2} \quad M_{y2} \quad M_{z2}\}^T. \end{aligned} \quad (89)$$

Carregamentos equivalentes

Para efeitos de conversão de carregamentos distribuídos em carregamentos nodais equivalentes, deve ser aplicado o procedimento descrito para os elementos de viga na referência [6], pagina 53 da Secção 3.4.2, com particular atenção para a aplicação dos valores nodais obtidos em $\mathbf{F}$ de acordo com o sistema de eixos global definido para a estrutura.

Estado de tensão nos elementos

O procedimento a seguir é semelhante ao descrito para o elemento de estrutura bidimensional (capítulo 3.1.1), por utilização das expressões (64) e (65). Para cálculo do estado de tensão, são utilizadas as expressões de flexão, corte (66) e esforço axiais (67) definidos anteriormente, com consideração de σ devido a flexão segundo 2 eixos, τ igualmente em 2 direcções ortogonais, e o corte devido à torção, $\tau_{\text{torção}}$, dado por

$$\tau_{\text{torção}} = \frac{T y}{J}, \quad (90)$$

onde y é a distância desde o centroide da secção e o ponto em análise. Para o cálculo de σ_{eq} , deve ser definido o ponto crítico da secção pela análise da distribuição de cada componente de tensão na mesma (caso seja necessário). Caso não seja prática a utilização da expressão (68) para cálculo de σ_{eq} devido à complexidade do estado de tensão, é recomendada a determinação das tensões principais σ_1 , σ_2 e σ_3 por obtenção dos valores próprios do tensor das tensões do ponto em análise (para mais detalhes ver [47]) e aplicação de expressão equivalente a (68)

$$\sigma_{eq} = \frac{1}{\sqrt{2}} \sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}. \quad (91)$$

4 Desenvolvimento do *software*

Neste capítulo será abordada a metodologia implementada para o desenvolvimento do programa. Consta neste capítulo os diagramas de programação, variáveis utilizadas, exemplo de utilização e comportamento do programa, assim como a validação dos casos de estudo usando como recurso exercícios previamente validados.

4.1 Estrutura

De maneira a abranger todos os casos de análise para a formulação de elementos finitos unidimensionais enumerados no capítulo anterior, considerou-se inicialmente para o programa desenvolvido a seguinte estrutura programática:



Figura 25 Diagrama base do programa

A partir deste diagrama base, conseguiu-se definir a estrutura final, considerando os seguintes tópicos:

- Janela de seleção de idioma;
- Após a seleção do caso de análise, lançar uma janela de importação de estruturas existentes ou de criação de nova estrutura;
- Após a seleção do tipo de análise, criar variável/memória com indicação do caso;
- Durante a criação dos elementos, é criada a matriz das coordenadas dos nós da estrutura e a matriz com a informação dos nós por elemento;
- Possibilidade de pré-visualização da estrutura a cada passo da criação;
- Possibilidade de introduzir propriedades diferentes para cada elemento (guardados em forma vetorial);
- Ainda durante a criação dos elementos, colocar campos para a introdução de propriedades por elemento, permitindo a introdução das unidades das propriedades em diferentes ordens de grandeza;
- Lembrar nos elementos subsequentes o valor e a ordem de grandeza das propriedades do elemento anterior, de maneira a não ser necessária a reintrodução de valores e grandezas no caso de serem iguais ao elemento anterior;
- Durante a introdução das ações e restrições, existir a possibilidade de escolher o nó a modificar;
- Visualizar, na janela de pré-visualização, a indicação das ações e restrições colocadas;
- Criação de vetor de ações e restrições;
- Na janela de resultados, visualização de estrutura inicial, deformada, distribuição de tensões axiais sobre cada elemento e mostrar os resultados da análise em forma tabular;
- Possibilidade de exportar resultados para ficheiros externos;
- Possibilidade de editar cargas, restrições e propriedades da estrutura em análise;
- Possibilidade de guardar a estrutura criada;
- Possibilidade de importação de estruturas previamente criadas.

Com estes pontos criou-se o programa, seguindo o esquema programático mostrado na Figura 26.

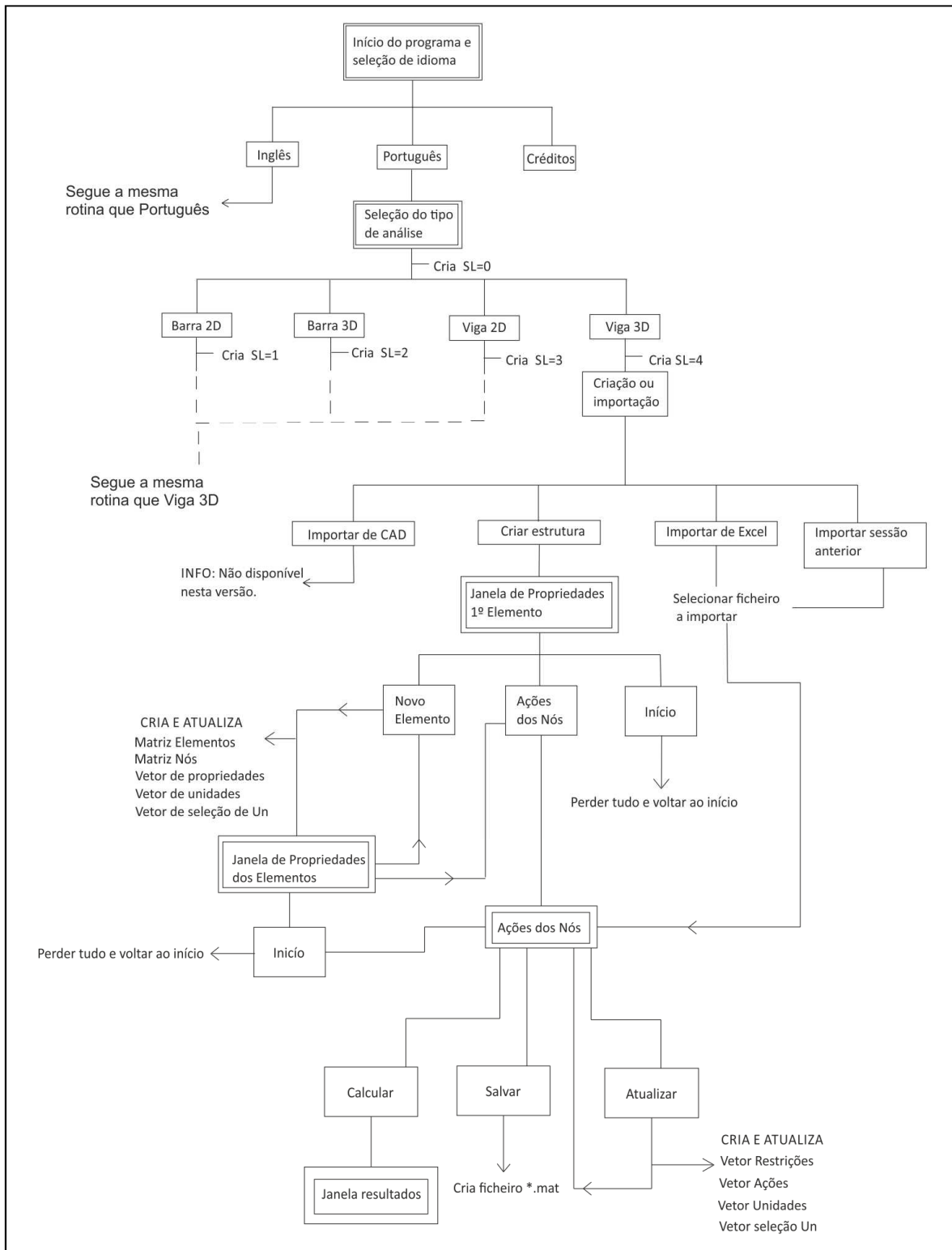


Figura 26 Diagrama programático expandido - parte 1 de 2

Verifique-se que no diagrama da Figura 26 só foi desenvolvido o ramo relativo ao caso de estudo para o elemento de viga 3D. Os passos programáticos são semelhantes para os outros tipos de análise.

Inicialmente o utilizador terá a opção de seleção de idioma a utilizar. Após realizada a seleção, surgirá a janela de seleção do tipo de caso a analisar. Após selecionar o caso de estudo, surge a janela para selecionar quer a criação de estrutura ou a importação de estruturas previamente criadas. Nesta fase, o programa apaga qualquer variável existente na memória. Caso o utilizador pretenda importar um caso previamente estudado, o programa tem capacidades para verificar o tipo de estudo adotado durante a análise do mesmo. Caso o utilizador pretenda criar a estrutura, surgirá a janela de formulação do primeiro elemento. Após introduzidas as variáveis para o primeiro elemento, o utilizador deverá selecionar o botão criar elemento. O programa grava as variáveis atuais e é mostrada a janela para a criação dos elementos subsequentes. Aqui o processo é cíclico, sendo que a cada novo elemento criado, o programa grava e atualiza as novas variáveis. Finalizado o processo de criação de elementos, o utilizador deverá selecionar o botão de ações nos nós, sendo mostrada a janela para a introdução das condições fronteira e carregamentos. Sobre esta janela, o utilizador deverá selecionar o nó no qual pretenda inserir as condições de estudo, selecionando o botão *update* de maneira ao programa guardar e atualizar a estrutura na pré-visualização. Finalizados estes passo, o utilizador poderá guardar a estrutura atual, selecionando o botão salvar, ou passar à fase de resultados selecionando o botão calcular.

A partir da janela de resultados o fluxo de acontecimentos segue o diagrama programático mostrado na Figura 27. Na janela de resultados são exibidos por defeito os resultados das reações e é pré-visualizada a estrutura com as condições de carregamento para a análise. Esta janela de pré-visualização pode ser alterada com os botões existentes por baixo da mesma, de maneira a ser mostrada a estrutura deformada ou a estrutura com a distribuição das tensões. Sobre o painel de resultados (tabela), existem 3 botões com indicação dos resultados disponíveis. Selecionando qualquer um destes botões, serão mostrados os resultados correspondentes à seleção. Na parte inferior da tabela de resultados existem os botões que permitem editar a estrutura e exportar a estrutura para o ficheiro *.xls. O primeiro botão abre a janela de edição de propriedades dos elementos e de ações nos nós. Esta janela permite editar até certo ponto, os dados relativos à estrutura. Esta edição segue o mesmo princípio utilizado nas janelas de criação de elementos e de atribuição de condições fronteira e cargas sobre os nós. Também esta disponível nesta janela, a opção de guardar a estrutura (em formato *.mat) e recalcular a estrutura com as novas condições. Caso o utilizador selecione este ultimo botão, será mostrada novamente a janela resultados.

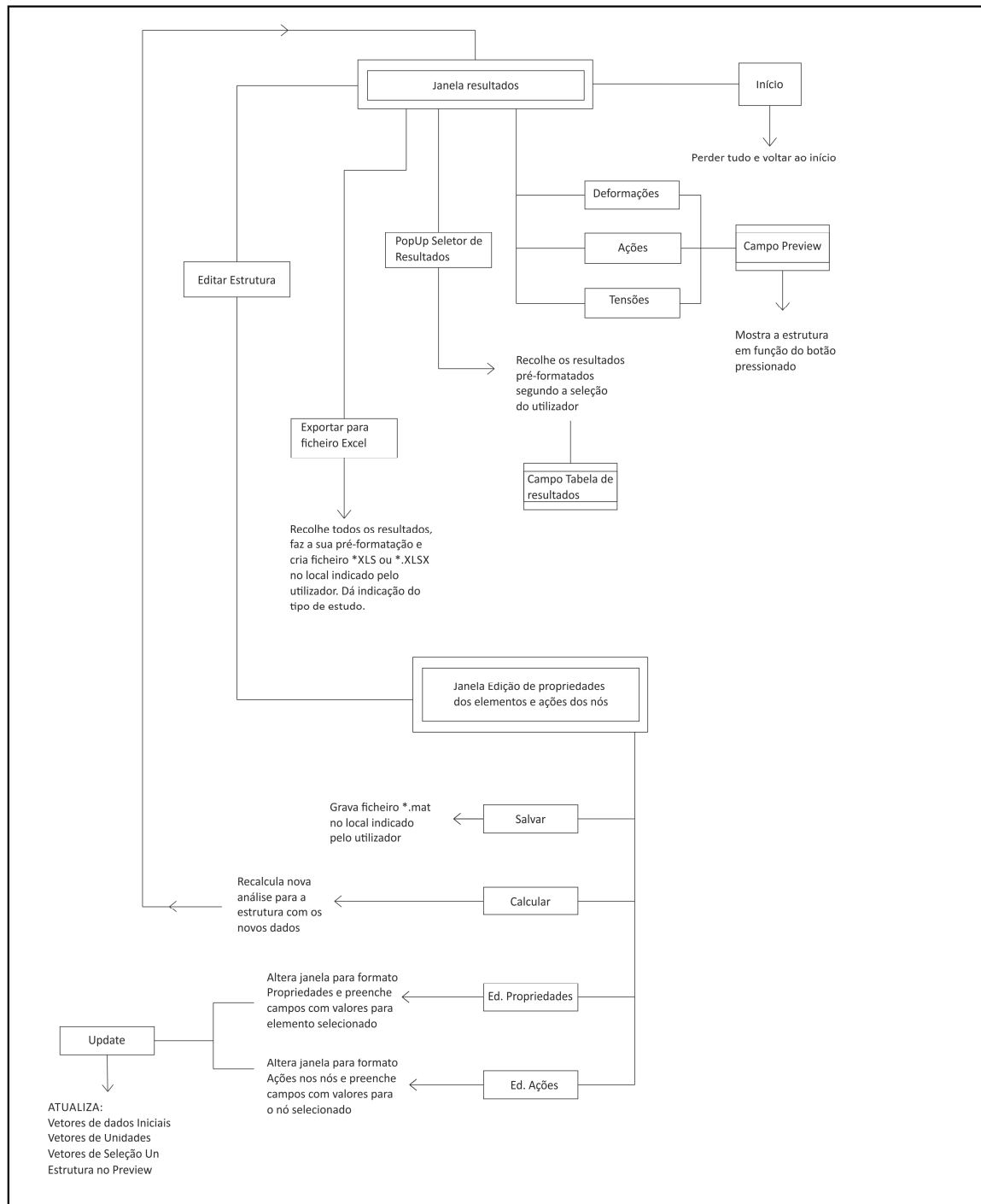


Figura 27 Diagrama programático expandido - parte 2 de 2

Foram ainda consideradas para o programa as variáveis presentes nas Tabelas: Tabela 2, Tabela 3 e Tabela 4. Note-se que existem casos de análise que não utilizam a totalidade das variáveis enunciadas.

