

Grupo 56	100137	Carlota Barros
	100163	Duarte Morais
	100202	João Dias

MECÂNICA COMPUTACIONAL – Engenharia Mecânica e Aeroespacial

Trabalho Computacional – Ano lectivo 2022/2023

Enunciado 56

Este trabalho tem como objectivo avaliar a capacidade de implementar computacionalmente um programa de elementos finitos em Matlab, para a análise linear de problemas planos (escalares neste caso). Assim, pretende-se que seja construído um modelo de elementos finitos para um dado problema e que seja revolido, quer pelo programa implementado, quer por software comercial de elementos finitos existente (NX, ANSYS, Abaqus, etc.). A malha base para esta análise deve ter entre 100 a 200 graus de liberdade.

O programa desenvolvido deve ler toda a informação referente à malha de elementos finitos através de um ficheiro de dados com uma formatação pré-definida (ver página da disciplina no fenix), e produzir um ficheiro com resultados que sejam relevantes. O programa desenvolvido deve permitir a aplicação correta das condições de fronteira deste enunciado. Além disso o programa deve poder utilizar, no mínimo, malhas de elementos finitos com os elementos indicados na ficha em anexo (OBRIGATÓRIO).

O código deve ser numérico e não deve usar cálculo simbólico.

Problema a resolver: **Condução de Calor**

Pretende analisar-se o problema de condução de calor num corpo plano como representado na figura.

O corpo é constituído por um ou mais materiais isotrópicos de condutividade térmica k conhecida. Admita que não há perda de calor nas direcções perpendiculares ao plano do corpo e que a espessura do corpo é unitária. Pretende determinar-se a distribuição de temperaturas no sólido em regime estacionário, bem como o fluxo de calor para a situação representada. Respeite as condições de fronteira especificadas no enunciado.

Considere os seguintes aspectos no seu relatório:

- Apresente a equação diferencial que rege o fenómeno físico que quer estudar e indique claramente as condições de fronteira empregues e todas as aproximações que fizer.
- Teste e valide o programa desenvolvido para um problema do mesmo tipo com uma geometria simples para o qual exista solução analítica. Neste caso simples, estude os resultados para a temperatura, o fluxo de calor e teste os diferentes tipos de condição de fronteira a usar no trabalho. Apresente a solução analítica e calcule os erros absolutos para a temperatura e para o fluxo de calor.
- Estabeleça a malha de elementos finitos base, resolva-a com o programa desenvolvido e com o software comercial. Compare as soluções de uma forma detalhada e crítica de modo a validar o código em Matlab face ao software comercial.
- Faça uma análise de convergência da solução, refinando uniformemente a malha inicial 4/5 vezes utilizando apenas o software comercial. Cada elemento deve ser subdividido em 4 para se passar para o nível de refinamento seguinte. Selecione um conjunto de pontos de interesse

(5 a 10) que devem estar presentes em todas as malhas e investigue se há ou não há convergência para a temperatura e para o fluxo de calor.

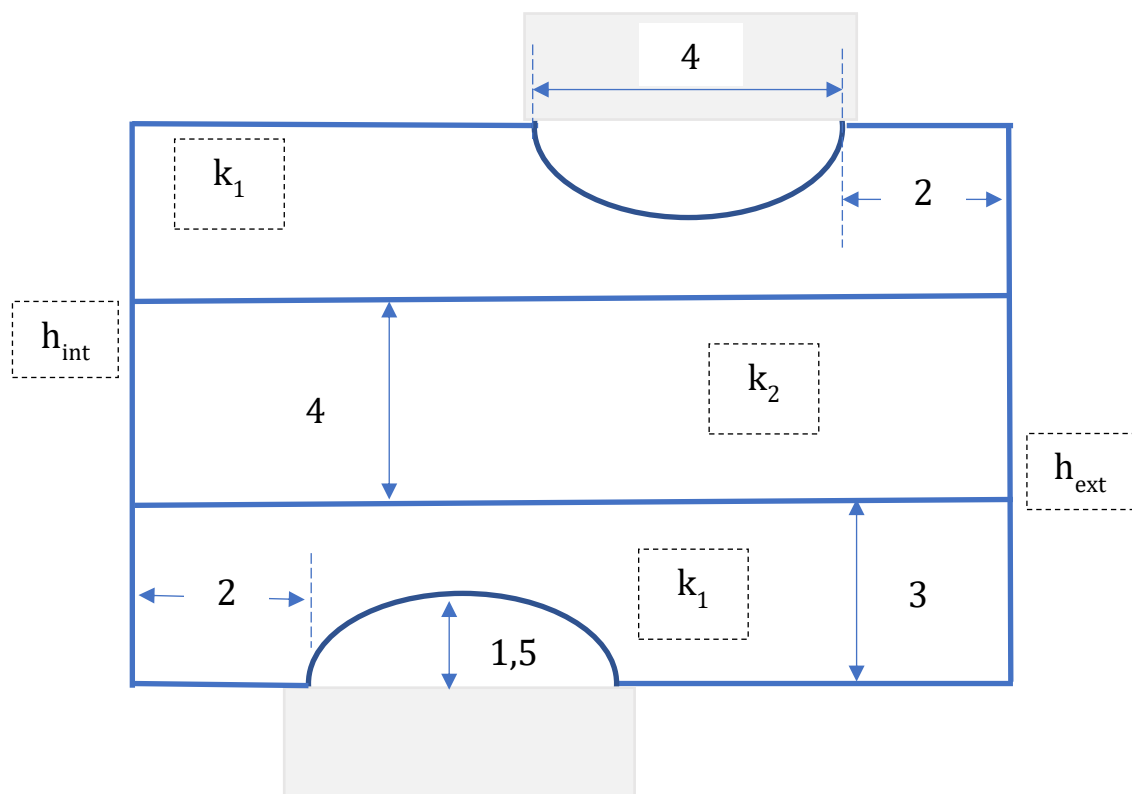
- Para a última malha obtida (refinada), analise detalhadamente os resultados obtidos. Compare o comportamento dos elementos testados.
- Represente graficamente a distribuição de temperatura e do fluxo de calor, identificando os aspectos que considerar relevantes.
- Calcule a potência calorífica total que atravessa a fronteira discriminando os troços relevantes.

Elabore um relatório **ORIGINAL** a descrever o programa, o modelo construído, bem como uma análise dos resultados, para as diferentes opções de matriz rigidez escolhida. Apresente o relatório de acordo com as normas disponibilizadas na página do fenix. Relatórios que não sigam as normas estipuladas não serão avaliados.

IMAGEM E DIMENSÕES

Enunciado 56

Condução de Calor



Análise com elementos finitos quadrangulares de 4 e 8 nós (Q4 e Q8)
Altura, $L_1 = 0,1 \text{ m}$

Comprimento total, $L_2 = 0,11 \text{ m}$

Temperatura na parede do entalhe superior: $175 \text{ }^\circ\text{C}$

Temperatura na parede do entalhe inferior: $425 \text{ }^\circ\text{C}$

Condutividade térmica do material isolante, $k_1 = 0,06 \text{ W/m K}$

Condutividade térmica do material estrutural, $k_2 = 2,1 \text{ W/m K}$

Coeficiente de convecção interna $h_{\text{int}} = 120 \text{ W/m}^2\text{K}$

Coeficiente de convecção externa $h_{\text{ext}} = 30 \text{ W/m}^2\text{K}$

Temperatura do vapor interior, $T_{\text{int}} = 250 \text{ }^\circ\text{C}$

Temperatura do ar exterior, $T_{\text{ext}} = 25 \text{ }^\circ\text{C}$

Restantes medidas em cm na figura



INSTITUTO SUPERIOR TÉCNICO
LEMEC

MECÂNICA COMPUTACIONAL
2022/2023

TRABALHO COMPUTACIONAL

PROFESSOR ALEXANDRE CORREIA

GRUPO 56:

Carlota Barros nº 100137

Duarte Morais nº 100163

João Dias nº 100202

Índice

1	Introdução	2
2	Considerações Teóricas	2
2.1	Condução de Calor	3
2.2	Equação Diferencial	4
2.3	Condições de Fronteira	5
2.4	Formulação Forte	6
2.4.1	Formulação Fraca	6
2.5	Matriz de Rigidez e Vetor de Cargas	7
2.6	Malhas de Elementos Finitos 2D	8
2.6.1	Malhas Retangulares	8
3	Implementação em MATLAB	11
3.1	Objetivos	11
3.2	Descrição de funções	11
3.2.1	Funções principais	11
3.2.2	Funções secundárias	12
3.3	Solução analítica	12
4	Siemens NX	14
4.1	Análise de convergência	14
4.2	Refinamento da malha	17
4.2.1	Malha de QUAD-4	18
4.2.2	Malha de QUAD-8	18
5	Comparação de Resultados entre MATLAB e NX	19
5.1	Malha de QUAD-4	19
5.2	Malha de QUAD-8	20
5.3	Interpretação dos resultados	21
6	Determinação da Potência Calorífica	22
7	Conclusões	23
8	Anexo	24
9	Referências	29

1 Introdução

Este trabalho computacional visa colocar em prática conhecimentos adquiridos na unidade curricular de Mecânica Computacional e, particularmente, analisar o problema de condução de calor num corpo plano, através da aplicação do Método dos Elementos Finitos (MEF).