Tabela 2 Variáveis utilizadas na programação - parte 1 de 3

<u>Nome</u>	<u>Variável</u>	<u>Formato</u>
Tipo de Estudo	SL	Escalar
Coordenadas dos nós	NodeCoordinates	Matriz
Contador de nós	NodeC	Escalar
Variável para preencher campos <i>popup menu</i> de seleção de nó	Nodes	Vetor
Contador de elementos	ElemC	Escalar
Ligações dos elementos	ElemNodes	Matriz
Secção do elemento	A	Vetor
Memória de unidade selecionada para a secção	A_sel	Vetor
Grandeza de unidade selecionada para a secção	A_unit	Vetor
Módulo de Young	E	Vetor
Memória de unidade selecionada para módulo de Young	E_sel	Vetor
Grandeza de unidade selecionada para módulo de Young	E_unit	Vetor
Módulo de elasticidade transversal	G	Vetor
Memória de unidade selecionada para módulo de elasticidade transversal	G_sel	Vetor
Grandeza de unidade selecionada para módulo de elasticidade transversal	G_unit	Vetor
Momento de inércia à flexão em y	Iy	Vetor
Memória de unidade selecionada para momento de inércia à flexão em y	Iy_sel	Vetor
Grandeza de unidade selecionada para momento de inércia à flexão em y	Iy_unit	Vetor
Momento de inércia à flexão em z	Iz	Vetor
Memória de unidade selecionada para momento de inércia à flexão em z	Iz_sel	Vetor
Grandeza de unidade selecionada para momento de inércia à flexão em z	Iz_unit	Vetor
Momento polar de inércia	J	Vetor
Memória de unidade selecionada para momento polar de inércia	J_sel	Vetor
Grandeza de unidade selecionada para momento polar de inércia	J_unit	Vetor
Ponto 3 no plano de suporte por elemento	P3	Matriz
Memória de aviso nos resultados	W_sem	Escalar
Componente raio de viga (aproximação perfil circular)	spr	Vetor

Tabela 3 Variáveis utilizadas na programação - parte 2 de 3

<u>Nome</u>	<u>Variável</u>	<u>Formato</u>
Componente raio de viga (aproximação perfil circular) seleção de unidade	r_sel	Vetor
Componente raio de viga (aproximação perfil circular) grandeza de unidade	r_unit	Vetor
Componente y de viga (aproximação perfil circular)	spy1	Vetor
Componente y de viga (aproximação perfil circular) seleção de unidade	y1_sel	Vetor
Componente y de viga (aproximação perfil circular) grandeza de unidade	y1_unit	Vetor
Componente z de viga (aproximação perfil circular)	spy2	Vetor
Componente z de viga (aproximação perfil circular) seleção de unidade	y2_sel	Vetor
Componente z de viga (aproximação perfil circular) grandeza de unidade	y2_unit	Vetor
Variável para preencher campos <i>popup menu</i> de seleção de elemento	Element	Vetor
Esforços iniciais por nó	Fo	Vetor
Restrições iniciais por nó	Rt	Vetor
Resultados das reações por nó	Reac	Vetor
Resultados das reações para tabela	ReacDat	Célula
Resultados das componentes das tensões aplicáveis por elemento	Sigma	Matriz
Resultados em forma tabular para tensões axiais	SigmaDat1	Célula
Resultados em forma tabular para tensões corte yy	SigmaDat2	Célula
Resultados em forma tabular para tensões corte zz	SigmaDat3	Célula
Resultados em forma tabular para tensões por torção em xx	SigmaDat4	Célula
Resultados em forma tabular para tensões por flexão em yy	SigmaDat5	Célula
Resultados em forma tabular para tensões por flexão em zz	SigmaDat6	Célula
Deslocamentos por nó	Desl	Vetor
Tabela resultados de deslocamentos	DisplDat	Célula
Resultados das componentes das ações resultantes aplicáveis por elemento	Fl_Elem	Matriz

Tabela 4 Variáveis utilizadas na programação - parte 3 de 3

<u>Nome</u>	<u>Variável</u>	<u>Formato</u>
Esforços axiais por elemento para tabela	Floc1	Célula
Esforços de corte yy por elemento para tabela	Floc2	Célula
Esforços de corte zz por elemento para tabela	Floc3	Célula
Momento em torno xx por elemento para tabela	Floc4	Célula
Momento em torno yy por elemento para tabela	Floc5	Célula
Momento em torno zz por elemento para tabela	Floc6	Célula

4.2 Métodos de programação

O programa foi desenvolvido utilizando a linguagem programática do Matlab®. Esta linguagem baseia-se na linguagem Fortran, Basic e C+, no entanto tem uma semântica própria para muitas das rotinas que foram implementadas.

Cada janela apresentada no programa resulta da criação de dois ficheiros com o mesmo nome, mas de tipos diferentes:

- Nos ficheiros *.fig são guardadas todas as informações relativas à constituição da janela. Aparência, painéis e ferramentas de interface (botões, *popup menus*, etc.) e as suas propriedades são elementos que apenas podem ser modificados utilizando a ferramenta de criação e edição do Matlab® GUI Builder;
- Nos ficheiros *.m são guardados todas as instruções e rotinas a serem executadas pela janela. A cada componente de interface corresponde um campo próprio dentro destes ficheiros. Estes tipos de ficheiros podem ser abertos e editados no Matlab®, no editor de funções, ou no *Notepad* do Windows®. Os ficheiros *.m, quando editados no Matlab®, mostram no seu conteúdo o código, diferenciando por cores: funções, rotinas, atributos, comentários, etc.

Optou-se por implementar os códigos programáticos de cálculo diretamente nos ficheiros *.m das janelas, evitando assim a criação de funções externas.

4.3 Bases de dados

O programa desenvolvido não possui qualquer tipo de base de dados antes de ser iniciado. Após iniciado, o programa irá criar o conjunto de variáveis necessárias, constituindo assim a base de dados para o problema em análise.

Existe ainda a possibilidade de importar um modelo previamente analisado, ao importar uma base de dados previamente criada. Esta base de dados, dependendo do estado do programa aquando da sua criação, poderá conter para além dos dados da criação da estrutura, dados relativos aos resultados.

Durante a fase de cálculo do programa, são criadas variáveis de transporte que existem só para auxiliar o processo de cálculo. Exemplos deste tipo de variáveis são a matriz de rigidez local e a matriz transformada para cada elemento. Estas matrizes são criadas durante a rotina para a criação da matriz global, e após isto a informação utilizada perder-se-á (ver Tabela 5).

Tabela 5 Exemplo de gestão de variáveis e montagem da matriz de rigidez global

```
NodeC = evalin('base', 'NodeC');
ElemC = evalin('base', 'ElemC');
NodeCoordinates = evalin('base', 'NodeCoordinates');
xx=NodeCoordinates(:,1);
yy=NodeCoordinates(:,2);
zz=NodeCoordinates(:,3);
ElemNodes = evalin('base', 'ElemNodes');
E = (evalin('base', 'E'));
A = (evalin('base', 'A'));
%CRIAÇÃO MATRIZ LOCAL Ke E GERAL KK
KK=zeros(3*NodeC);
for i=1:ElemC
    V=ElemNodes(:,1);
    W=ElemNodes(:,2);
    x1=xx(V(i));
    x2=xx(W(i));
    y1=yy(V(i));
    y2=yy(W(i));
    z1=zz(V(i));
    z2=zz(W(i));
    L(i)=sqrt((x2-x1)^2+(y2-y1)^2+(z2-z1)^2);
    %Matriz local Ke
    l=(x2-x1)/L(i);
    m=(y2-y1)/L(i);
    n=(z2-z1)/L(i);
    tt=[l^2 l*m l*n;l*m m^2 m*n;l*n m*n n^2];
    Ke=((E(i)*A(i))/L(i))*[tt -tt;-tt tt];
    %Montagem da Matriz
    elemDOF=[3*V(i)-2 3*V(i)-1 3*V(i) 3*W(i)-2 3*W(i)-1 3*W(i)];
    KK(elemDOF,elemDOF)=KK(elemDOF,elemDOF)+Ke;
end
```

4.4 Funcionamento do *software*

Na presente secção serão mostrados os passos a seguir para o correto funcionamento do programa. Nos seguintes tópicos, serão ilustradas as janelas do programa a cada passo de uma análise típica. Sempre que haja a necessidade, serão mostradas as partes mais importantes do código utilizado.

Deve-se referir que o programa não é do tipo *stand alone*, ou seja, deve ser inicializado desde a plataforma programática Matlab®.

4.4.1 Arranque do programa

Após abertura do Matlab®, deve-se alterar a diretoria de trabalho para o local onde se encontra o programa desenvolvido. Utiliza-se para esse efeito o botão assinalado a vermelho na Figura 28.



Figura 28 Mudança de diretoria de trabalho

Com a diretoria definida, arranca-se o programa com duplo clique sobre o ficheiro “A_Start.fig” ou digitando-se sobre a linha de comando: “A_Start”. Estes procedimentos irão inicializar a janela mostrada na Figura 29.



Figura 29 Início do programa

Sobre esta janela o utilizador deverá escolher o idioma pretendido, podendo escolher entre a língua Portuguesa e a língua Inglesa.

4.4.2 Seleção do tipo de estrutura a analisar

Após selecionado o idioma pretendido, será mostrada a janela de seleção do tipo de estrutura a analisar, representada na Figura 30. A partir deste ponto é apresentado o programa em língua Portuguesa.



Figura 30 Seleção do tipo de análise

O utilizador tem a possibilidade de voltar à janela mostrada na Figura 29, caso pretenda mudar o idioma, utilizando o botão voltar.

No botão créditos será exibida a janela com informação do programa e do autor da presente dissertação.



Figura 31 Método de entrada de dados

O utilizador dispõe de 4 botões de seleção de casos para análise, no entanto, optou-se por ilustrar os procedimentos para o caso de estudo mais abrangente, o de viga 3D. Após a seleção do caso de estudo, será inicializada a janela de escolha do método de entrada, onde o utilizador terá a opção de criar ou importar estrutura, como mostrado na Figura 31.

Se for selecionada a opção de importar estrutura será mostrada ao utilizador a janela de tipo de importação como na Figura 32.



Figura 32 Método de importação

No caso de ser selecionado o botão “Ficheiro CAD”, irá aparecer um aviso a informar que tal opção não está disponível para esta versão do programa (Figura 33). Esta situação foi abordada na secção 1.3 do capítulo 1.

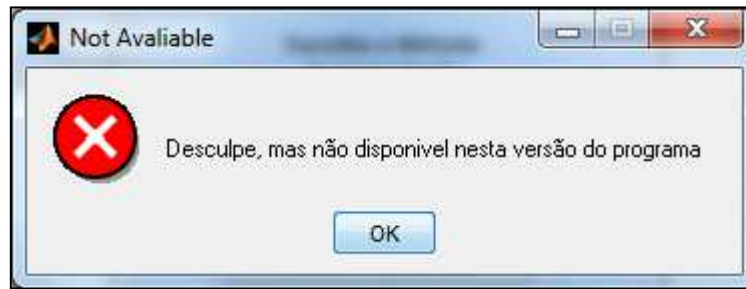


Figura 33 Aviso para importação de CAD

Para o caso de ser selecionada qualquer uma das outras opções de importação, o utilizador terá que indicar qual o ficheiro a importar. O programa mostrará uma janela de abertura de ficheiro, sendo que para o ficheiro tipo colha de cálculo a janela mostrada será a da Figura 34.

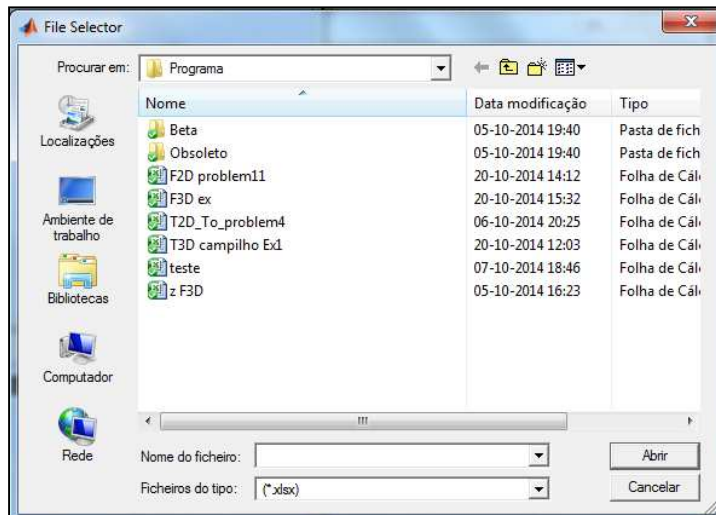


Figura 34 Abertura de ficheiro tipo folha de cálculo

Para a abertura de ficheiro de sessões anteriores (formato *.mat), o utilizador deverá seleccionar o tipo de ficheiro a abrir como indicado na Figura 35.

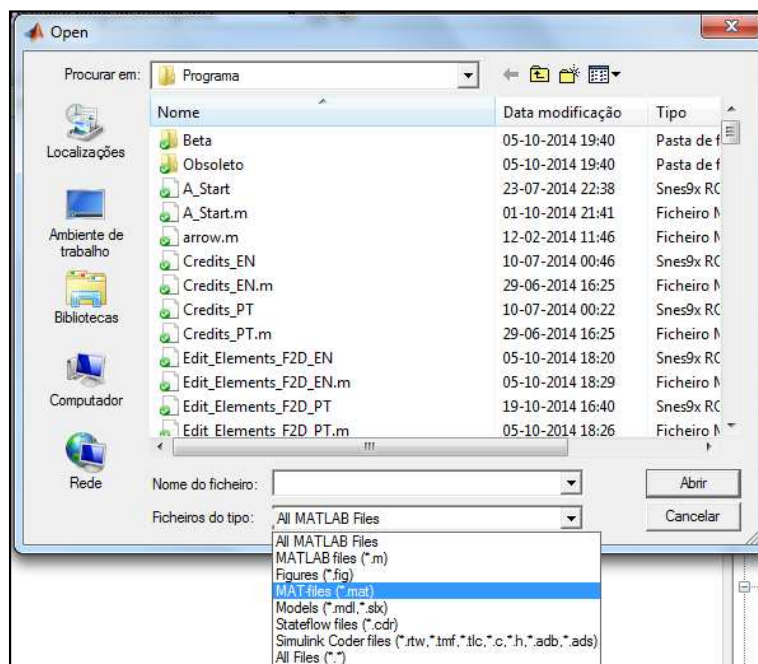


Figura 35 Abertura de ficheiro de sessão anterior

Após importação será lançada a janela das ações dos nós com a estrutura importada. A explicação da dita janela será abordada na secção 4.4.4 do presente capítulo.

De forma a continuar com o desenvolvimento do programa, admitir-se-á que foi utilizada a opção “Criar Estrutura” da janela mostrada na Figura 31, e o utilizador será levado até a janela das propriedades do primeiro elemento.

4.4.3 Propriedades dos elementos e geometria

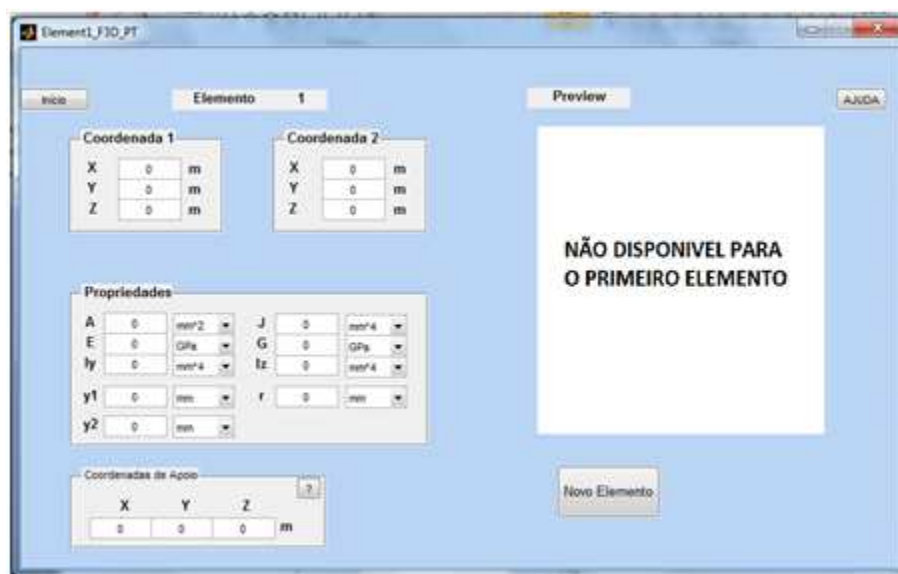


Figura 36 Propriedades do 1º elemento

Na janela da Figura 36 nenhum elemento foi criado. No topo da janela são apresentados os campos para inserir as coordenadas do nó inicial e do nó final para a constituição do primeiro elemento. No campo central estão dispostos os campos de preenchimento das propriedades do elemento a ser criado. O utilizador tem a opção de inserir as unidades escolhendo a partir das grandezas disponíveis nas *popup menus*. Apresentam-se como possibilidade de escolha as unidades mais comumente utilizadas em engenharia para cada uma das variáveis, nomeadamente m^2 , cm^2 e mm^2 para o caso da área da secção do elemento.