Análises deste tipo têm em conta diversas variáveis, entre elas o material e geometria do corpo analisado, o que introduz alguma complexidade nestes problemas, impossibilitando-se a sua compreensão através de uma simples abordagem analítica. Surge assim a necessidade de recorrer a programas computacionais como o Siemens NX e o MATLAB, de forma a que se possa construir um modelo de elementos finitos, neste caso para um problema 2D com incógnita escalar, que se possa resolver tanto pelo software comercial de elementos finitos utilizado, como pelo programa implementado.

Desta forma, procurando-se a validação do código desenvolvido, compararam-se as soluções analíticas que resultam da implementação computacional de um programa de elementos finitos em MATLAB com as várias soluções de referência, obtidas nas múltiplas simulações no programa Siemens NX.

2 Considerações Teóricas

Para formular um modelo de elementos finitos é necessário passar por diferentes fases, sendo elas:

- A discretização por elementos finitos do domínio, uma fase que nos permite obter uma malha, composta pelo contorno original, por pontos nodais e, no caso da figura apresentada, por elementos finitos triangulares ;

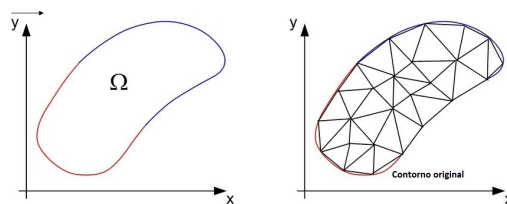


Figura 1: Discretização por elementos finitos do domínio

- A obtenção de funções aproximadas para cada um desses elementos, de acordo com a dimensão e tipo do problema ;
- Montagem dos elementos, tendo em conta a continuidade da solução;

Note-se que, ainda que este seja um método aproximado, a convergência do MEF permite-nos afirmar que em malhas consistentes, à medida que o tamanho dos elementos finitos tende para zero, a quantidade de nós tende consequentemente para infinito e a solução obtida converge para a solução exata do problema.

Assim, prova-se que quanto maior for o número de elementos numa malha, mais rigorosa será a análise obtida, pelo que a precisão do MEF depende não só da quantidade de nós e elementos na malha, mas também do tamanho e tipo desses elementos.

2.1 Condução de Calor

Considere-se um corpo bidimensional num sistema de coordenadas cartesianas. $Q(x, y)$ é a taxa de calor por unidade de volume e tempo e $q_x(x, y)$ e $q_y(x, y)$ são as componentes do vetor fluxo de calor num ponto genérico (x, y) de um domínio Ω .

Nestas condições, a equação que rege o problema de condução de calor pode ser deduzida considerando-se um elemento diferencial de lados dx e dy e um fluxo de calor que atravesse um contorno infinitesimal.

Admitindo-se uma espessura unitária para o corpo em análise, a taxa de calor inerente é $Qdxdy$ e, não havendo perda de calor na direção dos contornos, tem-se:

$$Qdxdy + q_x dy + q_y dx = (q_x + \frac{\partial q_x}{\partial x} dx) dy + (q_y + \frac{\partial q_y}{\partial y} dy) dx \quad (1)$$

Partindo-se da equação anterior é possível obter a solução do problema para regime estacionário,

$$-\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} + Q = 0 \quad (2)$$

que também se pode representar da forma,

$$-\nabla^T q + Q = 0 \text{ em } \Omega \quad (3)$$

onde

$$\nabla^T q = \text{div } q \quad (4)$$

A Lei de Fourier diz-nos que, para um fluxo unidimensional, se verifica proporcionalidade entre o fluxo de calor numa direção e a taxa de variação de temperatura u nessa mesma direção, pelo que se tem

$$q_x = -k_x \frac{\partial u}{\partial x} , \quad (5)$$

onde k_x é a condutividade térmica do material na direção longitudinal.

Generalizando para o caso bidimensional tem-se:

$$q(x, y) = -k \nabla u , \quad (6)$$

cujas matriz de conectividade é dada por

$$k = k(x, y) = \begin{bmatrix} k_{xx}(x, y) & k_{xy}(x, y) \\ k_{xy}(x, y) & k_{yy}(x, y) \end{bmatrix} \quad (7)$$

e

$$\nabla u = \left\{ \frac{\partial}{\partial x} \right\} u = \left\{ \frac{\partial u}{\partial x} \right\} = \text{grad } u \equiv \text{grad } T. \quad (8)$$

Desta forma conclui-se que,

$$\begin{aligned} q_x &= -\left(k_{xx}\frac{\partial u}{\partial x} + k_{xy}\frac{\partial u}{\partial y}\right) \\ q_y &= -\left(k_{xy}\frac{\partial u}{\partial x} + k_{yy}\frac{\partial u}{\partial y}\right) \end{aligned} \quad (9)$$

Relacionando este resultado com as equações (2) e (9) obtém-se:

$$\frac{\partial}{\partial x}\left(k_{xx}\frac{\partial u}{\partial x} + k_{xy}\frac{\partial u}{\partial y}\right) + \frac{\partial}{\partial y}\left(k_{xy}\frac{\partial u}{\partial x} + k_{yy}\frac{\partial u}{\partial y}\right) + Q = 0 \quad (10)$$

Para o caso em que (x, y) coincidem com as direções principais do material, $k_{xy} = 0$ e considerando-se um meio isotrópico, $k_{xx} = k_{yy} = k$, pelo que a matriz de condutividade térmica é dada por:

$$k = k(x, y) = \begin{bmatrix} k(x, y) & 0 \\ 0 & k(x, y) \end{bmatrix} = k(x, y) I \quad (11)$$

Em meios homogêneos k_{xx} e k_{yy} são constantes e, mais especificamente, num meio isentrópico $k_{xy} = 0$ e $k_{xx} = k_{yy} = k = cte$. Desta forma o resultado obtido anteriormente pode ser simplificado, obtendo-se a Equação de Poisson, uma equação que nos permite estudar diversos fenómenos em engenharia:

$$k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right) + Q = 0 \quad (12)$$

Não havendo nenhuma fonte de calor interna, $Q = 0$, tem-se apenas:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad (13)$$

Chega-se assim à Equação de Laplace.

2.2 Equação Diferencial

Aplicando o princípio da conservação de energia, podemos afirmar que as equações diferenciais gerais que descrevem o problema da condução de calor, estudado num domínio de 2 dimensões, em regime transiente e estacionário, respetivamente, são:

$$-\nabla \cdot (k\nabla u) + c.u = f \quad (14)$$

e

$$\nabla \cdot (k\nabla u) + f = 0 \quad (15)$$

onde k , b e f são funções contínuas em (x, y) que representam a condutividade térmica, a convecção e a geração de calor, respetivamente.

Já a relação entre o vetor fluxo de calor e o gradiente de temperatura é dada pela Lei de Fourier, que nos descreve o comportamento do material da seguinte forma:

$$q(x, y) = -\nabla \cdot (k \nabla u) \quad (16)$$

sendo q o vetor fluxo e u a temperatura.

Considere-se um domínio infinitesimal Ω com fronteira designada Γ_Ω , tal como representado na figura (2),

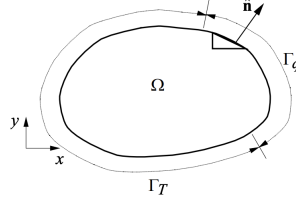


Figura 2: Fluxo de calor que atravessa um elemento infinitesimal

Pode-se então calcular o fluxo total de calor que atravessa a fronteira do domínio da seguinte forma:

$$\sum = \int_{\Gamma_\Omega} q_n ds = \int_{\Gamma_\Omega} \vec{q} \cdot \vec{n} ds = \int_{\Omega} \text{div} \vec{q} \cdot dxdy = \int_{\Omega} \nabla \cdot \vec{q} \cdot dxdy \quad (17)$$

2.3 Condições de Fronteira

As condições de fronteira podem ser de vários tipos.

- Condições de fronteira essenciais: Correspondem a fluxos impostos. Desta forma, sendo o valor da temperatura u especificado numa porção da fronteira Γ_{Ω_1} , a condição de fronteira é do tipo de Dirichlet. Visto que se estudam várias malhas neste problema, as condições de fronteira nos nós dependem do numero de nós em cada malha, pelo que se podem escrever as seguintes relações:

$$\hat{u} = u \Rightarrow \hat{u}_x = u_x, \hat{u}_y = u_y \quad (18)$$

- Condições de fronteira naturais: Na outra porção da fronteira, as condições de fronteira podem assumir ainda outras 2 formas:

1. Condições de Neumann (fluxo normal especificado):

$$q_n(s) = -k \frac{\partial u}{\partial n} = \hat{q}(s), \forall s \in \Gamma_{\Omega_2} \quad (19)$$

2. Condições de Robin, sendo $p(s)$ um coeficiente de convecção para o exterior:

$$q_n(s) = -k \frac{\partial u}{\partial n} = p(s)[u(s) - \hat{u}(s)], \forall s \in \Gamma_{\Omega_2} \quad (20)$$

2.4 Formulação Forte

Tal como já foi demonstrado anteriormente, ao substituir-se a equação constitutiva $q(x, y) = -k.u(x, y)$ no princípio da conservação de energia $\text{div } q = f$, obtém-se $-k.u(x, y) = f$.

Desta forma, é-nos possível obter a formulação forte

$$-k.u(x, y) = f \text{ em } \Omega \quad (21)$$

cujas condições de fronteira (essenciais e naturais) são respetivamente representadas na Figura 3 por Γ_{Ω_1} e Γ_{Ω_2} .