Os campos na parte inferior esquerda da janela são referentes à coordenada de apoio para permitir o correto cálculo da matriz λ , essencial para a mudança de coordenadas, como referido na secção 3.4.1 e ilustrado na Figura 24. Esta opção só estará disponível para análises de elementos de viga 3D, pois só neste caso de análise a orientação da secção do perfil em análise influencia o comportamento da estrutura. Esta informação em conjunto com as distâncias $y1$ e $y2$ relativos à geometria do perfil e os momentos de inércia Iy e Iz , permitem também o cálculo das tensões devido ao momento torsores em y e z nos nós do elemento (calculados pelo programa pela aproximação a uma secção circular).

Finalizado o preenchimento das propriedades e coordenadas, o utilizador deverá seleccionar o botão “Novo Elemento”. Assim, os dados inseridos são guardados e é mostrada a janela para os restantes elementos, como mostra a Figura 37, onde já se encontra disponível o elemento previamente criado.

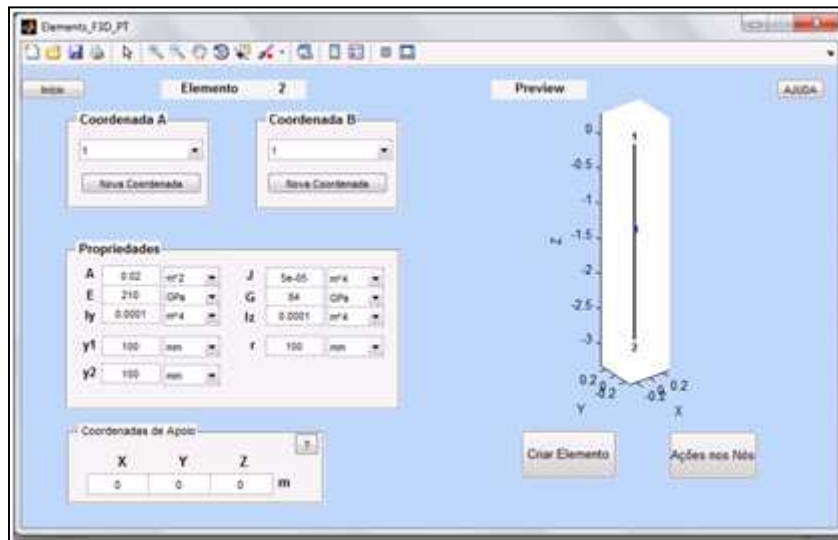


Figura 37 Propriedades dos elementos

Encontra-se agora disponível o botão “Ações nos nós” para o caso de se pretender realizar o estudo apenas com um elemento.

De igual forma encontra-se disponível no topo da janela o comando para manipulação da estrutura disponível no campo de *preview*.

Os campos das propriedades encontram-se preenchidos com as propriedades do último elemento criado (para este exemplo, o primeiro elemento), de forma a agilizar a inserção de dados. Desta forma o utilizador apenas terá que inserir novas propriedades para os elementos cujas propriedades sejam diferentes daquelas apresentadas.



Figura 38 Nova coordenada

Uma vez que os novos elementos podem partilhar nós anteriormente criados, o campo de introdução de coordenadas foi alterado. Caso se pretenda escolher um nó já existente, este deve ser

selecionado a partir da lista de nós criados. Caso contrário, o utilizador terá que criar o novo nó utilizando o botão “Nova Coordenada”. Esta ação irá lançar a janela mostrada na Figura 38.

Após inserir a nova coordenada, o *popup menu* dos nós (no topo da janela da Figura 37) e o *preview* da estrutura são atualizados, mostrando o novo ponto criado, como mostra a Figura 39.

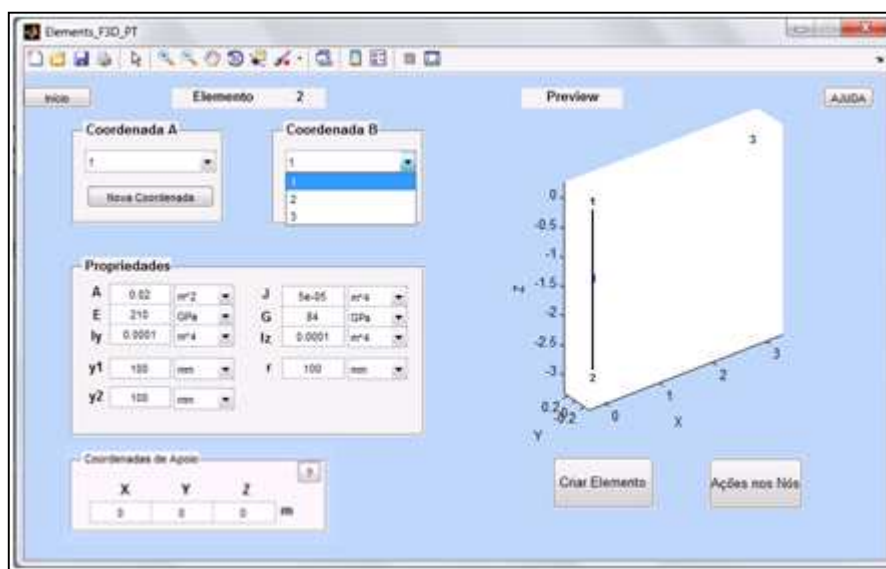


Figura 39 Novo ponto criado

O utilizador deverá então selecionar o par de nós que constituem o novo elemento e, à semelhança do primeiro elemento criado, definir as propriedades para este novo elemento. Deve-se ter particular atenção ao campo das coordenadas de suporte para a transformação de coordenadas. Uma vez que cada elemento terá o seu próprio ponto de suporte, os campos encontrar-se-ão sempre sem pré-preenchimento, sendo o valor mostrado o valor por defeito de 0 (zero).

Após a criação dos elementos necessários para o estudo o utilizador deverá selecionar o botão “Ações nos Nós” para ser lançada a janela da Figura 40.

4.4.4 Introdução de condições de análise, fronteira e carregamentos

Com a janela da Figura 40, o utilizador tem à sua disposição a possibilidade de introdução de restrições e ações sobre os nós. Deve-se referir que nesta janela estão disponíveis os botões:

- “Mais Elementos” - permite ao utilizador criar mais elementos para a estrutura em análise caso sejam necessários; este processo levará o utilizador até à secção anterior do programa, como ilustrado na Figura 37;

- “Salvar” - permite gravar sobre um ficheiro *.mat o estado atual do programa, ficando este ficheiro guardado no local indicado pelo utilizador. É aconselhado realizar este procedimento assim que forem introduzidas todas as ações sobre os nós;
- “Atualizar” - este botão deverá ser utilizado pelo utilizador de forma a pré-visualizar e guardar os dados introduzidos por nó nas variáveis;
- “Calcular” - dá ordem ao programa para realizar os cálculos necessários e é lançada a janela de resultados.

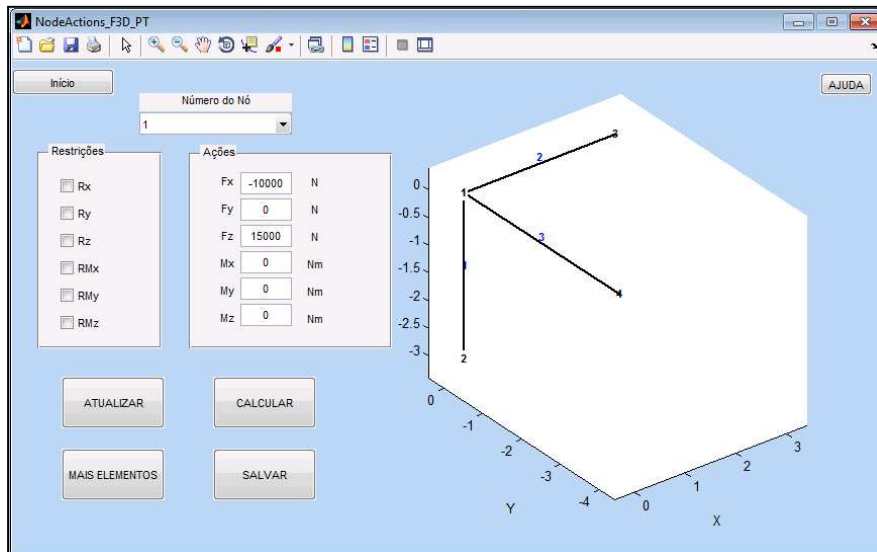


Figura 40 Janela ações nos nós

Nesta secção da análise, o utilizador deverá introduzir as restrições na estrutura e as ações exteriores. Uma vez que este programa permite unicamente a introdução destes parâmetros sobre os nós da estrutura, o utilizador deverá seleccionar o nó pretendido, usando o *popup menu* para este efeito (localizado no topo da janela). Depois de preencher as caixas relativas às restrições e às ações impostas, o utilizador deverá seleccionar o botão “Atualizar”. Com isto a janela é atualizada, como mostrado na Figura 41.

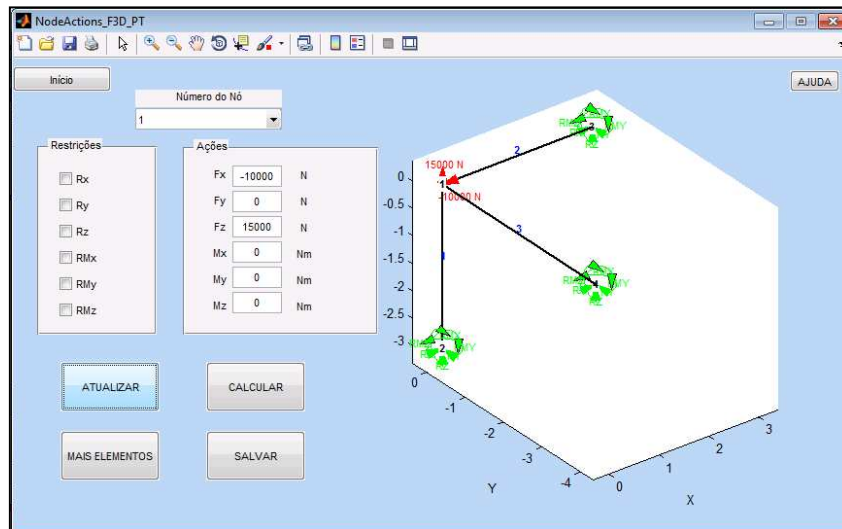


Figura 41 Ações e restrições sobre a estrutura

Deve-se clarificar que as indicações para as reações seguem o princípio vetorial, o que quer dizer que para um plano 2D as restrições seguem o princípio ilustrado na Tabela 6:

Tabela 6 Representação de reações em 2D

Tipo de apoio	Representação habitual	Representação no programa
Simple		
Duplo		
Encastrado		

Como consequência, para um nó encastrado em 3D, a representação vetorial no programa é do tipo da representação ilustrada na Figura 42:

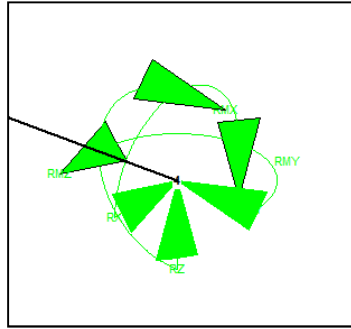


Figura 42 Representação de apoio encastrado em análise de elementos viga em 3D

Finalizada toda a introdução, o utilizador deverá seleccionar o botão “Calcular” para ser apresentada a janela de resultados, como mostra a Figura 43.

4.4.5 Apresentação de resultados

Inicialmente a janela de resultados mostrará a estrutura com as ações e imposições inicialmente impostas e com os resultados das reações. Deve-se informar que existem, na tabela dos resultados apresentados, valores desprezáveis. Tomemos o exemplo das reações: não deveriam existir valores para o nó 1, no entanto, os valores apresentados devem-se a valores residuais relativos ao cálculo matricial. Observe-se na Tabela 7 a comparação do valor residual com um valor real a considerar.

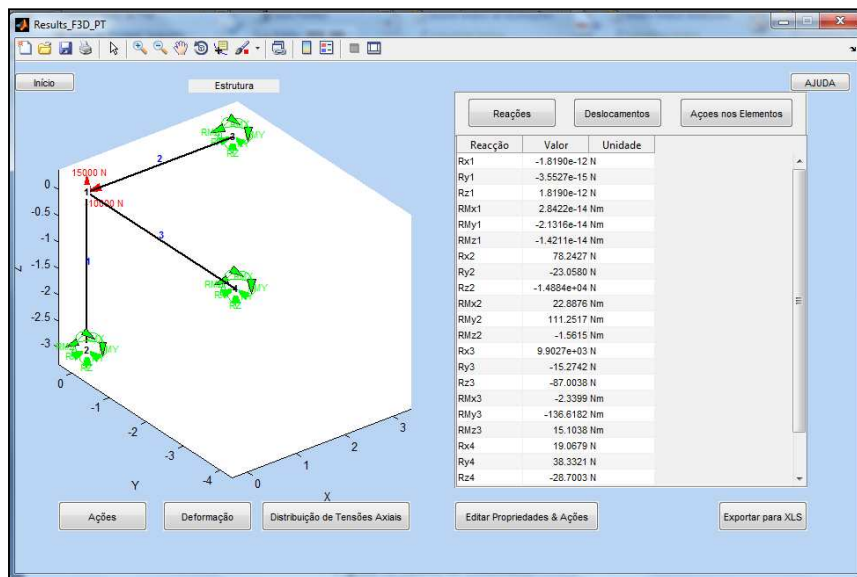


Figura 43 Janela de resultados

Tabela 7 Valor real vs. valor residual

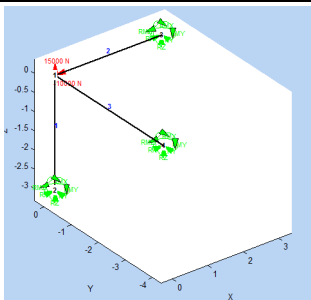
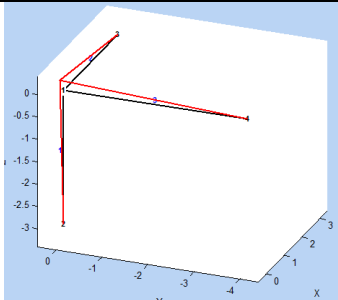
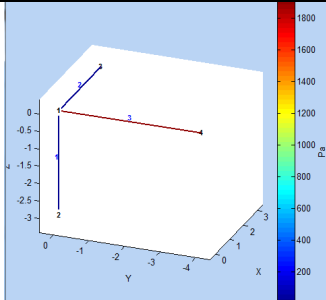
Valor real	Valor residual
$Rx_3 = 9.9027 * 10^3 \text{ N}$	$Rx_1 = -1.8190 * 10^{-12} \text{ N}$

Pode-se verificar pela ordem de grandeza quais os valores que devem ser considerados.

Estruturas para visualização

O utilizador tem à sua disposição 3 botões para visualizar a estrutura final. Na Tabela 8 são mostradas as estruturas resultantes para cada botão:

Tabela 8 Tipos de estruturas disponíveis para visualização

Botão “Ações”	Botão “Deformação”	Botão “Distribuição de tensões axiais”
		

Resultados tabelados

Existem 3 botões no topo da tabela de resultados. Estes botões irão alterar a informação exibida. Os botões apresentam a seguinte informação quando selecionados:

Botão “Reações”: Irá apresentar na tabela a informação das reações por nó. Será indicado o nó, a direção da reação, o valor e a unidade, como mostra a Figura 44;

Reacção	Valor	Unidade
Rx1	-1.8190e-12	N
Ry1	-3.5527e-15	N
Rz1	1.8190e-12	N
RMx1	2.8422e-14	Nm
RMy1	-2.1316e-14	Nm
RMz1	-1.4211e-14	Nm
Rx2	78.2427	N
Ry2	-23.0580	N
Rz2	-1.4884e+04	N
RMx2	22.8876	Nm
RMy2	111.2517	Nm

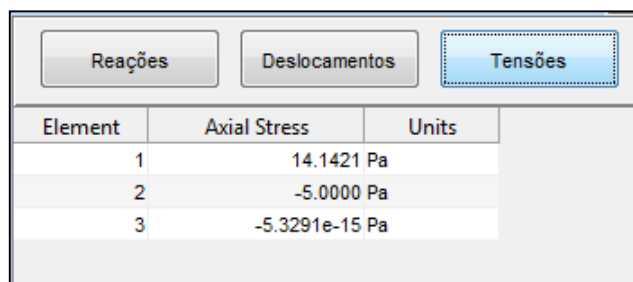
Figura 44 Resultados: Reações

Botão “Deslocamentos”: Com este botão a informação apresentada diz respeito aos valores das variáveis nodais por nó, conforme mostrado na Figura 45;

Direcção	Valor	Unidade
U1	-7.0733e-06	m
V1	-3.6507e-08	m
W1	1.0632e-05	m
Ou1	1.6713e-06	rad
Ov1	8.7320e-07	rad
Ow1	1.1154e-06	rad
U2	0	m
V2	0	m
W2	0	m
Ou2	0	rad
Ov2	0	rad
Ow2	0	rad
U3	0	m

Figura 45 Resultados: Deslocamentos

Botões “Ações nos Elementos” e “Tensões”: Ao serem pressionados terão respostas diferentes. Para casos de análise de elementos de barra 2D e 3D, irá ser apresentada a informação da distribuição da tensão axial por elemento, como ilustrado na Figura 46.



Element	Axial Stress	Units
1	14.1421 Pa	
2	-5.0000 Pa	
3	-5.3291e-15 Pa	

Figura 46 Tensões para elementos de barra

Para os casos de análise de elementos viga 2D e 3D será exibida uma janela advertindo o utilizador que os cálculos realizados para as tensões seguem o princípio de aproximação de perfil circular, conforme mostrado na Figura 47. Também são dados a conhecer ao utilizador os resultados das forças resultantes nos nós dos elementos, de forma a poder realizar uma aproximação mais correta, de forma manual e com o conhecimento exato da secção de cada elemento.

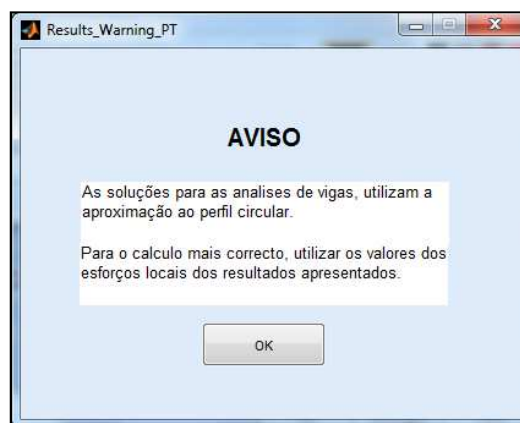
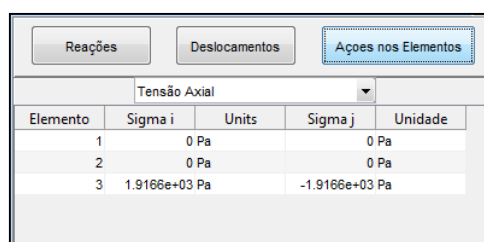


Figura 47 Aviso das soluções das ações nos elementos

Este aviso apenas será dado a 1º vez que o utilizador pressionar o botão com os resultados atuais. Qualquer alteração no problema em análise irá reativar o aviso da Figura 47. Após o utilizador ter pressionado “OK”, o *layout* da tabela de resultados é alterado para o da Figura 48, aparecendo por baixo dos botões de resultados o *popup menu* seletor de resultados para as ações nos elementos. Todos os resultados são dispostos pelos nós iniciais e finais dos elementos, como ilustra a Figura 48.



Tensão Axial				
Elemento	Sigma i	Units	Sigma j	Unidade
1	0 Pa		0 Pa	
2	0 Pa		0 Pa	
3	1.9166e+03 Pa		-1.9166e+03 Pa	

Figura 48 Resultados: Tensão axial

Ao pressionar o seletor de resultados aparecerão as opções de resultados, segundo o caso em estudo, como mostrado na Figura 49:

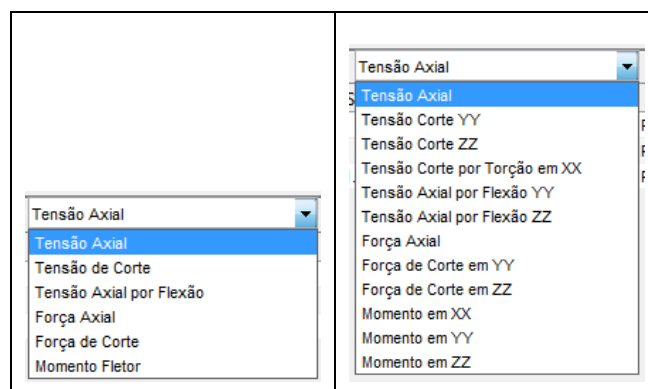


Figura 49 Comparação de seletores de resultados para as ações nos elementos. À esquerda para elementos de vigas 2D, à direita para elementos de viga 3D.

Aproximação ao perfil circular

Poucos são os códigos programáticos semelhantes ao desenvolvido nesta tese que permitam realizar o cálculo de tensões para elementos do tipo viga. Esta situação deve-se ao fato do cálculo das tensões ser dependente da geometria utilizada para o perfil. O código utilizado no programa realiza o cálculo das tensões sobre os nós dos elementos, utilizando uma aproximação ao perfil circular. Foi adotado para este processo de cálculo a metodologia habitualmente encontrada em literatura de mecânica dos materiais.

Por esta razão é aconselhado analisar preferencialmente os valores dos esforços locais e não os valores das tensões locais apresentadas nos resultados.

Carregamentos equivalentes

Deve-se lembrar que para facilitar a programação por elementos finitos, na presença de carregamentos distribuídos, aplica-se o critério dos carregamentos equivalentes (referido em [6], pagina 53 da Secção 3.4.2.). Por esta razão, os resultados apresentados para as ações nos nós para elementos de viga, são em função dos nós do elemento.

Exportar para XLS

Existe, abaixo da tabela de resultados, um botão “Exportar para XLS” que permite criar um ficheiro com todos os dados do problema em análise. Cada tipo de dados e resultados é separado pelas diferentes folhas de cálculos do ficheiro. Este ficheiro resultante pode ser utilizado como modelo para uma nova estrutura a analisar. Basta, para isso, alterar os dados das abas “NODES”, “PROPERTIES” e “ACTIONS”, e fazer importação da estrutura, conforme o processo explicado

na secção 3.4.1, Figura 34. Deve-se ter particular atenção que a disposição dos dados é diferente para os 4 tipos de análise realizados pelo programa, por isso é aconselhável criar, para este efeito (utilizando o programa), 4 modelos diferentes de relatórios XLS, de forma a abranger os 4 tipos de caso de estudo.

É possível acrescentar linhas para aumentar o número de nós ou de elementos, como mostrado na Figura 50.

	A	B	C	D	E	F	G
1	Tipo de es		4 Vigas 3D		F3D ex.xlsx		
2							
3	COORDINATES						
4	X	Y	Z				
5		0	0	0			
6		0	0	-3			
7		3	0	0			
8		0	-4	0			
9		3	0				
10							
11							
12							
13							
14							

Figura 50 Edição de dados na folha de cálculo

4.4.6 Edição de propriedades dos elementos e condições de análise

Existe, abaixo da tabela de resultados, um botão “Editar Propriedades & Ações”, como mostrado na Figura 43. Ao pressionar este botão é mostrada a janela ilustrada na Figura 51.

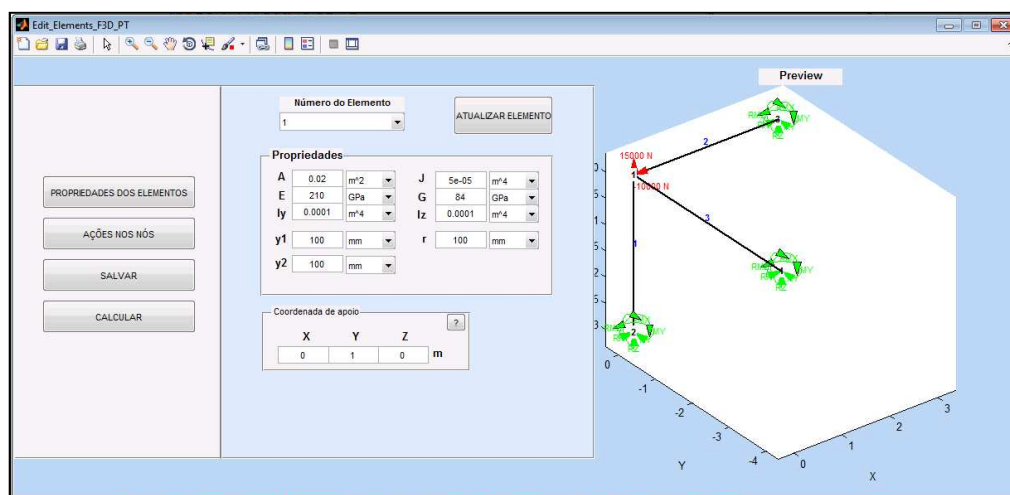


Figura 51 Janela edição, modo “propriedades”

Do lado esquerdo da janela estão dispostos os seguintes botões:

Propriedades dos elementos

Este botão altera a disposição da janela para o modo “propriedades” (modo mostrado por defeito na Figura 51), permitindo selecionar o elemento para o qual se querem alterar as propriedades. O *layout* para a parte central da janela segue o mesmo princípio aplicado nas janelas da criação dos elementos (Figura 37). No entanto, foram retirados os campos relativos às coordenadas. Após alterar as propriedades de cada elemento o utilizador deverá pressionar o botão “Atualizar elemento” de forma a atualizar as variáveis.

Ações nos nós

Ao pressionar este botão será alterada a disposição da janela para o modo “ações”, como mostrado na Figura 52. O *layout* da parte central da janela segue o mesmo princípio utilizado na janela mostrada na Figura 40, relativa à introdução de condições de análise. O utilizador deverá selecionar o nó cujas condições deseja alterar. Após as alterações do nó, o utilizador deverá pressionar o botão “Atualizar figura e nó” de forma a atualizar as variáveis.

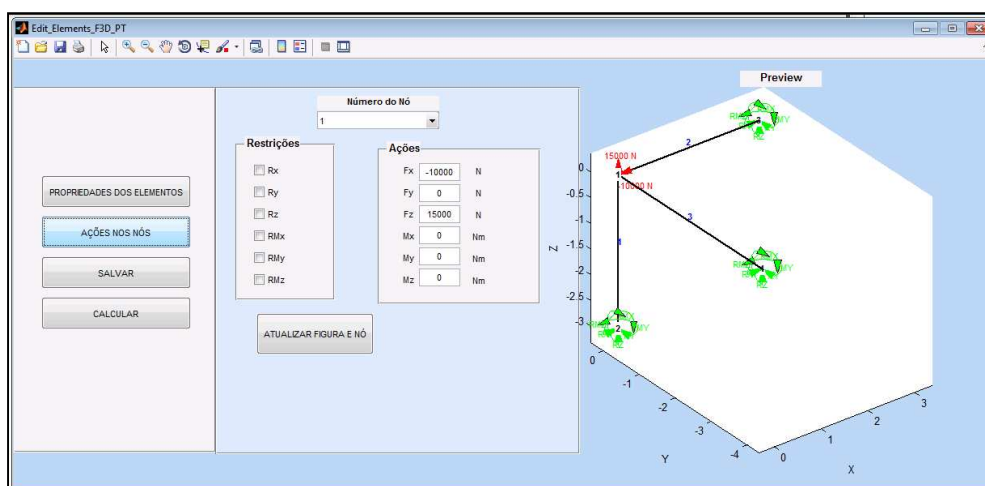


Figura 52 Janela edição, modo “ações”

Salvar

Este botão permite salvar o estado das variáveis sobre um ficheiro do tipo *.mat. O processo é em tudo semelhante ao descrito para o botão Salvar no capítulo 4.4.4.

Calcular

Ao pressionar este botão o programa refaz os cálculos com as novas variáveis e mostra ao utilizador a janela de resultados mostrada na Figura 43.

4.5 Validação do *software* desenvolvido

De forma a validar os resultados obtidos pelo programa, foram realizadas as comparações que se seguem abaixo para cada tipo de estudo. Os exercícios para a validação provieram de diferentes fontes.

4.5.1 Elementos de barra 2D

Seguindo o exercício titulado "First 2D truss problem, problem4.m" [2] realizou-se a seguinte comparação:

Problema original:

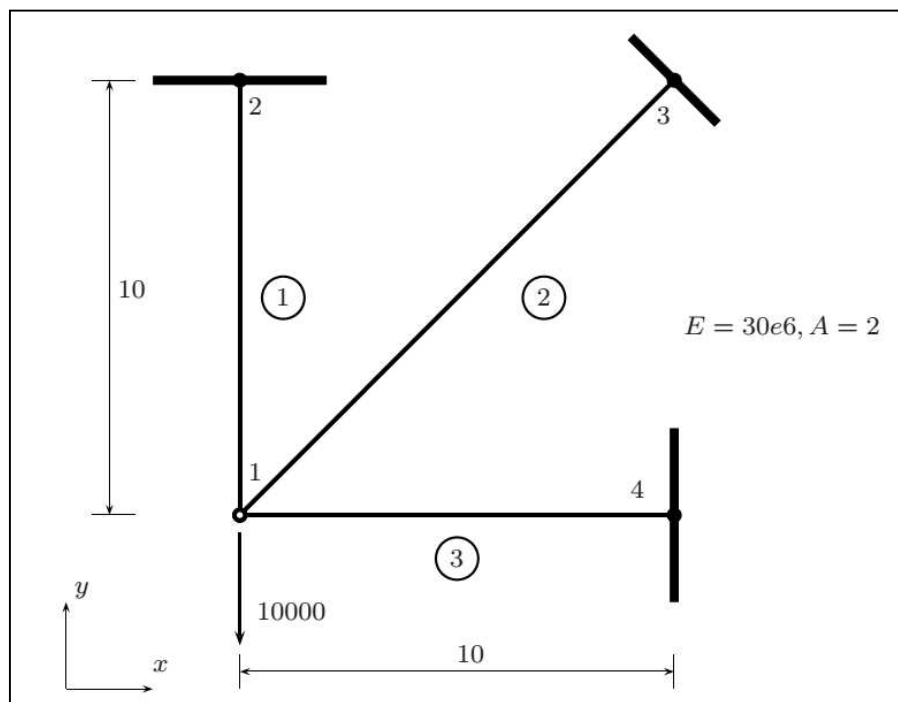


Figura 53 Problema para validação do estudo de elementos de barra 2D [2]

Formulação do problema no programa:

Com base dos dados recolhidos na Figura 53 criou-se a estrutura para análise mostrada na Figura 54.