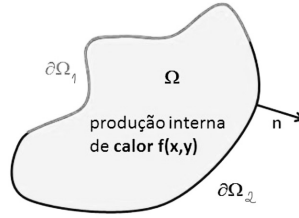


Figura 3: Domínio Ω

2.4.1 Formulação Fraca

Pretendendo-se chegar à formulação fraca do problema da condução de calor, obtém-se o resíduo $R(x, y)$ dado por:

$$R(x, y) = -k.u(x, y) - f = 0 \quad (22)$$

que se multiplica por uma qualquer função de teste v e que se integra em todo o domínio, obtendo-se

$$\int_{\Omega} R(x, y).v \, dxdy = \int_{\Omega} ((-k.u(x, y)) - f).v \, dxdy = - \int_{\Omega} v(k.u(x, y)) \, dxdy - \int_{\Omega} f v \, dxdy = 0 \quad (23)$$

Esta expressão pode ser desenvolvida ao longo de uma sucessão de passos até que se obtenha

$$\int_{\Omega} k.u(x, y).v(x, y) \, dxdy = \int_{\Omega} f v \, dxdy + \int_{\Gamma_{\Omega_2}} \hat{q} v \, ds \quad (24)$$

onde v é um campo de temperaturas virtuais, isto é, uma variação do campo de temperaturas real u . Escrevendo de outro modo, temos:

$$\int_{\Omega} k \left(\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y} \right) \, dxdy = \int_{\Omega} f v \, dxdy + \int_{\Gamma_{\Omega_2}} \hat{q} v \, ds \quad (25)$$

ou ainda,

$$\int_{\Omega} (k \nabla u \cdot \nabla v + c \cdot u \cdot v) d\Omega + \int_{\Gamma_{\Omega_2}} p v [u - \hat{u}_{\infty}] ds = \int_{\Omega} f v d\Omega \quad (26)$$

Para que se obtenha a formulação de um problema de elementos finitos, há que substituir o domínio Ω por uma aproximação Ω_h , que não é mais do que uma porção de E elementos finitos com N nós, resultantes da discretização desse mesmo domínio.

Escolhendo um qualquer elemento finito arbitrário Ω_e , podemos definir várias funções aproximadoras contínuas $\Phi_i (i=1, \dots, N)$ dentro desse mesmo elemento. Pelo método de Galerkin tem-se:

$$\begin{aligned} v_h(x, y) &= \sum_{i=1}^N v_i \Psi_i(x, y) \\ u_h(x, y) &= \sum_{i=1}^N u_i \Psi_i(x, y) \end{aligned} \quad (27)$$

Aplicando v_h e u_h na formulação fraca obtida anteriormente, chega-se às equações definidas por:

$$\sum_{j=1}^N K_{ij} T_j = F_i \text{ para } (i = 1, \dots, N) \quad (28)$$

onde K_{ij} representa os coeficientes da matriz de rigidez, para o problema de condução de calor.

2.5 Matriz de Rigidez e Vetor de Cargas

Partindo do sistema de equações algébricas lineares apresentado na secção anterior, tem-se que a matriz de rigidez K representa a condutividade térmica dos materiais e calcula-se da seguinte forma:

$$K_{ij} = \int_{\Omega_h} \left[k \left(\frac{\partial \Phi_i}{\partial x} \frac{\partial \Phi_j}{\partial x} + \frac{\partial \Phi_i}{\partial y} \frac{\partial \Phi_j}{\partial y} \right) + c \Phi_i \Phi_j \right] + \int_{\Gamma_{\Omega_{2h}}} p \Phi_i \Phi_j ds \quad (29)$$

F_i são as componentes do "vetor de forças", também no caso concreto da condução de calor, e são dadas por:

$$F_i = \int_{\Omega_h} f \Phi_i d\Omega + \int_{\Gamma_{\Omega_{2h}}} \gamma \Phi_i ds \quad (30)$$

com $\gamma \equiv \hat{q}$ e $\Gamma_{\Omega_{2h}} = \Gamma_{\Omega_{qh}}$.

Alterando-se o sistema de forma a ter em conta as condições de fronteira essenciais, obtém-se os valores nodais u_j que definem a solução aproximada. Para \hat{u} , aplica-se a aproximação

$$\hat{u}_h(s) = \sum_{j=1}^N \hat{u}_j \Phi_j(x(s), y(s)) \quad (31)$$

para que se tenha apenas de especificar \hat{u}_j nos nós de $\Gamma_{\Omega_{1h}}$.

A temperatura num ponto interior de um elemento pode ser obtida por interpolação usando-se as funções de forma Ψ , onde Ψ designa as funções de forma, tal que:

$$u_h^e(x, y) = \sum_{i=1}^{N_e} \Psi_i^e(x, y) u_i \quad (32)$$

Já os fluxos nos pontos interiores dos elementos podem ser obtidos derivando essas funções de forma:

$$q_h^e = -k \nabla u_h^e(x, y) = -k \sum_{i=1}^{N_e} u_i \nabla \Psi_i^e(x, y) \quad (33)$$

Como os fluxos em elementos adjacentes a um mesmo nó são diferentes, existe uma descontinuidade nos nós e o erro do fluxo que os atravessa é maior. De forma a melhorar a precisão dos cálculos num nó arbitrário, o software usado faz uma média ponderada dos valores dos fluxos de todos os elementos que partilham esse nó.

2.6 Malhas de Elementos Finitos 2D

Uma malha de elementos finitos pode ter o formato de triângulos, quadriláteros ou outros.

Para garantir a continuidade da solução obtida, é necessário que dois elementos adjacentes tenham o mesmo tipo de interpolação sobre o lado que lhes é comum, sendo que a qualidade da malha depende da existência de distorções e dos parâmetros dos elementos.

Também o aumento do grau das funções de forma dos elementos pode influenciar positivamente o resultado das simulações. Assim, em vez das funções de forma serem apenas de grau 1, poderão ser de grau 2, 3, ou superior, alcançando-se maior precisão nos resultados, necessitando-se no entanto de um maior poder computacional.

2.6.1 Malhas Retangulares

- **QUAD-4: elementos com 4 nós, lineares**

A temperatura no interior de um qualquer elemento, $u_h^e(x, y)$, é dada por:

$$u_h^e = \sum_{j=1}^4 \Psi_j^e u_j = \Psi_1^e u_1 + \Psi_2^e u_2 + \Psi_3^e u_3 + \Psi_4^e u_4 \quad (34)$$

onde Ψ_1^e , Ψ_2^e e Ψ_3^e são as funções de forma para um elemento quadrangular linear, que se relacionam com as coordenadas dos seus nós e com a área do elemento.

A matriz de rigidez para elementos deste tipo é dada por:

$$[K^e] = k \int_0^b \int_0^a [B]^t [B] dxdy = \frac{k}{6ab} \begin{bmatrix} 2(a^2 + b^2) & a^2 - 2b^2 & -(a^2 + b^2) & b^2 - 2a^2 \\ a^2 - 2b^2 & 2(a^2 + b^2) & b^2 - 2a^2 & -(a^2 + b^2) \\ -(a^2 + b^2) & b^2 - 2a^2 & 2(a^2 + b^2) & a^2 - 2b^2 \\ b^2 - 2a^2 & -(a^2 + b^2) & a^2 - 2b^2 & 2(a^2 + b^2) \end{bmatrix} \quad (35)$$

sendo k a condutividade térmica do material e a e b as dimensões do elemento retangular.

Já o vetor de cargas é dado por:

$$[F^e] = f \frac{ab}{4} \begin{Bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{Bmatrix} \quad (36)$$

O cálculo da matriz de rigidez é feito através do método de integração numérica de Gauss-Legendre, que implica a mudança de variáveis dada por:

$$\begin{cases} x = x(\zeta, \eta) = \sum_{i=1}^4 \Psi_i^e(x, y) x_i \\ y = y(\zeta, \eta) = \sum_{i=1}^4 \Psi_i^e(x, y) y_i \end{cases} \quad (37)$$

como ilustra a figura seguinte

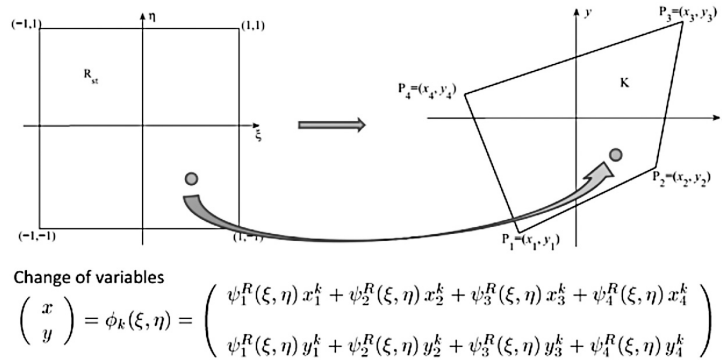


Figura 4: Transformação de coordenadas para um elemento isoparamétrico

A maioria dos elementos finitos de 4 nós considerados têm geometria rectangular com lados paralelos e ortogonais, no entanto, é possível considerar elementos quadriláteros não rectangulares, ao se aplicar o conceito de elementos finitos isoparamétricos através da mudança de variável abordada anteriormente. Desta forma, o integral de área

$$\int_{\Omega_e} f(x, y) dxdy \quad (38)$$

é convertido em

$$\int_{-1}^1 \int_{-1}^1 f(\zeta, \eta) J(\zeta, \eta) d\zeta d\eta \quad (39)$$

onde J representa o determinante da matriz Jacobiana.

H_{ij}^e e P_i^e são coeficientes adicionais que podem ser calculados através de integrais de fronteira. Desta forma, sendo componentes relativas à convecção só se podem aplicar a elementos afetados por condições de fronteira desse tipo. Estes coeficientes são então dados por:

$$[H^e] = \frac{\beta_{12}^e h_{12}^e}{6} \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \frac{\beta_{23}^e h_{23}^e}{6} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \frac{\beta_{34}^e h_{34}^e}{6} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} + \frac{\beta_{41}^e h_{41}^e}{6} \begin{bmatrix} 2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 \end{bmatrix} \quad (40)$$

$$[P^e] = \frac{\beta_{12}^e T_{\infty}^{12} h_{12}^e}{2} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \frac{\beta_{23}^e T_{\infty}^{23} h_{23}^e}{2} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \frac{\beta_{34}^e T_{\infty}^{34} h_{34}^e}{2} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} + \frac{\beta_{41}^e T_{\infty}^{41} h_{41}^e}{2} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (41)$$

onde o fator β^e representa o coeficiente de convecção, que se assume constante.

A matriz H e o vetor P acima descritos dependem do tipo de malha a que são aplicados e de parâmetros como o coeficiente de convecção β^e ou até o distanciamento entre os nós h e a temperatura do meio ambiente, T_{∞} .

Estas componentes foram implementadas computacionalmente em *MATLAB* consoante as seguintes expressões algébricas:

$$\begin{cases} K_{ij}^e = \int_{\Omega} \left(k_x \frac{\partial \Psi_i^e}{\partial x} \frac{\partial \Psi_j^e}{\partial x} + k_y \frac{\partial \Psi_i^e}{\partial y} \frac{\partial \Psi_j^e}{\partial y} \right) dx dy \\ H_{ij}^e = \beta^e \oint_{\Gamma_c} \Psi_i^e \Psi_j^e ds \\ P_i^e = \beta^e \oint_{\Gamma_c} \Psi_i^e T_{\infty} ds \end{cases} \quad (42)$$

- **QUAD-8: elementos com 8 nós, quadráticas**

Para o caso de elementos quadrangulares com 8 nós, o raciocínio é análogo ao do caso anterior, com a diferença de que, neste caso, o polinómio é quadrático em x e y e tem um termo biquadrático em xy , sendo dado por:

$$u_h^e(x, y) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 + a_6xy^2 + a_7x^2y + a_8x^2y^2$$

Relativamente às matrizes de convenção, na implementação do código em *MATLAB*, optou-se por calculá-las de uma forma diferente à mencionada anteriormente. Recorreu-se à interpolação quadrática dos lados dos elementos da fronteira, seguida de uma integração numérica de Gauss-Legendre 1D a 3 pontos (grau 5), tal como pode ser visto em (43)

$$\begin{cases} H_{ij}^e = \sum_{k=1}^3 \beta^e \Psi_i(\xi_k) \Psi_j(\xi_k) \frac{ds}{d\xi}(\xi_k) w_k \\ P_i^e = \sum_{k=1}^3 \beta^e T_\infty \Psi_i(\xi_k) \frac{ds}{d\xi}(\xi_k) w_k \end{cases} \quad (43)$$

onde w_k representa o peso do ponto de integração em questão.

3 Implementação em MATLAB

3.1 Objetivos

O desenvolvimento de um programa deste tipo em *MATLAB* tem como objetivo o de construir um modelo de elementos finitos para o problema de condução de calor que possa, posteriormente, ser comparado com o software comercial Siemens NX. Este programa lê toda a informação referente à malha de elementos finitos presente num ficheiro de dados de formatação pré-definida, e produz um ficheiro com resultados relevantes para, pelo menos, malhas de elementos finitos retangulares de 4 e 8 nós, com graus de liberdade entre nós 100 e 200.

Tanto a estrutura como o funcionamento do programa são explorados em detalhe na subsecção que se segue.

3.2 Descrição de funções

Para visualizar a árvore do programa em *MATLAB*, ver o fluxograma (40) presente no anexo.

3.2.1 Funções principais

main(): A função `main()` é a função principal que chama todas as outras funções do projeto para simular o problema de condução de calor da placa. Ao correr, na consola, será perguntado ao utilizador qual malha pretende executar. Dependendo do número que o utilizador escreva, o programa irá correr a malha correspondente a esse número (1 - Malha quadrangular de 4 nós, simples; 2 - Malha quadrangular de 8 nós, simples; 3- Malha quadrangular de 4 nós; 2 - Malha quadrangular de 8 nós).

ler_ficheiro(): Lê toda a informação do ficheiro `.txt` no formato definido e guarda-a em variáveis que serão utilizadas por outras funções do programa.

escolher_data(): Esta função executa algumas operações na informação lida anteriormente para facilitar os cálculos que serão feitos com estes dados.

CalculosElementares_e_Assemblagem(): A partir da informação lida e trabalhada anteriormente, esta função calcula as matrizes elementares e faz a assemblagem, criando as matrizes globais (B_g e P_g) do respetivo problema, tendo em conta se existe convenção.

calculo_temperatura_fluxo(): Esta função utiliza as matrizes B_g e P_g para calcular a temperatura nodal. É também nesta função que são calculados os vários valores do gradiente da temperatura e fluxo, tal como o ponto de origem de cada vetor.

representacao_grafica(): Função responsável por apresentar uma janela final com três gráficos: um representante da malha utilizada e respetivos nós, um representante da temperatura nodal e outro onde são apresentados os vários vetores de fluxo e respetivos pontos de origem.

resultados(): Após ser tudo calculado, esta função apresenta na consola os valores da temperatura em cada nó e o módulo do gradiente e fluxo, em cada ponto calculado.

3.2.2 Funções secundárias

calc_Ke_1elementoQ(): Função que calcula a matriz de rigidez para um elemento quadrangular de quatro ou oito nós. É chamada na função principal **CalculosElementares_e_Assemblagem()**.

calc_robin_quad(): No caso de existir convenção para um elemento de 8 nós, esta função calcula a matriz simétrica H_e que é utilizada para obter matriz de rigidez elementar e o vetor P_e que contribui para o vetor de forças global. É chamada na função principal **CalculosElementares_e_Assemblagem()**.

3.3 Solução analítica

Pode-se encontrar a solução analítica para este problema, partindo-se de uma geometria simples de uma malha quadrangular, como mostra a figura que se segue.

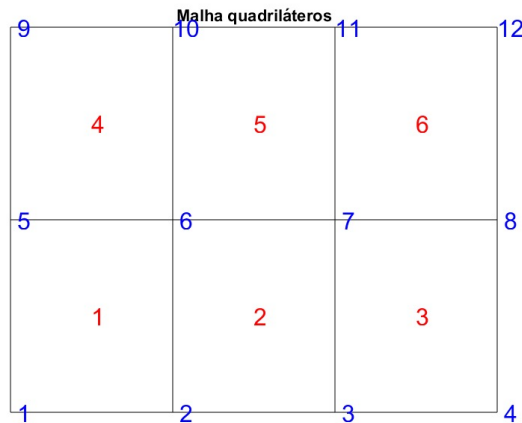


Figura 5: Malha de quadriláteros

Testar o programa para uma malha deste tipo é suficiente, pelo que, assumindo valor unitário para a largura dos quadrados e substituindo valores na equação (35) já estudada, obtém-se a matriz de rigidez:

$$[K^e] = \frac{k}{6} \begin{bmatrix} 4 & -1 & -2 & -1 \\ -1 & 4 & -1 & -2 \\ -2 & -1 & 4 & -1 \\ -1 & -2 & -1 & 4 \end{bmatrix}$$

Sabe-se que o vetor de cargas $[F^e]$ é nulo, a temperatura do ambiente exterior é $T_{amb} = 25^\circ C$ e a condutividade térmica do material extrutural é $k = 2.1 W/mK$. Na ausência de geração de calor ou convecção em regime estacionário, tem-se também que $-k\nabla^2 u = 0$. Sendo que $[K^e][T^e] = [Q^e]$ e que,

$$Q_i^e = \oint_{\Gamma_c} q_n \Psi_i ds \quad (44)$$

as temperaturas em cada nó seriam dadas por U_i e,

$$\begin{cases} U_4 = U_8 = U_{12} = 0 \\ U_9 = T_{amb} \\ U_{10} = \frac{\sqrt{3}}{2} \\ U_{11} = \frac{T_{amb}}{2} \\ F_1 = F_2 = F_3 = F_5 = 0, \text{ fluxo nulo devido ao isolamento} \end{cases}$$

e o balanço de energia interna implica $F_6 = F_7 = 0$. Assim, as incógnitas do problema são: $F_4, F_8, F_9, F_{10}, F_{11}$ e F_{12} e U_1, U_2, U_3, U_5, U_6 e U_7 . Sabe-se também que:

$$\begin{cases} K_{11} = K_{11}^1, K_{12} = K_{12}^1, K_{15} = K_{14}^1 \\ K_{16} = K_{13}^1, K_{22} = K_{22}^1 + K_{11}^2, K_{23} = K_{12}^2, K_{25} = K_{24}^1 \\ K_{26} = K_{23}^1 + K_{14}^2, K_{27} = K_{13}^2 \text{ etc.} \\ F_1 = Q_1^1, F_2 = Q_2^1 + Q_1^2, F_3 = Q_2^2 + Q_1^3, F_4 = Q_2^3 \text{ etc.} \end{cases}$$