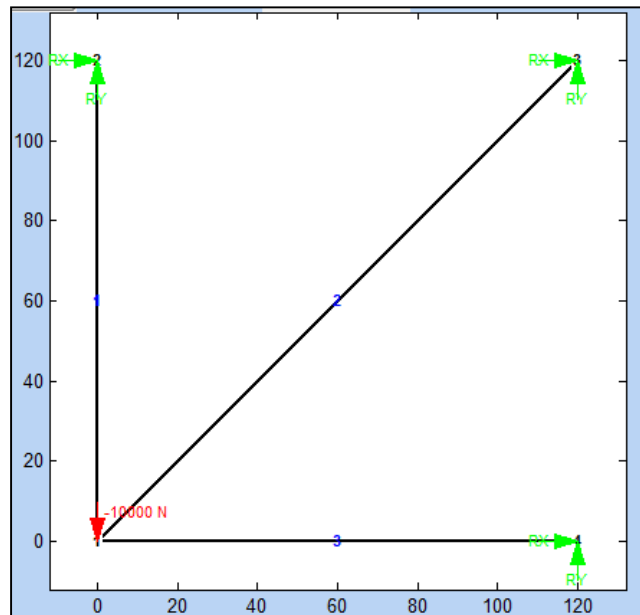


Figura 54 Estrutura para validação do estudo de elementos de barra 2D

Foram introduzidos os seguintes dados:

Coordenadas:

Tabela 9 Coordenadas utilizadas na validação do estudo de elementos de barra 2D

Nó	Coordenada X [m]	Coordenada Y [m]
1	0	0
2	0	10
3	10	10
4	10	0

Ligações entre elementos:

Tabela 10 Ligações utilizadas na validação do estudo de elementos de barra 2D

Elemento	Nó i	Nó j
1	1	2
2	1	3
3	1	4

Propriedades:

Tabela 11 Propriedades dos elementos na validação do estudo de elementos de barra 2D

Propriedade	Valor
Módulo de Young	$70 \cdot 10^6 \text{ Pa}$ (para todos os elementos)
Secção do elemento	2 m^2 (para todos os elementos)

Restrições:

Com exceção do nó 1, todos os nós se encontram encastrados.

Ações:

Foi aplicada uma carga de 10 kN no sentido negativo de Y sobre o nó 1.

Resultados:

Os resultados mostrados serão dispostos em grupos, sendo eles: deslocamentos, reações e tensões no elemento. Por cada grupo são mostrados os resultados previamente calculados e os resultados propostos pelo programa desenvolvido.

Deslocamentos:

Tabela 12 Resultados: Deslocamentos para validação do estudo de elementos de barra 2D

Deslocamentos originais [m]		Deslocamentos propostos [m]	
U1	0,0041	U1	0,0041
V1	-0,0159	V1	-0,0159
U2	0	U2	0
V2	0	V2	0
U3	0	U3	0
V3	0	V3	0
U4	0	U4	0
V4	0	V4	0

Como mostrado pela Tabela 12, verificou-se uma correspondência de 100% até os algarismos significativos utilizados pelo autor [2].

Reações:

Tabela 13 Resultados: Reações para validação do estudo de elementos de barra 2D

Reações originais [N]		Reações propostas [N]	
R _{x1}	0	R _{x1}	0
R _{y1}	0	R _{y1}	1,8189*10 ⁻¹²
R _{x2}	0	R _{x2}	0
R _{y2}	7928,9	R _{y2}	7928,932
R _{x3}	2071,1	R _{x3}	2071,068
R _{y3}	2071,1	R _{y3}	2071,068
R _{x4}	-2071,1	R _{x4}	-2071,068
R _{y4}	0	R _{y4}	0

Como mostrado pela Tabela 13, verificou-se uma correspondência de 100% para os valores a considerar, até os algarismos significativos utilizados pelo autor [2]. O valor de R_{y1} é um resultado desprezável, como referido na secção 4.4.5.

Tensões:

Tabela 14 Resultados: Tensões para validação do estudo de elementos de barra 2D

Tensões originais [Pa]		Tensões propostas [Pa]	
Elemento 1	3964.5	Elemento 1	3964.466
Elemento 2	1464.5	Elemento 2	1464.466
Elemento 3	-1035.5	Elemento 3	-1035.534

Como mostrado pela Tabela 14, verificou-se uma correspondência de 100%, até os algarismos significativos utilizados pelo autor [2].

Conclusão: Para uma análise do tipo elemento de barra 2D, o programa encontra-se validado.

4.5.2 Elementos de barra 3D

Seguindo o exercício titulado "Exemplo 1" na página 42 de [6], realizou-se a seguinte comparação:

Problema original:

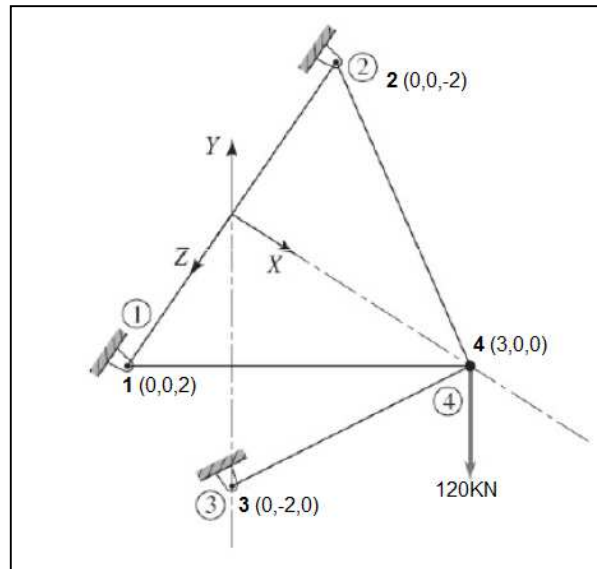


Figura 55 Problema para validação do estudo de elementos barra 3D [6]

Formulação do problema no programa:

Com base dos dados recolhidos na de Figura 55 criou-se a estrutura para análise mostrada na Figura 56. Deve-se referir que devido ao modo de criação do 1º elemento por parte do programa, o nó referido como 4 no exercício original, passará a ser referido como nó 2. Isto traz como consequência que:

- O nó 2 original passe a ser chamado de nó 3;
- O nó 3 original passe a ser chamado de nó 4.

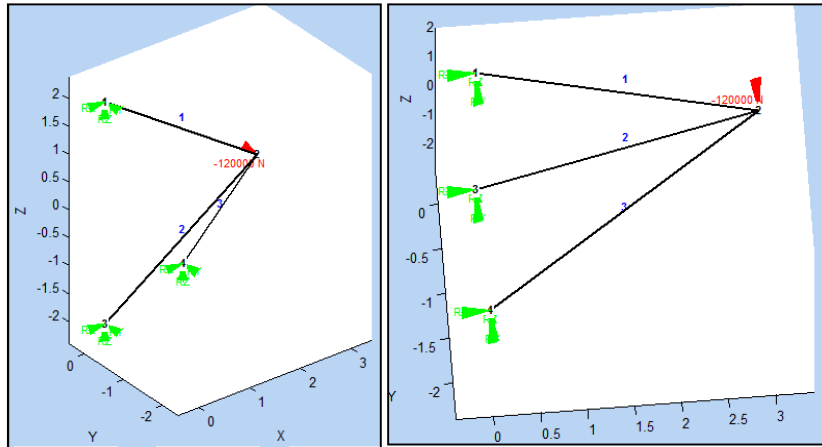


Figura 56 Estrutura para validação do estudo de elementos de barra 3D

Foram introduzidos os seguintes dados:

Coordenadas:

Tabela 15 Coordenadas utilizadas na validação do estudo de elementos de barra 3D

Nó	Coordenada X [m]	Coordenada Y [m]	Coordenada Z [m]
1	0	0	2
2	3	0	0
3	0	0	-2
4	0	-2	0

Ligações entre elementos:

Tabela 16 Ligações utilizadas na validação do estudo de elementos de barra 3D

Elemento	Nó i	Nó j
1	1	2
2	3	2
3	4	2

Propriedades:

Tabela 17 Propriedades dos elementos do problema na validação do estudo de elementos de barra 3D

Propriedade	Valor
Módulo de Young	210 MPa (para todos os elementos)
Secção do elemento	20 cm ² (para todos os elementos)

Restrições:

Com exceção do nó 2, todos os nós se encontram encastrados.

Ações:

Foi aplicada uma carga de 120 kN no sentido negativo de Y sobre o nó 2.

Resultados:

Os resultados mostrados serão dispostos em grupos, sendo eles: deslocamentos, reações e tensões no elemento. Por cada grupo são mostrados os resultados previamente calculados e os resultados propostos pelo programa.

Deslocamentos:

Tabela 18 Resultados: Deslocamentos para validação do estudo de elementos de barra 3D

Deslocamentos originais [m]		Deslocamentos propostos [m]	
U1	0	U1	0
V1	0	V1	0
W1	0	W1	0
U2	$1,116 \cdot 10^{-3}$	U2	$1,116 \cdot 10^{-3}$
V2	$-5,022 \cdot 10^{-3}$	V2	$-5,022 \cdot 10^{-3}$
W2	0	W2	0
U3	0	U3	0
V3	0	V3	0
W3	0	W3	0
U4	0	U4	0
V4	0	V4	0
W4	0	W4	0

Como mostrado pela Tabela 18, verificou-se uma correspondência de 100%, até os algarismos significativos utilizados pelo autor [6].

Reações:

Tabela 19 Resultados: Reações para validação do estudo de elementos de barra 3D

Reações originais [N]		Reações propostas [N]	
R _{x1}	-90000	R _{x1}	-90000
R _{y1}	0	R _{y1}	0
R _{z1}	60000	R _{z1}	60000
R _{x2}	0	R _{x2}	0
R _{y2}	0	R _{y2}	0
R _{z2}	0	R _{z2}	0
R _{x3}	-90000	R _{x3}	-90000
R _{y3}	0	R _{y3}	0
R _{z3}	60000	R _{z3}	60000
R _{x4}	180000	R _{x4}	180000
R _{y4}	120000	R _{y4}	120000
R _{z4}	0	R _{z4}	0

Como mostrado pela Tabela 19, verificou-se uma correspondência de 100%, até os algarismos significativos utilizados pelo autor [6].

Tensões:

Para o problema referido em [6] não é calculada a tensão por elemento. No entanto, este problema foi desenvolvido durante as aulas do autor na disciplina "Método de Elementos Finitos", do mestrado de construções mecânicas no ISEP. Segue-se os resultados obtidos para o mesmo exercício.

Tabela 20 Resultados: Tensões para validação do estudo de elementos de barra 3D

Tensões originais [Pa]		Tensões propostas [Pa]	
Elemento 1	$5,408 \cdot 10^7$	Elemento 1	$5,408 \cdot 10^7$
Elemento 2	$5,408 \cdot 10^7$	Elemento 2	$5,408 \cdot 10^7$
Elemento 3	$-1,1 \cdot 10^8$	Elemento 3	$-1,1 \cdot 10^8$

Como mostrado pela Tabela 20, verificou-se uma correspondência de quase 100%, até os algarismos significativos utilizados pelo autor [6].

Conclusão: Para uma análise do tipo elemento de barra 3D, o programa encontra-se validado.

4.5.3 Elementos de viga 2D

Seguindo o exercício titulado "2D frames, problem11.m" [2] realizou-se a seguinte comparação:

Problema original:

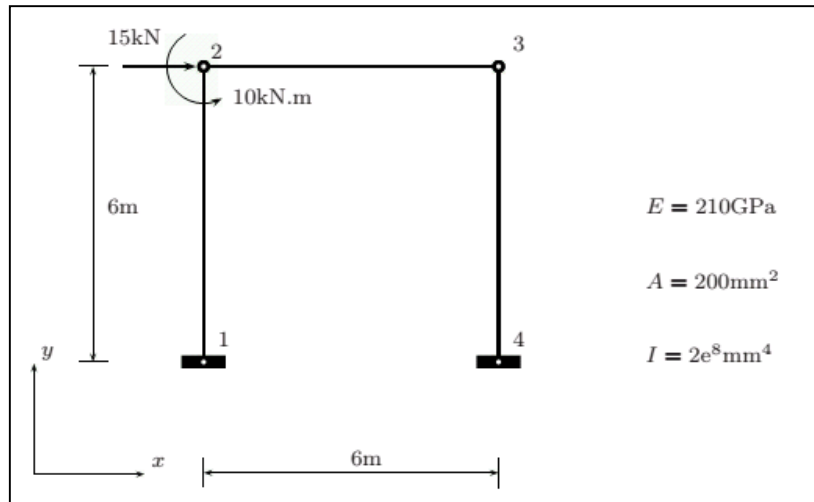


Figura 57 Problema para validação do estudo de elementos de viga 2D [2]

Formulação do problema no programa:

Com base dos dados recolhidos na Figura 57 criou-se a estrutura para análise mostrada na Figura 58.

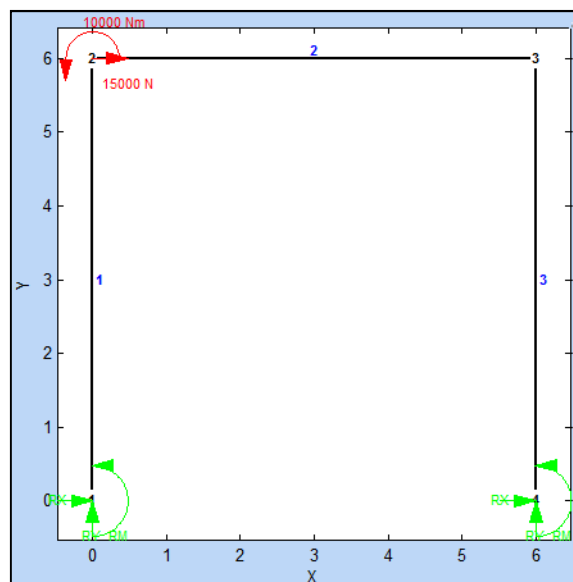


Figura 58 Estrutura para validação do estudo de elementos de viga 2D

Foram introduzidos os seguintes dados:

Coordenadas:

Tabela 21 Coordenadas utilizadas na validação do estudo de elementos de viga 2D

Nó	Coordenada X [m]	Coordenada Y [m]
1	0	0
2	0	6
3	6	6
4	6	0

Ligações entre elementos:

Tabela 22 Ligações utilizadas na validação do estudo de elementos de viga 2D

Elemento	Nó <i>i</i>	Nó <i>j</i>
1	1	2
2	2	3
3	4	4

Propriedades:

Tabela 23 Propriedades dos elementos na validação de estudo de elementos de viga 2D

Propriedade	Valor
Módulo de Young	210 MPa (para todos os elementos)
Secção do elemento	200 mm ² (para todos os elementos)
Momento de inércia à flexão	2*10 ⁸ mm ² (para todos os elementos)

Restrições:

Os nós 1 e 4 encontram-se encastrados.

Ações:

Foi aplicada uma carga de 15 kN no sentido positivo de X e um momento positivo de valor 10 kN.m, ambos sobre o nó 2.

Resultados:

Os resultados mostrados serão dispostos em grupos, sendo eles: deslocamentos, reações e ações nos elementos. Por cada grupo são mostrados os resultados previamente calculados e os resultados propostos pelo programa.

Deslocamentos:

Tabela 24 Resultados: Deslocamentos para validação do estudo de elementos de viga 2D

Deslocamentos originais		Deslocamentos propostos	
U1	0	U1	0
V1	0	V1	0
$\Theta 1$	0	$\Theta 1$	0
U2	$5,284 \cdot 10^3$ [m]	U2	$5,284 \cdot 10^3$ [m]
V2	$6,522 \cdot 10^4$ [m]	V2	$6,522 \cdot 10^4$ [m]
$\Theta 2$	$-5 \cdot 10^4$ [rad]	$\Theta 2$	$-4,977 \cdot 10^4$ [rad]
U3	$4,405 \cdot 10^3$ [m]	U3	$4,405 \cdot 10^3$ [m]
V3	$-6,522 \cdot 10^4$ [m]	V3	$-6,522 \cdot 10^4$ [m]
$\Theta 3$	$-6 \cdot 10^4$ [rad]	$\Theta 3$	$-5,893 \cdot 10^4$ [rad]
U4	0	U4	0
V4	0	V4	0
$\Theta 4$	0	$\Theta 4$	0

Como mostrado pela Tabela 24, verificou-se uma correspondência quase total. Esta diferença é causada por arredondamentos.

Reações:

Tabela 25 Resultados: Reações para validação do estudo de elementos de viga 2D

Reações originais		Reações propostas	
R _{x1}	$-9 \cdot 10^3$ [N]	R _{x1}	$-8,846 \cdot 10^3$ [N]
R _{y1}	$-5 \cdot 10^3$ [N]	R _{y1}	$-4,565 \cdot 10^3$ [N]
R _{m1}	$30,022 \cdot 10^3$ [N.m]	R _{m1}	$30,022 \cdot 10^3$ [N.m]
R _{x2}	0	R _{x2}	$-1,455 \cdot 10^{-11}$ [N]
R _{y2}	0	R _{y2}	$-9,095 \cdot 10^{-13}$ [N]
R _{m2}	0	R _{m2}	$-3,638 \cdot 10^{-12}$ [N.m]
R _{x3}	0	R _{x3}	$1,273 \cdot 10^{-11}$ [N]
R _{y3}	0	R _{y3}	$9,095 \cdot 10^{-13}$ [N]
R _{m3}	0	R _{m3}	0
R _{x4}	$-6 \cdot 10^3$ [N]	R _{x4}	$-6,154 \cdot 10^3$ [N]
R _{y4}	$5 \cdot 10^3$ [N]	R _{y4}	$4,565 \cdot 10^3$ [N]
R _{m4}	$22,586 \cdot 10^3$ [N.m]	R _{m4}	$22,586 \cdot 10^3$ [N.m]

Como mostrado pela Tabela 25, verificou-se uma correspondência quase total para os valores a considerar. Esta diferença é motivada por arredondamentos efetuados. Os valores de R_{x2}, R_{y2}, R_{m2}, R_{x3}, R_{y3} e R_{m3}, são resultados desprezáveis, como referido na secção 4.4.5.

Ações nos elementos:

Uma vez que para o exercício proposto não existem resultados para as ações por nós dos elementos, não é possível continuar com a comparação. Outra razão para não continuar a comparação deve-se ao fato dos resultados para as tensões por nós de elementos, serão baseadas na aproximação do perfil, circular como referido na secção sobre o tema no capítulo 4.4.5.

Conclusão: Para uma análise do tipo elemento de barra 2D, o programa encontra-se validado.

4.5.4 Elementos de viga 3D

Seguindo o exercício titulado "Exemplo 1" do capítulo 3.6 do livro [6] realizou-se a seguinte comparação:

Problema original:

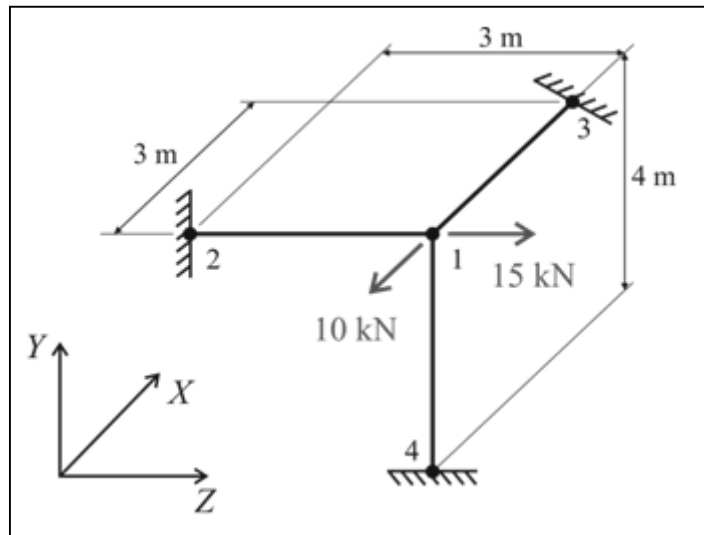


Figura 59 Problema para validação do estudo de elementos viga 3D [6]

Formulação do problema no programa:

Com base dos dados recolhidos na Figura 59 criou-se a estrutura para análise mostrada na Figura 60.

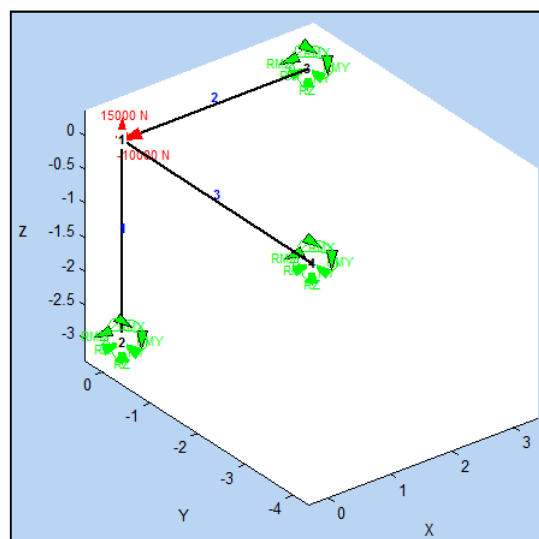


Figura 60 Estrutura para validação do estudo de elementos de viga 3D

Foram introduzidos os seguintes dados:

Coordenadas:

Tabela 26 Coordenadas utilizadas na validação do estudo de elementos de viga 3D

Nó	Coordenada X [m]	Coordenada Y [m]	Coordenada Z [m]
1	0	0	2
2	0	0	-3
3	3	0	0
4	0	-4	0

Ligações entre elementos:

Tabela 27 Ligações utilizadas na validação do estudo de elementos de viga 3D

Elemento	Nó <i>i</i>	Nó <i>j</i>
1	1	2
2	1	3
3	1	4

Pontos de suporte:

Tabela 28 Pontos de suporte para indicação de orientação das vigas

Elemento	P3x [m]	P3y [m]	P3z [m]
1	0	1	0
2	0	1	0
3	1	0	0

Propriedades:

Tabela 29 Propriedades dos elementos na validação de estudo de elementos de viga 3D

Propriedade	Valor
Módulo de Young	210 MPa (para todos os elementos)
Módulo de elasticidade transversal	84 MPa (para todos os elementos)
Secção do elemento	$2 \cdot 10^{-2} \text{ m}^2$ (para todos os elementos)
Momento de inércia à flexão segundo y	$10 \cdot 10^{-5} \text{ m}^4$ (para todos os elementos)
Momento de inércia à flexão segundo z	$10 \cdot 10^{-5} \text{ m}^4$ (para todos os elementos)
Momento polar de inércia	$5 \cdot 10^{-5} \text{ m}^4$ (para todos os elementos)

Restrições:

Com exceção do nó 1, todos os nós se encontram encastrados.

Ações:

Foram aplicadas as seguintes cargas no nó 1:

- Força de 10 kN no sentido negativo de X;
- Força de 15 kN na direção Z.

Resultados:

Os resultados mostrados serão dispostos em grupos, sendo eles: deslocamentos, reações e tensões no elemento. Por cada grupo são mostrados os resultados previamente calculados e os resultados propostos pelo programa.

Deslocamentos:

Tabela 30 Resultados: Deslocamentos para validação do estudo de elementos de viga 3D

Deslocamentos originais		Deslocamentos propostos	
U1	$-7,073 \cdot 10^{-6}$ [m]	U1	$-7,073 \cdot 10^{-6}$ [m]
V1	$-3,651 \cdot 10^{-8}$ [m]	V1	$-3,651 \cdot 10^{-8}$ [m]
W1	$1,063 \cdot 10^{-6}$ [m]	W1	$1,063 \cdot 10^{-6}$ [m]
Θ U1	$1,671 \cdot 10^{-6}$ [rad]	Θ U1	$1,671 \cdot 10^{-6}$ [rad]
Θ V1	$8,732 \cdot 10^{-7}$ [rad]	Θ V1	$8,732 \cdot 10^{-7}$ [rad]
Θ Z1	$1,115 \cdot 10^{-6}$ [rad]	Θ Z1	$1,115 \cdot 10^{-6}$ [rad]
U2	0	U2	0
V2	0	V2	0
W2	0	W2	0
Θ U2	0	Θ U2	0
Θ V2	0	Θ V2	0
Θ W2	0	Θ W2	0
U3	0	U3	0
V3	0	V3	0
W3	0	W3	0
Θ U3	0	Θ U3	0
Θ V3	0	Θ V3	0
Θ W3	0	Θ W3	0
U4	0	U4	0
V4	0	V4	0
W4	0	W4	0
Θ U4	0	Θ U4	0
Θ V4	0	Θ V4	0
Θ W4	0	Θ W4	0

Como mostrado pela Tabela 30, verificou-se uma correspondência 100%, para os algoritmos significativos utilizados pelo autor [6].

Reações:

Tabela 31 Resultados: Reações para validação do estudo de elementos de viga 3D

Reações originais		Reações propostas	
Rx1	-1,819*10 ⁻¹² [N]	Rx1	-1,819*10 ⁻¹² [N]
Ry1	-3,552*10 ⁻¹⁵ [N]	Ry1	-3,552*10 ⁻¹⁵ [N]
Rz1	1,819*10 ⁻¹² [N]	Rz1	1,819*10 ⁻¹² [N]
RMx1	2,842*10 ⁻¹⁴ [N.m]	RMx1	2,842*10 ⁻¹⁴ [N.m]
RMy1	-2,132*10 ⁻¹⁴ [N.m]	RMy1	-2,132*10 ⁻¹⁴ [N.m]
RMz1	-1,421*10 ⁻¹⁴ [N.m]	RMz1	-1,421*10 ⁻¹⁴ [N.m]
Rx2	78,242 [N]	Rx2	78,242 [N]
Ry2	-23,058 [N]	Ry2	-23,058 [N]
Rz2	-14884,296 [N]	Rz2	-14884,296 [N]
RMx2	22,8876 [N.m]	RMx2	22,8876 [N.m]
RMy2	111,252 [N.m]	RMy2	111,252 [N.m]
RMz2	-1,562 [N.m]	RMz2	-1,562 [N.m]
Rx3	9902,689 [N]	Rx3	9902,689 [N]
Ry3	-15,274 [N]	Ry3	-15,274 [N]
Rz3	-87,004 [N]	Rz3	-87,004 [N]
RMx3	-2,339 [N.m]	RMx3	-2,339 [N.m]
RMy3	-136,618 [N.m]	RMy3	-136,618 [N.m]
RMz3	15,104 [N.m]	RMz3	15,104 [N.m]
Rx4	19,068 [N]	Rx4	19,068 [N]
Ry4	38,332 [N]	Ry4	38,332 [N]
Rz4	-28,700 [N]	Rz4	-28,700 [N]
RMx4	-66,175 [N.m]	RMx4	-66,175 [N.m]
RMy4	-0,917 [N.m]	RMy4	-0,917 [N.m]
RMz4	-43,991 [N.m]	RMz4	-43,991 [N.m]

Como mostrado pela Tabela 31, verificou-se uma correspondência total para os valores considerados, para os algarismos significativos utilizados pelo autor [6]. Note-se que, à semelhança dos resultados do programa desenvolvido, os resultados de referência também possuem valores desprezáveis para as reações do nó 1.

Ações nos elementos:

Mais uma vez, o exercício proposto para validação da análise do elemento de viga 3D não inclui resultados para as ações por nós dos elementos, e como tal, não é possível continuar com a comparação. Outra razão para não continuar a comparação deve-se ao fato dos resultados para as tensões por nós de elementos, estão baseadas na aproximação do perfil circular, como referido na secção sobre o tema no capítulo 4.4.5.

Conclusão: Para uma análise do tipo elemento de barra 3D, o programa encontra-se validado.

5 Conclusões

A partir dos objetivos mencionados no início deste trabalho, pode-se afirmar que o programa desenvolvido permite, de forma clara, realizar as diferentes análises propostas para elementos unidimensionais de estruturas com ligações rígidas e rotuladas, utilizando como base o método de elementos finitos.

Conseguiu-se implementar uma interface de fácil interpretação, onde é pré-visualizada a estrutura a cada passo da sua criação. A referida interface possibilita ainda, a introdução de propriedades distintas para os diferentes elementos criados. Durante a introdução das condições fronteira e cargas por nós, foi conseguida a pré-visualização das posições das ditas restrições e cargas utilizando informação gráfica sob forma de vetores. De igual forma foi possível mostrar nos resultados a estrutura deformada e a distribuição de tensões ao longo da estrutura final. Os resultados são também apresentados sob forma tabular dentro da janela respetiva, permitindo uma fácil leitura sobre os mesmos.

Atingiram-se também os sub-objetivos propostos. O programa permite a introdução de dados usando como recurso ficheiros previamente criados em estudos anteriores, quer no formato do Matlab® (*.mat), que no formato de folha de cálculo (*.xls e *.xlsx). No caso de ficheiros no formato de folha de cálculo, estes podem ser editados, permitindo agilizar o processo de introdução de dados a serem utilizados em futuras análises pelo programa. Ainda sobre os ficheiros no formato de folha de cálculo, estes podem ser facilmente utilizados para redigir relatórios de análise das estruturas analisadas pelo programa.

O programa também permite a edição das estruturas em análise até um certo limite. Pode-se editar as propriedades dos elementos, assim como as condições fronteira da estrutura. No entanto, não é possível editar a geometria da estrutura em análise. Esta imposição é motivada pelo fato da informação geométrica se encontra interligada em duas matrizes. A primeira contendo as coordenadas dos nós, e a segunda com a informação dos nós dos elementos. Caso se pretenda alterar as coordenadas para um nó de um dado elemento, isto iria trazer como consequência uma alteração para todos os elementos ligados a esse nó. O mesmo acontece se se pretender eliminar um

nó. Os elementos ligados ao dito nó iriam perder as suas coordenadas, criando desequilíbrio no programa e no método de elementos finitos.