Desta forma, obtém-se a seguinte equação na forma matricial:

$$\frac{k}{6} \begin{bmatrix} 4 & -1 & 0 & -1 & -2 & 0 \\ -1 & 8 & -1 & -2 & -2 & -2 \\ 0 & -1 & 8 & 0 & -2 & -2 \\ -1 & -2 & 0 & 8 & -2 & 0 \\ -2 & -2 & -2 & -2 & 16 & -2 \\ 0 & -2 & -2 & 0 & -2 & 16 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_5 \\ U_6 \\ U_7 \end{bmatrix} = \frac{k}{6} \begin{bmatrix} 0 \\ 0 \\ 0 \\ T_{amb} + \sqrt{3}T_{amb} \\ 2T_{amb} + \sqrt{3}T_{amb} + T_{amb} \\ \sqrt{3}T_{amb} + T_{amb} \end{bmatrix} \quad (45)$$

Como pode ser consultado em [1], podem-se calcular as soluções analíticas para este caso da seguinte forma:

$$\begin{cases} U_1 = 0.6128 * T_{amb} = 15.32^\circ C, U_2 = 0.5307 * T_{amb} = 13.27^\circ C, U_3 = 0.3064 * T_{amb} = 7.66^\circ C \\ U_5 = 0.7030 * T_{amb} = 17.58^\circ C, U_6 = 0.6088 * T_{amb} = 15.22^\circ C, U_7 = 0.3515 * T_{amb} = 8.79^\circ C \end{cases}$$

Desta forma, tendo as soluções analíticas e os dados obtidos no programa, é possível obter os erros relativos que se apresentam de seguida. O facto destes valores serem baixos é um forte indicador do bom funcionamento do programa.

Table 1: Análise de resultados

Nº do Nó	Temperaturas nodais					
	1	2	3	5	6	7
Soluções Analíticas	15.32 °C	13.27 °C	7.66 °C	17.58 °C	15.22 °C	8.79 °C
Solução do Software	12.257 °C	10.615 °C	6.128 °C	14.06 °C	12.176 °C	7.03 °C
Erro (%)	20	20	20	20	20	20

Para o problema de geometria simples resolvido por uma a malha quadrática de 8 nós, foi feito o mesmo exemplo da malha linear alterando o número de nós. Os resultados obtidos podem ser consultados no anexo nas figuras (42), (44) e (46).

4 Siemens NX

Para que se possam comparar os resultados da implementação em *MATLAB*, foram realizadas simulações do problema no software *SIEMENS NX*. Procedeu-se então ao refinamento das malhas, de forma a avaliar a convergência dos resultados, tendo cada refinamento aproximadamente 4 vezes mais elementos que o refinamento anterior. Para isso, escolheram-se 6 pontos em zonas críticas da placa, que representam nós comuns a todas as malhas. A partir desses pontos, realiza-se a análise de convergência por meio de comparações entre iterações sucessivas, de forma a analisar o efeito dos constrangimentos aplicados em toda a peça.

Em ambiente de simulação foram geradas no total 22 malhas, sendo que, metade foram malhas lineares de quadriláteros e as restantes foram malhas quadráticas de quadriláteros. Ambos os tipos de malhas sofreram 5 refinamentos. As malhas quadrangulares geradas tinham um total de 151, 656, 2632, 9510, 39603 e 156412 elementos.

4.1 Análise de convergência

De forma a que se pudesse interpretar a evolução dos resultados obtidos, foram seleccionados 6 nós da placa, dispostos tal como se mostra na figura (6) e traçaram-se os gráficos da temperatura em função do número de nós, para cada elemento.

- **Convergência da temperatura**

Apresentam-se então os gráficos da convergência do valor da temperatura para cada um desses nós, onde estão representados simultaneamente os resultados para malhas do tipo QUAD-4 e QUAD-8.

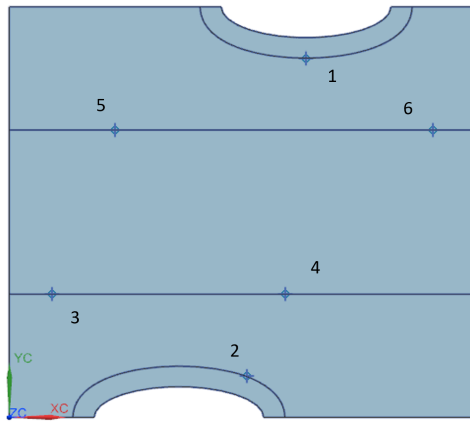


Figura 6: Localização dos nós analisados

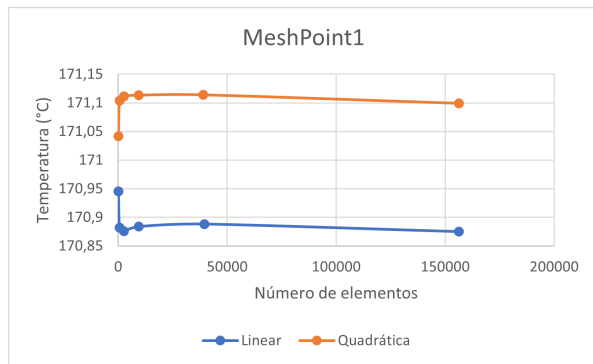


Figura 7: Convergência do 1º Nó

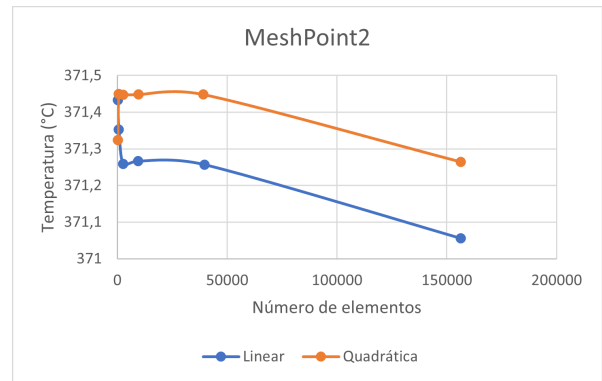


Figura 8: Convergência do 2º Nó

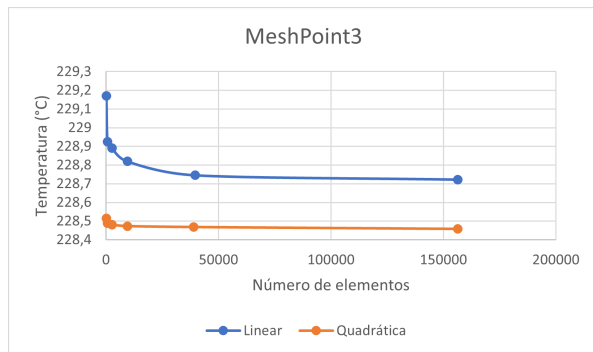


Figura 9: Convergência do 3º Nó

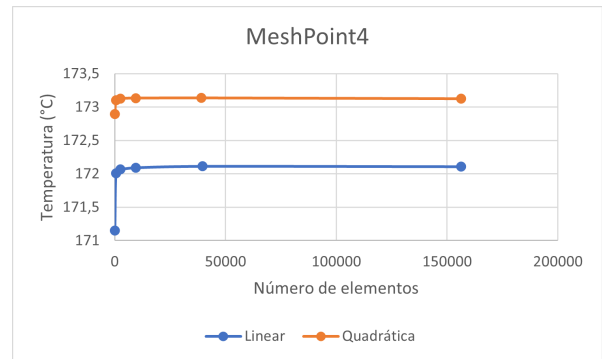


Figura 10: Convergência do 4º Nó

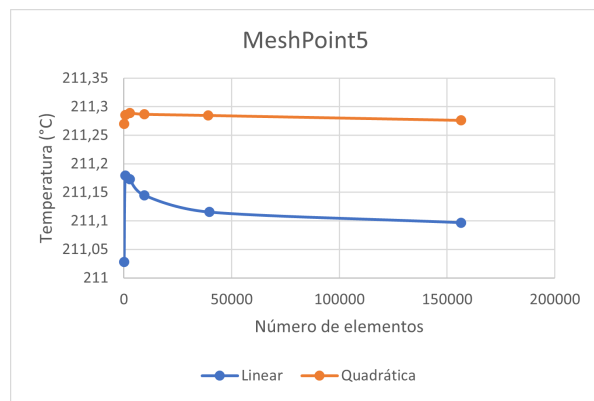


Figura 11: Convergência do 5º Nó

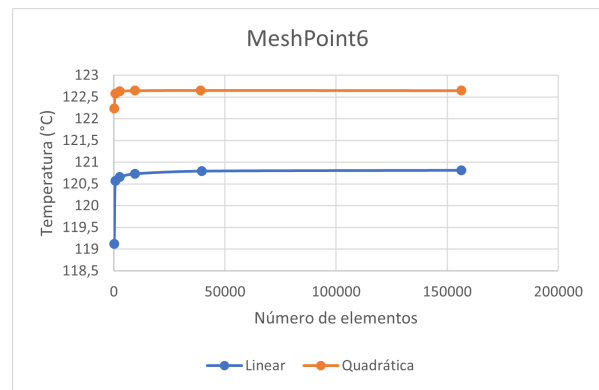


Figura 12: Convergência do 6º Nó

Em primeiro lugar, a partir da análise dos gráficos, é possível concluir que os nós convergem para ambas as malhas, como já seria de esperar. No entanto, é de notar que para o mesmo número de elementos, as malhas de quadriláteros quadráticas convergem mais rapidamente do que as malhas de quadriláteros lineares, já que os seus valores no caso quadrático flutuam entre intervalos mais pequenos. Por esse motivo, as malhas de quadriláteros quadráticas são uma opção mais precisa, mas que em contrapartida requer uma maior carga computacional, na medida em é necessário executar um maior número de operações computacionais.

Posto isto, foi possível compreender que, para obter melhores resultados para o mesmo número de elementos, deve-se utilizar a malha quadrática, mas em certas circunstâncias a malha linear poderá ser a melhor opção. É portanto necessário encontrar sempre um balanço entre o número e tipo de elementos e a precisão que se pretende dos resultados para cada caso em concreto, porque nem sempre se optam pelos fatores da mesma forma.

Outro aspeto a analisar é a convergência do meshpoint2, dado que demora mais tempo para convergir do que qualquer outro ponto de entre os escolhidos. Isso deve-se ao facto de se apresentar numa zona onde irão existir elevadas variações de temperatura, pois a temperatura da parede do entalhe inferior é bastante elevada, afetando assim a convergência desse ponto.

• Convergência do fluxo

Apresentam-se também os gráficos da convergência do fluxo para cada um dos nós, onde estão representados simultaneamente os resultados para malhas do tipo QUAD-4 e QUAD-8.

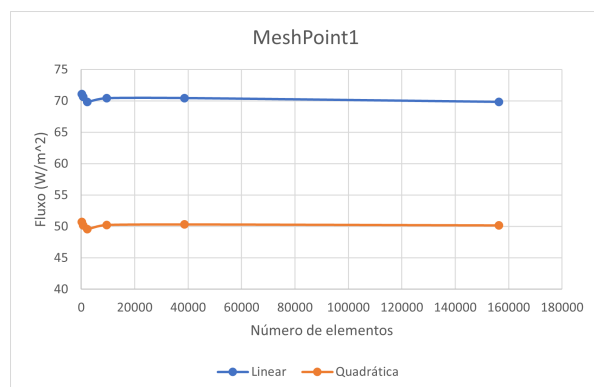


Figura 13: Convergência do 1º Nó

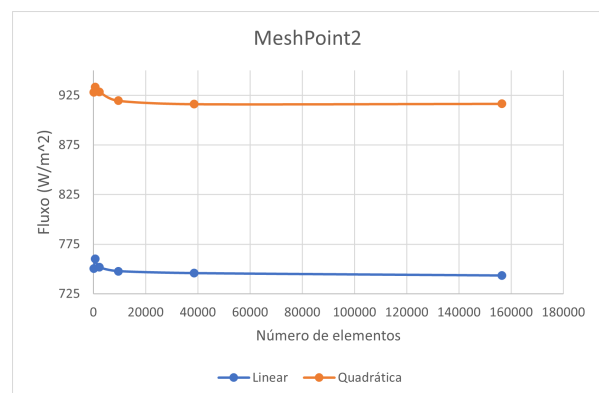


Figura 14: Convergência do 2º Nó

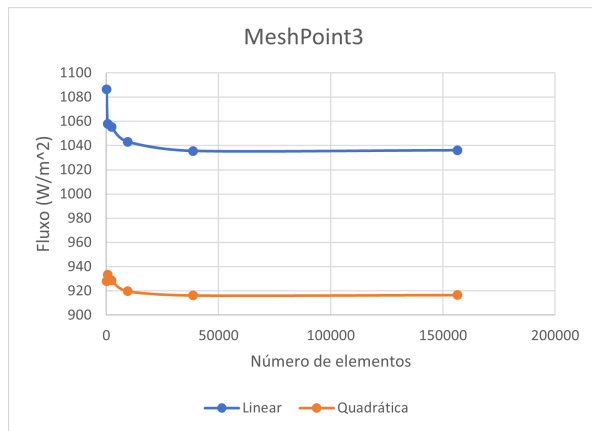


Figura 15: Convergência do 3º Nó

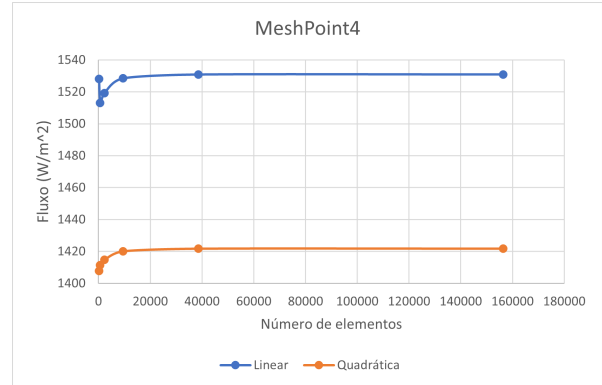


Figura 16: Convergência do 4º Nó

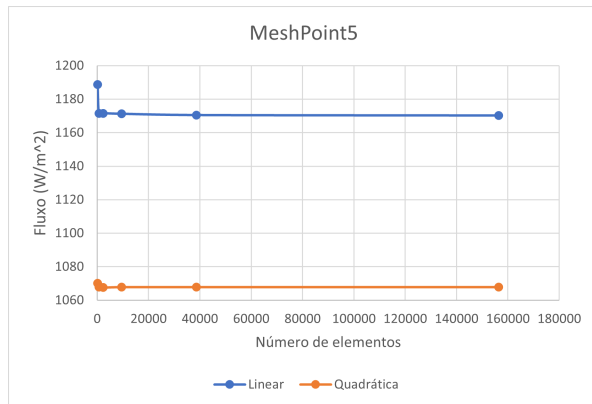


Figura 17: Convergência do 5º Nó

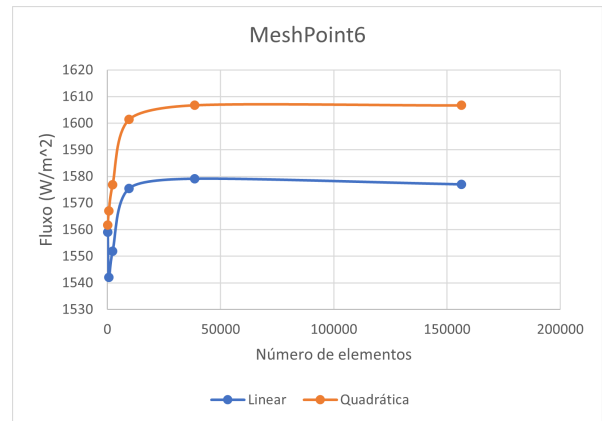


Figura 18: Convergência do 6º Nó

Analisando novamente os gráficos obtidos, conclui-se que existe convergência para ambas as malhas, a nível de fluxo. Ainda que não seja tão evidente como no caso da convergência da temperatura, continua a haver uma convergência mais rápida das malhas quadráticas.

Como os nós são pontos de descontinuidade, o cálculo do fluxo no NX é feito através de uma média aproximada. Este pode ser o motivo para eventuais erros no cálculo do fluxo.

Para além disso, é de notar a ocorrência de uma elevada discrepância de valores entre o ponto para onde a malha quadrática converge e o ponto para onde a malha linear converge. Uma explicação para este acontecimento pode ser o facto de que para elementos quadráticos há o dobro do número de nós por elemento, pelo que o erro que o *software* comercial faz ao calcular os fluxos leva a uma diferença nos resultados obtidos.

4.2 Refinamento da malha

Nesta secção são apresentados os gráficos obtidos para a distribuição de temperaturas e fluxo de calor após 5 refinamentos em NX, de onde se obtiveram 156412 elementos, para malhas lineares e quadráticas.

Os gráficos relativos ao gradiente de temperatura, (47) e (48), podem ser consultados no anexo.