Em função do exposto pode-se concluir que o programa conseguiu atingir todos os objetivos propostos, conseguindo-se assim um programa de melhor qualidade comparativamente àqueles que se encontram atualmente disponíveis, relativamente à informação disponível durante o processo de criação da estrutura e à possibilidade de edição da mesma, sem por de parte, a forma como são apresentados os resultados e a possibilidade de exportação dos mesmos. Deve-se referir que, apesar disto, o programa pode ser melhorado. Como proposta de melhorias surgem os seguintes pontos:

- Condensação do código utilizado, evitando repetições. Para este efeito, podem ser implementadas, funções externas ao código utilizado nas janelas programáticas;
- Redução do número de janelas. Verificou-se que o *layout* utilizado nas diferentes janelas para os 4 tipos de análise é muito semelhante. Estas diferenças traduzem-se em, botões e campos de escrita que são modificados entre os 4 tipos de análise. Por esta razão, a utilização de camadas que podem ser ativadas ou não, dependendo das necessidades do caso em estudo, é uma solução a implementar;
- A implementação da edição geométrica da estrutura;
- Capacidade de importação de ficheiros do tipo CAD;
- Eliminar os resultados residuais e desprezáveis, introduzindo no código funções de arredondamento para zero, em função da ordem de grandeza dos restantes resultados;
- Possibilidade de introdução de cargas distribuídas, evitando o critério dos carregamentos equivalentes;
- Aumentar o numero de tipos de análise disponível, permitindo efetuar análises por elementos 2D e 3D;
- Criação de bibliotecas com geometria de perfis para elementos viga, de forma a evitar a aproximação ao elemento circular.

Referências

- [1] Y. Liu, Finite Element Method, Beijing, China: Tsinghua University Press, 2009.
- [2] A. J. M. Ferreira, MATLAB Codes for Finite Element Analysis, Porto, Portugal: Springer, 2009.
- [3] A. J. M. Ferreira, Método dos Elementos Finitos, Porto, Portugal: Faculdade de Engenharia da Universidade do Porto, 2003.
- [4] H. L. Soriano e S. S. Lima, Método de Elementos Finitos em Análise de estruturas Vol. 48, São Paulo, Brasil: Edusp, 2002.
- [5] M. A. Austin, Engineering Programing in MATLAB: A Primer, Maryland, EUA: University of Maryland, 2000.
- [6] R. D. S. G. Campilho, Método de Elementos Finitos, Porto, Portugal: Publindústria, 2012.
- [7] S. S. Rao, The Finite Element Method in Engineering (fifth edition), Florida, EUA: HB, 2011.
- [8] C. G. a. J.-F. Remacle, “Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities,” Outubro 2014. [Online]. Available: <http://geuz.org/gmsh/>. [Acedido em Outubro 2014].
- [9] J. Fish e T. Belytschko, A first course in Finite Elements, New York, EUA: Wiley, 2007.
- [10] Engenhariacivil.com, “Engenhariacivil.com,” Engenhariacivil.com, Outubro 2014. [Online].

- Available: <http://www.engenhariacivil.com/rfem-4>. [Acedido em Outubro 2014].
- [11] A. F. Alves, “CAD xpert.com.br,” CAD xpert, 01 Julho 2012. [Online]. Available: <http://www.cadxpert.com.br/ideias/aplicacoes-e-beneficios-da-analise-por-elementos-finitos/>. [Acedido em Outubro 2014].
- [12] ESSS, “ESSS.com.br,” ESSS, Outubro 2014. [Online]. Available: <http://www.esss.com.br/blog/pt/category/artigos-tecnicos/tecnologia-de-simulacao/multiphysics/>. [Acedido em Outubro 2014].
- [13] Field Precision software tips, “Electric and magnetic field calculations with finite-element methods,” Field Precision software tips, Outubro 2014. [Online]. Available: <http://fieldp.com/myblog/2014/electric-and-magnetic-field-calculations-with-finite-element-methods/>. [Acedido em Outubro 2014].
- [14] COMSOL, “COMSOL 4.3b Multiphysics Software - Release Highlights,” COMSOL, Outubro 2014. [Online]. Available: <http://www.comsol.com/products/4.3b/>. [Acedido em Outubro 2014].
- [15] J. Berger, A. Horton, G. Joldes, A. Wittek e K. Miller, “Coupling Finite Element and Mesh-free Methods for Modelling Brain Deformation in Response to Tumour Growth,” The University of Western Australia, Abril 2008. [Online]. Available: <http://www.midasjournal.org/browse/publication/567>. [Acedido em Outubro 2014].
- [16] J. Belinha e L. M. J. S. Dinis, “O MÉTODO LIVRE DE ELEMENTOS DE GALERKIN NA ANÁLISE NÃO-LINEAR DE ESTRUTURAS ANISOTRÓPICAS,” *Revista da Associação Portuguesa de Análise Experimental de Tensões*, n.º Método livre Galerkin, 2007.
- [17] T. Deisboeck, “On Simulating Growth and Form,” Biomedical Computation Review, Abril 2008. [Online]. Available: <http://biomedicalcomputationreview.org/content/simulating-growth-and-form>. [Acedido em Outubro 2014].

- [18] R. D. Cook, Finite Element modeling for stress analysis, New York, EUA: Wiley, 1995.
- [19] C. W.F., A primer for Finite Elements in elastic structures, New York, EUA: Wiley, 1999.
- [20] D. L. Logan, A First Course in the Finite Element Method (fourth edition), EUA: Thomson, 2007.
- [21] D. Hutton, Fundamentals of Finite Element Analysis, McGraw, 2004.
- [22] MSC Software, “MSC NASTRAN Multidisciplinary Structural Analysis,” MSC Software, Outubro 2014. [Online]. Available: <http://www.mscsoftware.com/product/msc-nastran>. [Acedido em Outubro 2014].
- [23] ANSYS, “ANSYS.com,” ANSYS, Outubro 2014. [Online]. Available: <http://www.ansys.com/>. [Acedido em Outubro 2014].
- [24] Dassault Systems, “Abaqus Unified FEA Complete solutions for realistic simulation,” Dassault Systems, Outubro 2014. [Online]. Available: <http://www.3ds.com/products-services/simulia/portfolio/abaqus/overview/>. [Acedido em Outubro 2014].
- [25] Wikipedia, “Abaqus,” Outubro 2014. [Online]. Available: <http://en.wikipedia.org/wiki/Abaqus>. [Acedido em Outubro 2014].
- [26] Dassault Systems, “SolidWorks,” Dassault Systems, Outubro 2014. [Online]. Available: <http://www.solidworks.com/>. [Acedido em Outubro 2014].
- [27] AUTODESK, “INVENTOR features,” AUTODESK, Outubro 2014. [Online]. Available: <http://www.autodesk.com/products/inventor/features/all/gallery-view>. [Acedido em Outubro 2014].
- [28] Mathworks, “Matlab Central,” Mathworks, Outubro 2014. [Online]. Available:

- http://www.mathworks.com/matlabcentral/fileexchange/?s_tid=gn_mlc_fx. [Acedido em Outubro 2014].
- [29] H. Rahami, “Matrix Structural Analysis,” Matlab File Exchange, Maio 2010. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/27012-matrix-structural-analysis>. [Acedido em Outubro 2014].
- [30] S. Asare, “2D-TRUSS WITH GUI PROGRAM BY YAN LUO & STEPHEN ASARE,” Matlab File Exchange, Fevereiro 2009. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/26755-2d-truss-with-gui-program-by-yan-luo---stephen-asare>. [Acedido em Outubro 2014].
- [31] R. Lahav, “Beam and Truss Analysis utilities,” Matlab File Exchange, Julho 2005. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/8008-beam-and-truss-analysis-utilities>. [Acedido em Outubro 2014].
- [32] Y. Bigdeli, “Structure analysis,” Dezembro 2012. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/39491-structure-analysis>. [Acedido em Outubro 2014].
- [33] M. C. Ruggeri, “Matrix Analysis of Three Dimensional Bar Structures MABS3D,” Matlab File Exchange, Maio 2010. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/27452-matrix-analysis-of-three-dimensional-bar-structures-mabs3d>. [Acedido em Outubro 2014].
- [34] A. Ozgul, “Space Truss Systems as Linear Static Analysis,” Matlab File Exchange, Fevereiro 2007. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/14020-space-truss-systems-as-linear-static-analysis>. [Acedido em Outubro 2014].
- [35] B. Ananthasayanam, “3D truss analysis\user interface in FEM,” Matlab File Exchange, Fevereiro 2005. [Online]. Available: <http://www.mathworks.com/matlabcentral/fileexchange/6832-3d-truss-analysis-user->

interface-in-fem. [Acedido em Outubro 2014].

- [36] R. D. Ziemian e W. McGuire, “MASTAN2 V3.3,” 2010. [Online]. Available: <http://www.mastan2.com/download.html>. [Acedido em Outubro 2014].
- [37] R. D. S. G. Campilho, “Código de calculo barra 3D - Campilho”.
- [38] U. D. Ahmad, “Academia.edu,” Academia.edu, Setembro 2013. [Online]. Available: http://www.academia.edu/6046519/Development_of_a_Structural_analysis_program_SABSM_using_MATLAB_featuring_graphical_user_interface. [Acedido em Outubro 2014].
- [39] Scilab Enterprises, “Scilab - About,” Scilab Enterprises, Outubro 2014. [Online]. Available: <http://www.scilab.org/scilab/about>. [Acedido em Outubro 2014].
- [40] GNU Octave, “GNU Octave,” GNU Octave, Outubro 2014. [Online]. Available: <https://www.gnu.org/software/octave/index.html>. [Acedido em Outubro 2014].
- [41] MathWorks, “MatLab Central,” MathWorks, Outubro 2014. [Online]. Available: <http://www.mathworks.com/matlabcentral/>. [Acedido em Outubro 2014].
- [42] WEISANG, “FlexPro 9,” WEISANG, Outubro 2014. [Online]. Available: <http://www.weisang.com/en/products/flexpro-9/get-to-know-flexpro/features/>. [Acedido em Outubro 2014].
- [43] Sage, “SageMath,” Outubro 2014. [Online]. Available: <http://www.sagemath.org/tour.html>. [Acedido em Outubro 2014].
- [44] S. Bhavikatti, Finite Element Analysis, New Dehli, India: 2005, 2005.
- [45] T. R. Chandrupatla e A. D. Belegundu, Introduction to Finite Elements in engieneering, Upper Sadle River, EUA: Prentice-Hall, 1997.

- [46] J. N. Reddy, Introduction to the Finite Element Method. Second Edition, New York, EUA: McGraw Hill, 1993.

- [47] J. J. E. D. J. Beer F.P., Mechanics of Materials, New York, EUA: McGraw Hill, 2006.

Bibliografia consultada

Além das referências bibliográficas anteriores, também foi consultada a seguinte bibliografia:

MATLAB - MathWoks, Creating Graphical User Interfaces, Massachusetts, EUA: The MathWorks, INC, 2002.

StackOverflow, “StackOverflow,” Outubro 2014. [Online]. Available: <http://stackoverflow.com/>. [Acedido em Outubro 2014].

Undocumented Matlab, “Undocumented Matlab - Charting Matlab's unsupported hidden underbely,” Outubro 2014. [Online]. Available: <http://undocumentedmatlab.com/>. [Acedido em Outubro 2014].

KOT Kotchergenki Engenharia, “KOT Kotchergenki Engenharia,” Outubro 2014. [Online]. Available: <http://www.kot.com.br/index.php>. [Acedido em Outubro 2014].

Y. W. Kwon e B. Hyochoong, The Finite Element Method using MATLAB, EUA: CRC Press, 1996.

M. Patrick, Graphics and GUIs with MATLAB (Third Edition), Florida, EUA: Chapman & Hall/CRC, 2003.

U. D. Ahmad, “Development of a Structura analysis program (SABSM) using MATLAB featuring graphical user interface,” City University London, Londres, Inglaterra, 2013.

T. R. Chandrupatla e A. D. Belegundu, Introduction to Finite Elements in engieneering, Upper Sadle River, EUA: Prentice-Hall, 1997.

A. J. M. Ferreira, Manual de Elementos Finitos, 2006.

A. B. M. Nasiruzzaman, Using MATLAB to develop standalone graphical user interface (GUI) software packages for educational purposes, Rajshahi, Bangladesh: Rajshahi University of Engeneering & Technology.

A. H. Register, A Guide to MATLAB Object-Oriented Programming, Atlanta, Georgia, EUA: Scitech, 2007.

A. Ponti, Advanced Programming in MATLAB, Suíça, 2010.

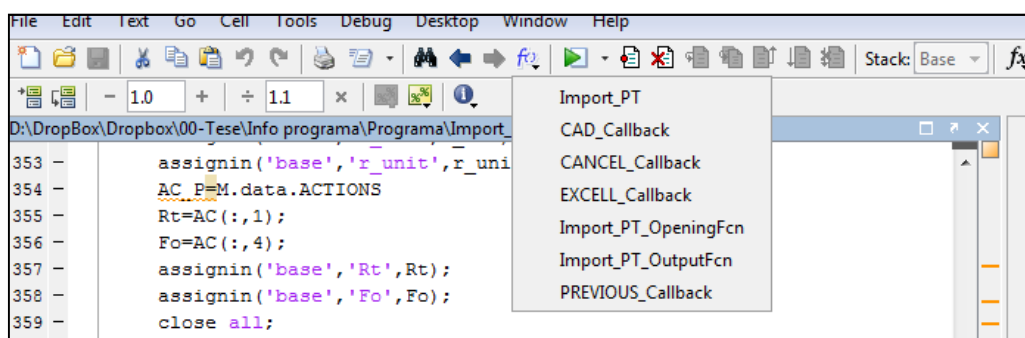
C. S. Lent, Learning to Program With MATLAB - Building GUI Tools, USA: MPS Limited, Chennai, 2013.

Anexo A. Guia e dicas de códigos Matlab®

Em esta secção o autor irá compilar os códigos utilizados na programação em Matlab® que por vezes não aparecem nos manuais do utilizador ou a sua aplicação não é tão direta ou perceptível.

Em primeiro lugar, aconselho vivamente a criar um esboço de qualquer programa em papel. Analisar as entradas e saídas de variáveis e prever o comportamento do programa.

Em segundo lugar, durante a criação de qualquer elemento das diferentes GUIs, atribuir etiquetas de nomes nas propriedades do elemento criado. Isto ajudará durante a programação, principalmente quando o programa se torne extenso. É possível saltar entre secções do ficheiro *.m da janela programada utilizando para esse efeito o botão para chamar funções. Por vezes será também necessário obrigar à criação das funções de certos elementos dentro das GUIs, pois nem todos são gerados de forma automática.



Após a criação de qualquer janela GUI, o Matlab® cria 2 ficheiros, um do tipo *.fig que contem a informação gráfica da janela, e um do tipo *.m, com a informação do comportamento. É este último ficheiro que terá de ser programado.

De maneira a agilizar a criação do código para qualquer programa, também é aconselhável utilizar o mesmo nome (etiqueta) de elementos semelhantes entre GUIs. Exemplo: se existe um campo que recolhe o mesmo tipo de informação em diferentes GUIs e o guarda no mesmo local, é bom ter o mesmo nome dos campos que o precedem. Assim evita-se alterar o código específico para cada campo.

Em qualquer altura da programação pode-se alterar o nome das etiquetas dos elementos presentes na GUI. O Matlab® atualiza os nomes dentro do M-file associado à janela GUI automaticamente. No entanto, é aconselhado verificar o código.