4.2.1 Malha de QUAD-4

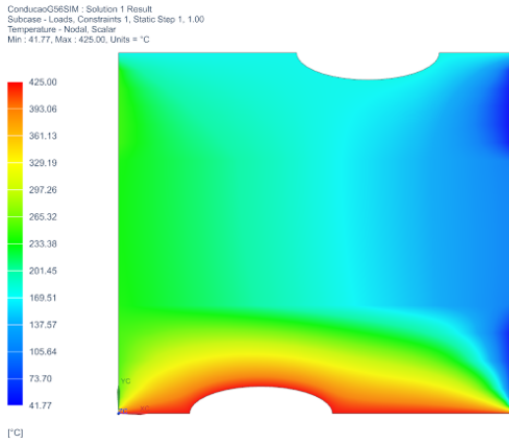


Figura 19: Temperatura nodal [°C] - QUAD-4

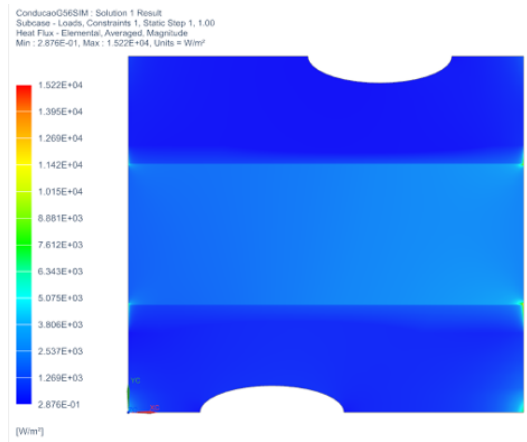


Figura 20: Fluxo de calor [W/m²] - QUAD-4

4.2.2 Malha de QUAD-8

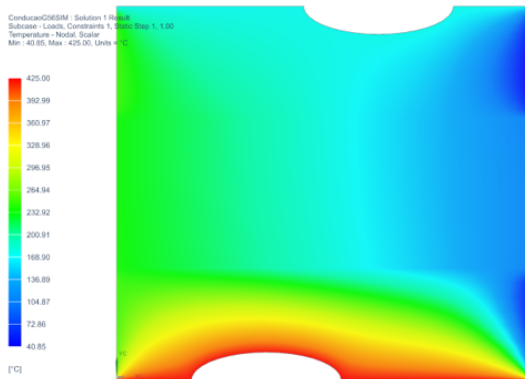


Figura 21: Temperatura nodal [°C] - QUAD-8

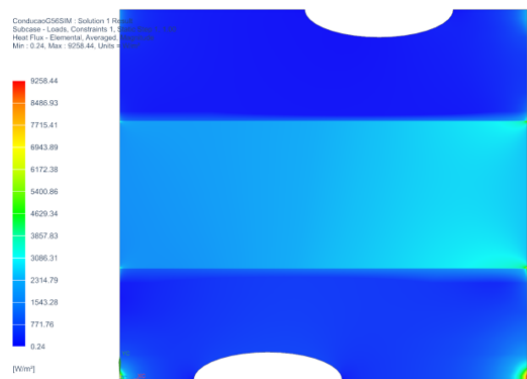


Figura 22: Fluxo de calor [W/m²] - QUAD-8

Por observação dos gráficos relativos aos resultados em NX, tanto para o caso QUAD-4 como para QUAD-8, pode-se concluir que a distribuição de temperatura nodal vai de encontro ao que se esperava, visto que a temperatura no entalhe inferior é muito superior à da restante placa, devido ao constrangimento que lhe é aplicado. Também pelos constrangimentos dados, é expectável que a temperatura na parede interior seja sempre superior à da exterior, algo que se verifica.

Pode-se também concluir que o gradiente de temperatura é maior na região da placa em que se verifica uma maior variação de temperaturas (ponto inferior direito), pelo que também o fluxo será maior nessa região.

Este resultado é apoiado pelas considerações teóricas apresentadas em secções anteriores do relatório, uma vez que o gradiente e o fluxo se relacionam pela Lei de Fourier dada por:

$$q(x, y) = -\nabla \cdot (k \nabla u) \quad (46)$$

Ainda sobre o fluxo, verifica-se também por observação da figura (22), que é maior nas regiões da placa com condutividade térmica superior, um resultado mais uma vez confirmado pela lei enunciada, uma vez que quanto maior for k , maior será o fluxo, q .

Para uma melhor leitura dos valores presentes nos gráficos, sugere-se a consulta do anexo e as figuras (49), (50), (51) e (52).

5 Comparação de Resultados entre MATLAB e NX

Nesta secção comparam-se os gráficos obtidos nas 2 abordagens utilizadas para a resolução deste problema, de forma a que se possam tirar conclusões quanto ao grau de precisão do Método de Elementos Finitos desenvolvido em MATLAB.