Chamar uma nova GUI:**Ex:**

```
%-----
%RUN THE PROGAM IN ENGLISH
%-----
close all;
run ('Type_EN');
```

Esta serie de comandos, fecha a as janelas atuais e abre uma nova janela. No caso de se pretender fechar uma janela específica, deve-se utilizar o nome da janela ('JanelaX') em vez de utilizar o comando `all`;

Mover GUI para o centro:

Este comando tem de ser chamado no início do código, na secção onde aparece a informação dos códigos executados antes da janela tornar-se visível. NOTA: O comando só funciona quando existe uma só janela. As subsequentes janelas em sobreposição não irão obedecer o código.

Ex:

```
% --- Executes just before Element1_2D_EN is made visible.
function Element1_2D_EN_OpeningFcn(hObject, eventdata, handles,
varargin)

% Choose default command line output for Element1_2D_EN
handles.output = hObject;

%-----
%moves the GUI to center
%-----
movegui('center')
```

Transitar variáveis entre janelas:

O Matlab® na sua base estipula um ambiente de trabalho por cada GUI, Programa, Script. Existindo ainda o ambiente da linha de comando designada como '`base`'. Assim, o melhor método de transitar variáveis entre janelas passa pela importação e exportação das variáveis por um ambiente de trabalho conhecido. Recomendo para este efeito o ambiente de trabalho '`base`'.

Importar e exportar variáveis para o Workspace geral:**Ex:**

```
%-----
%Importa e atualiza o contador de elementos
%-----
ElemNumb = evalin('base','ElemNumb'); %Importa do ambiente base a
variável ElemNumb

assignin('base','ElemNumb',ElemNumb); %Exporta para ambiente base a
variável ElemNumb
```

Imagens em Botões

Pode-se colocar imagens nos botões, tratando estes como espaços para PLOT. No entanto o método mais expedito passa pela mudança de propriedades dos botões durante a criação dos mesmos. Assim a imagem é guardada dentro do ficheiro *.fig.

- Duplo Click no botão para abrir o *Property Inspector*.
- Ir até o campo CData e introduzir o comando `imread('MyImageFile.jpg')`. Deve conter o caminho até a imagem caso ela não esteja na diretoria onde se encontra guardada a GUI.

Para colocar imagens tipo logo:

Existem dois métodos e nenhum funciona 100% dos casos.

1º Método

Obriga à criação de um espaço de PLOT com o nome pretendido, e criar a função de chamada usando o botão da direita do ponteiro. >object inspector>CreateFcn

Ex:

```
% --- Executes during object creation, after setting all properties.
function LOGO_CreateFcn(hObject, eventdata, handles)
% hObject    handle to LOGO (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

%----
%LOGO
%----
imshow('logo_test.BMP');
```

2º Método

O axes1 foi o nome atribuído à área de PLOT.

Ex:

```
%-----
%PREVIEW NOT AVAILABLE
%-----
f=imread('N_Avaliable.png');
imshow(f, 'Parent', handles.axes1);%modified
```

Perguntas código MATLAB®:

Quando se quer que uma janela automática do Matlab® faça uma pergunta ao utilizador antes de executar uma função.

Ex:

```
% --- Executes on button press in Home.
function Home_Callback(hObject, eventdata, handles)
% hObject      handle to Home (see GCBO)
% -----
%LOSE ALL AND GO BACK TO THE SELECTION SCREEN ALERT
% -----
question_ans = questdlg('Do you want to discard all and go back to the
"Choose analisys type screen"?',...
'GO BACK???', 'Yes', 'No', 'No')
if strcmp(question_ans, 'Yes')
close all;
run ('Type_EN');
end
```

Avisos código MATLAB®:

Quando se quer que uma janela automática do Matlab® faça forneça informação ao utilizador antes de executar uma função.

Ex:

```
% --- Executes on button press in CAD.
function CAD_Callback(hObject, eventdata, handles)
% hObject      handle to CAD (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
%-----
%NÃO DISPONIVEL EM ESTA VERSÃO
%-----
msgbox('Desculpe, mas não disponível em esta versão do programa', 'Not
```



```
Avaliable', 'error');
```

Pegar em valores das caixas de texto:

Ex:

(na função da caixa de texto)

```
function Xa_txt_Callback(hObject, eventdata, handles)

input = str2num(get(hObject, 'String'));
%checks to see if input is empty. if so, default input1_editText to zero
if isempty(input)
set(hObject, 'String', '0')
end
guidata(hObject, handles);
```

(na função da ação de recolha)

```
function NewElement_Callback(hObject, eventdata, handles)

Xa = get(handles.Xa_txt, 'String');
```

NOTA: esta última instrução também pode ser usada para recolher informação de qualquer função.

Para transformar uma variável texto (string) em valor numérico

```
str2num(Xa)
```

Popular uma Popup com uma vetor:

Normalmente as popup são populadas com valores predefinidos (texto ou numero). No entanto há situações em que é necessário uma popup dinâmica que seja atualizada conforme vai avançando o programa. Em este exemplo a informação é guardada em um vetor e posteriormente é mostrado o vetor na popup. Mais uma vez, este tipo de código deve ser introduzido na secção que executa as rotinas antes de mostrar a janela.

Ex:

```
%Para o PopUp A
M_teste = evalin('base', 'M_teste');
set(handles.Pop_Node_A, 'string', M_teste);
guidata(hObject, handles);
```

Pegar em valores do Popup populado e utilizar/gravar em vetor/matriz:**Ex:**

```
function Pop_Node_A_Callback(hObject, eventdata, handles)
%-----
%Selecciona o ponto
%-----
items_A = get(hObject, 'String');
index_selected_A = get(hObject, 'Value');
item_selected_A = items_A{index_selected_A};
display(item_selected_A);

function NewElement_Callback(hObject, eventdata, handles)
%-----
%Pega os pontos selecionados e cria nova linha na matriz ElemNodes
%-----
N_A=get(handles.Pop_Node_A, 'Value');
N_B=get(handles.Pop_Node_B, 'Value');
ElemNodes = evalin('base', 'ElemNodes');
ElemNodes=[ElemNodes;(N_A) (N_B)];
assignin('base', 'ElemNodes', ElemNodes);
```

Preseleccionar um valor numa popup na próxima vez que é chamada a janela**Ex:**

```
set(handles.Nodes_Pop, 'value', 3)
```

Sendo o valor 3 referente à posição pretendida**Código rápido para PopUp de Unidades.**

Esta porção de código recolhe a informação da caixa de texto para um valor numérico com unidades associadas e da PopUp que o acompanha mostrando a informação da unidade. Este tipo de PopUp é tradicional, pelo que foi introduzida a informação aquando da criação da PopUp na GUI. Este tipo de código é praticamente essencial quando além da informação a transitar (valor numérico) é requerido conhecer o que foi selecionado pelo utilizador.

Ex:

```
% --- Executes on button press in UPDATE_E.
function UPDATE_E_Callback(hObject, eventdata, handles)
%-----
%RECOLHE DADOS E GUARDA COMO VECTOR A E I J G
%-----
A=evalin('base', 'A');
A_unit = evalin('base', 'A_unit');
%Recolho os dados das popup unidades
%-----
```

```

switch get(handles.A_units, 'Value')
case 1 %mm^2
    A_unit_a=10^(-6);
case 2 %cm^2
    A_unit_a=10^(-4);
case 3 %m^2
    A_unit_a=1;
otherwise
end
Ne=get(handles.Elements_Pop, 'Value'); %Recolho os dados do popup
elemento
A(Ne)=((str2num(get(handles.A_txt, 'String')))*A_unit_a);
A_unit(Ne)=A_unit_a;
A_sel = evalin('base', 'A_sel');
A_sel(Ne)=get(handles.A_units, 'Value');
assignin('base', 'A', A);
assignin('base', 'A_sel', A_sel);
assignin('base', 'A_unit', A_unit);

```

Colocar a barra de ferramentas dos axes no topo da janela.

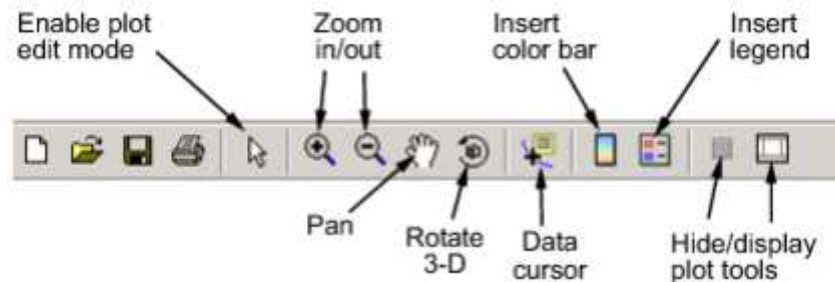
Deve ser introduzido na secção que executa as rotinas antes de mostrar a janela.

Ex:

```

%-----
%COMANDO PARA COLOCAR BARRA DE COMANDOS DE AXES
%-----
set(hObject, 'toolbar', 'figure');

```



Salvar todas as variáveis do ambiente de trabalho base num ficheiro *.mat

Esta instrução é só útil se as variáveis de trabalho são exportadas para o ambiente de trabalho. Assim consegue-se guardar estas variáveis para uma nova utilização em uma nova sessão. Este código pode ser executado quer na linha de comando quer em um botão da GUI.

Ex:

```

evalin('base', 'uisave');

```

Recuperar todas as variáveis do ambiente de trabalho base dum ficheiro *.mat

Pode-se fazer duplo Click sobre o ficheiro pretendido no painel à esquerda do ambiente Matlab®. Mas quando é necessário um botão da GUI realizar o Load das variáveis e importar para o ambiente base...

Ex:

```
evalin('base', 'uiopen');
```

Popular uma UI-Table com resultados

Para este procedimento, deve-se alimentar a UI-Table com matrizes numéricas, ou com *Cell* ou *Cell-Arrays*. No código também está presente a atribuição de nomes dos cabeçalhos, assim como a informação para ajustar os espaçamentos (no entanto no exemplo encontra-se desativada).

Ex:

```
ReacDat = evalin('base', 'ReacDat');
set(handles.RESULTS, 'Data', ReacDat);
set(handles.RESULTS, 'ColumnName', {'Node', 'Constraints', 'Value',
'Units'});
%set(handles.RESULTS, 'ColumnWidth', {'100' 'auto' 'auto' 'auto'});
set(handles.RESULTS, 'RowName', []);
```

Regras de Cell-Array

Há poucas diferenças, entre uma matriz e uma *Cell-Array*. Em aparência são o mesmo, a diferença está no tipo de informação que se pode guardar em cada tipo de elemento. Uma célula (*Cell*) permite a introdução de números e palavras (*strings*). Uma *Cell-Array* é um conjunto de células. Essencialmente é uma macro-celula. Veja o seguinte exemplo:

Matriz	Cell ou Cell-Array
$D(3 \times 3) \rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & 4 \\ 3 & 4 & 1 \end{bmatrix}$	$E\{3 \times 3\} \rightarrow \begin{bmatrix} [1] & [2] & [3] \\ [2] & [olá] & [4] \\ [3] & [4] & [1] \end{bmatrix}$
$D(2,3)=4$	$E[2,1]=2$

A vantagem de uma *Cell-Array* é que pode ser composta por mais do que um tipo de informação.

$A=(3;4;-2) \rightarrow A=\begin{pmatrix} 3 \\ 2 \\ -2 \end{pmatrix}$	Cria um vetor 3x1
$B=\{'A', 'B', 'C'\} \rightarrow B=[A] \quad [B] \quad [C]$	Cria uma célula 1x3 de informação <i>string</i>
$C=[A; B] \rightarrow C=\begin{bmatrix} 3 \times 1 \\ 1 \times 3 \end{bmatrix}$	Cria um <i>Cell-Array</i> 2x1 composto pelo vetor e a célula. No entanto, a informação que é mostrada é a informação contida em cada campo.
$C[2,1]=[A] \quad [B] \quad [C]$	Mostra o que está contido na posição 2x1 da <i>Cell-Array</i> .

Notar à utilização de parêntesis curvos, retos ou chavetas. Também deve-se salientar que no caso de uma *Cell-Array* que é exportada para um ficheiro Excel®, o resultado não é a informação do conteúdo, mas sim o conteúdo. A utilidade principal de uma *Cell-Array* é a possibilidade de guardar em um único ficheiro (variável) todo tipo de informação e um grande número de variáveis.

Existe um comando muito útil para estes casos:

$XX=\begin{pmatrix} -3 \\ 5 \\ 7 \end{pmatrix}$	$YY=num2cell(XX)$	$\begin{bmatrix} -3 \\ 5 \\ 7 \end{bmatrix}$
---	-------------------	--

Criar ficheiro Excel® com informação matricial ou célula

Após a criação dos dados a salvar como XLS ou XLSX. Isto abre uma janela do tipo Save e guarda o ficheiro. NOTA: também é colocado o nome do ficheiro na célula E1 da primeira página.

Ex:

```
%ESCRITA DO FICHEIRO
[filename, pathname] = uiputfile('*.xlsx', 'Salvar relatório como');
if isequal(filename,0) || isequal(pathname,0)
%Não faz nada (cancelou)
else
    outname = fullfile(pathname, filename)
    %primeira pagina estrutura
    xlswrite(outname,NODE_INI,1,'A1');
    xlswrite(outname,{filename},1,'E1');
```

```
end
```

Para guardar em outra página:

```
%terceira pagina ações
xlswrite(outname,AC_INI,3,'A1');
```

NOTA: de maneira a reduzir o tempo na criação do ficheiro Excel®, é recomendável compilar toda a data de uma página em uma variável *Cell-Array*, pois de cada vez que é executado o comando `xlswrite`, o Matlab® executa os seguintes processos:

- Abre em segundo plano o programa Excel.
- Abre o ficheiro a escrever.
- Copia a informação.
- Fecha o Excel® e grava o ficheiro.

Importar ficheiro Excel® com informação matricial ou célula

Ex:

```
[filename, pathname] = uigetfile({'*.xlsx'; '*.xls'}, 'File Selector');
if isequal(filename,0) || isequal(pathname,0)
%Não faz nada (cancelou)
else
    outname = fullfile(pathname, filename)
    M=importdata(outname)
    teste=M.data.NODES %0 nome da primeira pagina é NODE
    assignin('base','teste',teste);
end
```

Abre uma janela de procura de ficheiro com a predefinição da seleção XLS e XLSX. Após importar data do Excel®, muita data vem com campos não numéricos: (nota: a função vai importar CAMPOS NUMERICO. Por tanto reduz as células a aquilo que é importante).

teste =

NaN	4	NaN
NaN	NaN	NaN
NaN	NaN	NaN
NaN	NaN	NaN
0	0	0
0	0	-3
3	0	0
0	-4	0

Tratando só a primeira coluna...

```
>> a=teste(:,1)
```

a =

```
NaN
NaN
NaN
NaN
0
0
3
0
```

>> a(isfinite(a(:, 1)), :)

ans =

```
0
0
3
0
```

Alterar o nome das abas de um ficheiro Excel[®] existente

Esta rotina pode ser executada logo após estar salvo o ficheiro e dentro do **else** da rotina para salvar o ficheiro.

Ex:

```
e = actxserver('Excel.Application');
ewb = e.Workbooks.Open(outname);
ewb.Worksheets.Item(1).Name = 'NODES';
ewb.Worksheets.Item(2).Name = 'PROPERTIES';
ewb.Worksheets.Item(3).Name = 'ACTIONS';
ewb.Worksheets.Item(4).Name = 'SOLUTIONS';
ewb.Save
ewb.Close(false)
```