5.1 Malha de QUAD-4

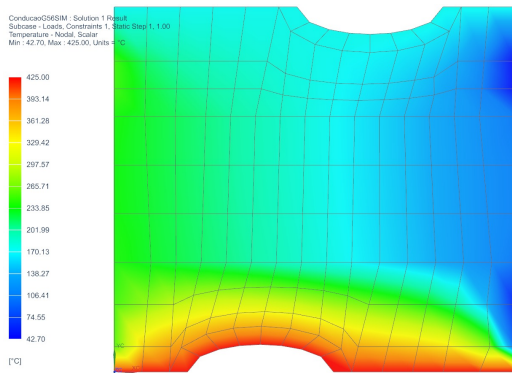


Figura 23: Distribuição de temperatura em NX

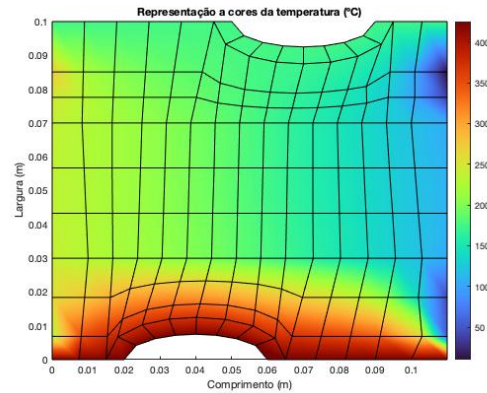


Figura 24: Distribuição de temperatura em MATLAB

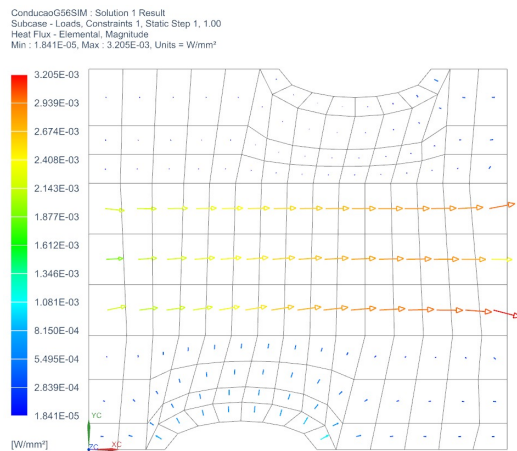


Figura 25: Fluxo de calor em NX

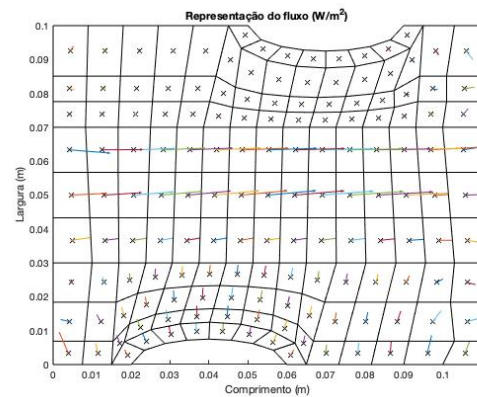


Figura 26: Fluxo de calor em MATLAB

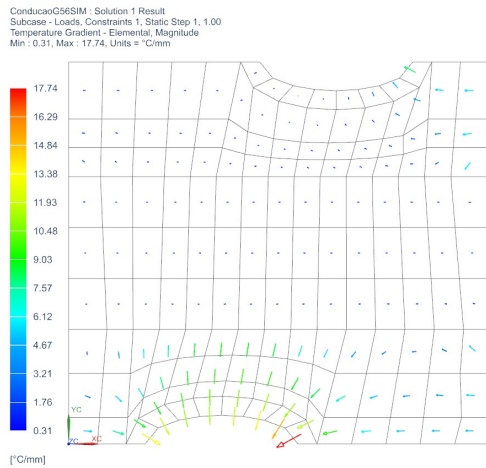


Figura 27: Gradiente de temperatura em NX

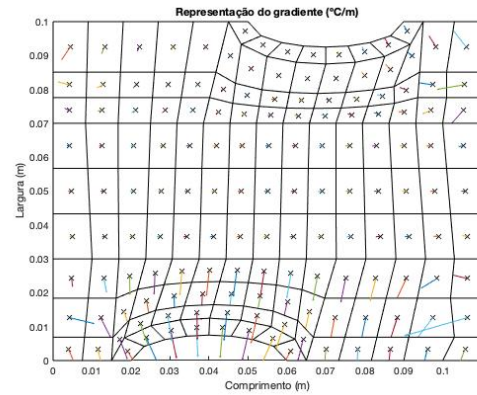


Figura 28: Gradiente de temperatura em MATLAB

5.2 Malha de QUAD-8

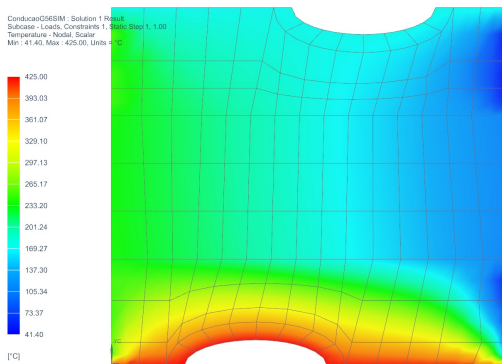


Figura 29: Distribuição de temperatura em NX

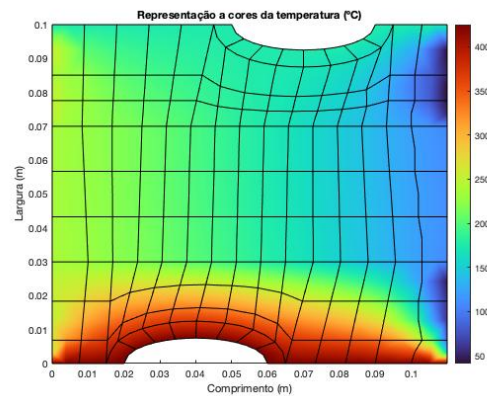


Figura 30: Distribuição de temperatura em MATLAB

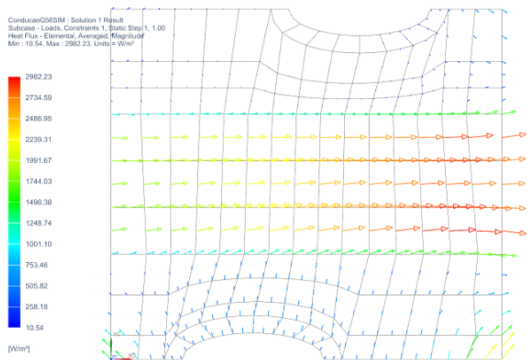


Figura 31: Fluxo de calor em NX

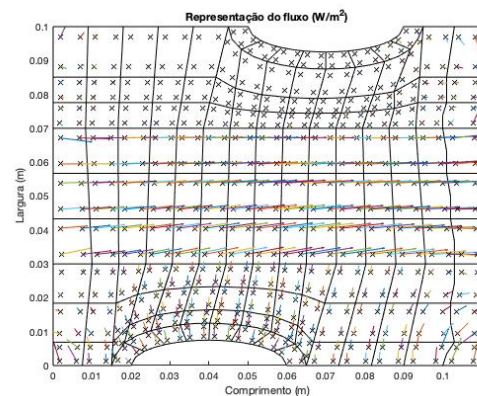


Figura 32: Fluxo de calor em MATLAB

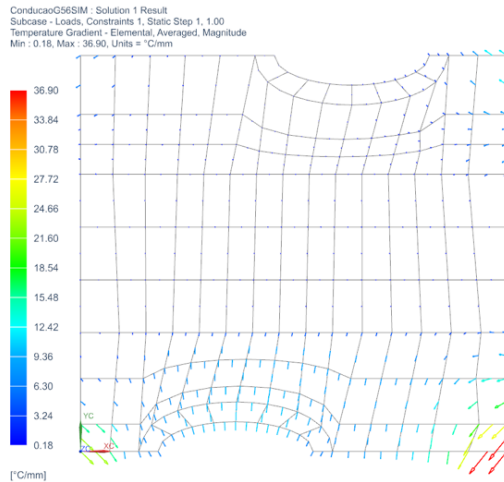


Figura 33: Gradiente de temperatura em NX

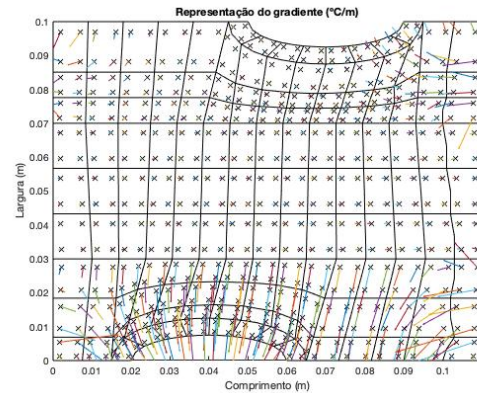


Figura 34: Gradiente de temperatura em MATLAB

5.3 Interpretação dos resultados

Comparando os gráficos obtidos em NX com os de MATLAB, pode-se verificar que, tanto para QUAD-4 como para QUAD-8 se obtiveram soluções semelhantes, uma vez que a distribuição de temperaturas é idêntica para ambos os casos, bem como a direção dos vetores de fluxo e de gradiente.

Nos gráficos que se seguem apresentam-se os desvios relativos entre os resultados obtidos na simulação em NX e no programa desenvolvido em Matlab, para os valores da temperatura, fluxo e gradiente de temperatura, que completam esta interpretação. No anexo é possível visualizar melhor os dados dos gráficos.

• Malha de QUAD-4

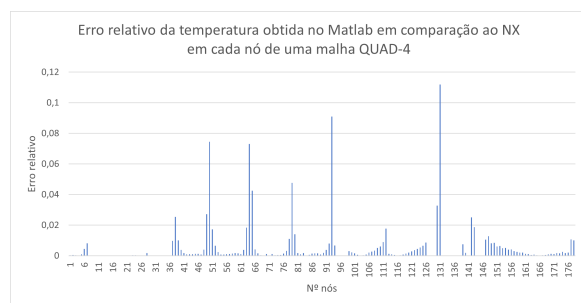


Figura 35: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores da temperatura

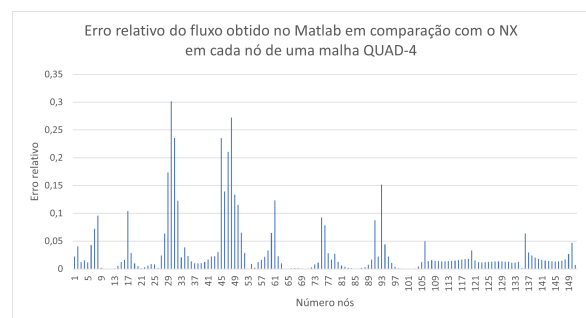


Figura 36: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores de fluxo

Como é possível observar pelos gráficos apresentados, a maioria dos nós apresenta um erro relativo do MATLAB em relação ao NX próximo de 0, ainda que haja casos em que este erro tem valores consideráveis. Isto permite-nos novamente concluir que os resultados

obtidos em NX e em MATLAB foram bastante semelhantes, pelo que se valida o programa desenvolvido.

- **Malha de QUAD-8**

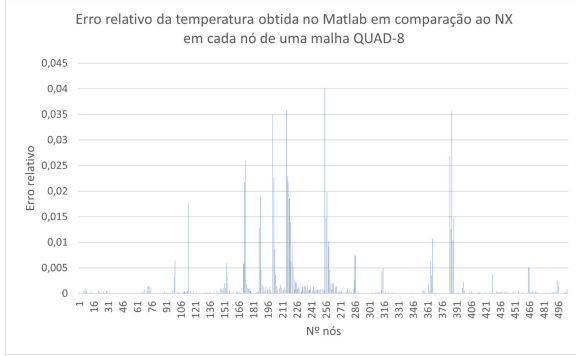


Figura 37: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores da temperatura

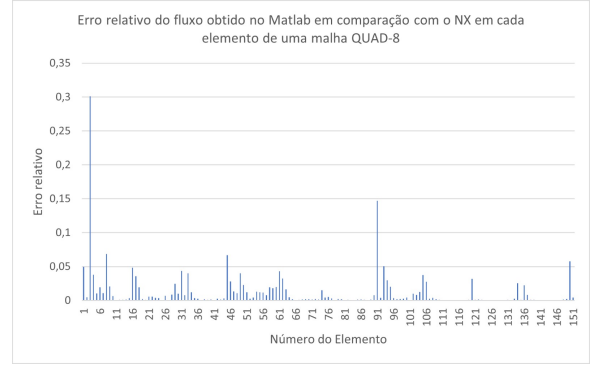


Figura 38: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores de fluxo

Mais uma vez, verifica-se que o erro relativo do MATLAB em relação ao NX é próximo de 0 tanto para a temperatura como para o fluxo.

Posto isto, ao comparar os erros relativos obtidos na malha linear e na quadrática, é possível analisar que os erros relativos são inferiores para a malha quadrática, tanto para a temperatura como para o fluxo. Estes resultados podem indicar que o programa do MATLAB para a malha quadrática pode ter sido mais eficiente do que para a malha linear.

É de frisar que para elementos quadráticos, o programa em *MATLAB* calcula o fluxo em 4 pontos diferentes do elemento, sendo depois feita uma média entre eles para comparar com os valores do NX.

6 Determinação da Potência Calorífica

A potência calorífica total que atrevesa a fronteira é dada pelo integral do fluxo ao longo da superfície de fronteira. Uma vez que a espessura da placa em análise neste projeto é unitária, basta integrar o fluxo ao longo da linha de fronteira vertical, como é dado pela equação 47:

$$P = \int_{\Gamma} q \, dy \quad (47)$$

Este integral pode ser aproximado ao somatório da potência calorífica em cada elemento que constitui a fronteira. Por sua vez, esta potência é obtida ao multiplicar o fluxo no determinado elemento pela sua altura, como podemos verificar pela equação 48:

$$P = \sum_{e=1}^n q^e |y_2^e - y_1^e| \quad (48)$$

Os resultados obtidos encontram-se sumarizados na tabela que se segue:

	Parede Interior	Parede Exterior	Parede Interior	Parede Exterior
	4 nós		8 nós	
Potência Calorífica em cada elemento da fronteira (W)	2,47112	5,101815	5,980515	10,72746
	1,724278	3,877786	1,226977	2,843742
	0,708882	1,915331	1,198111	3,595677
	27,36764	39,63439	25,86522	38,45211
	23,98206	34,02521	24,55323	35,94245
	24,90616	42,37222	24,26359	39,66267
	0,194912	4,931259	2,192692	7,329851
	5,411838	13,86419	5,364811	12,37654
	3,161891	5,044816	10,12514	16,14555
Potência Calorífica total (W)	89,93	150,77	100,77	167,08

Figura 39: Potência calorífica em cada elemento da fronteira e total (W)

7 Conclusões

A análise de convergência permitiu-nos concluir que, para o mesmo tipo de malha, o número de elementos influencia a precisão dos resultados obtidos. Verificou-se ainda que elementos de ordem superior convergem mais rápido do que elementos lineares, sendo a utilização de elementos quadráticos mais vantajosa para que a solução convirja com menos etapas de refinamento. No entanto, há que ter em conta que a utilização destes elementos implica maiores custos computacionais, o que nem sempre é desejável.

Relativamente aos resultados obtidos nas simulações em NX e no programa de MATLAB e tendo em conta os respetivos erros relativos já apresentados anteriormente, podemos concluir que os resultados obtidos no programa desenvolvido são uma boa aproximação dos resultados extraídos do programa de elementos finitos, o que era de certa forma expectável, uma vez que no desenvolvimento do programa em MATLAB se aplicou a teoria que rege os problemas de condução de calor.

Apesar de o erro entre estas abordagens ter sido baixo, o erro entre a solução analítica do problema simples e a solução computacional do MATLAB foi superior ao esperado, o que pode indicar que houve algumas aproximações a nível computacional que podem ter contribuído para essa discrepância de valores.

8 Anexo

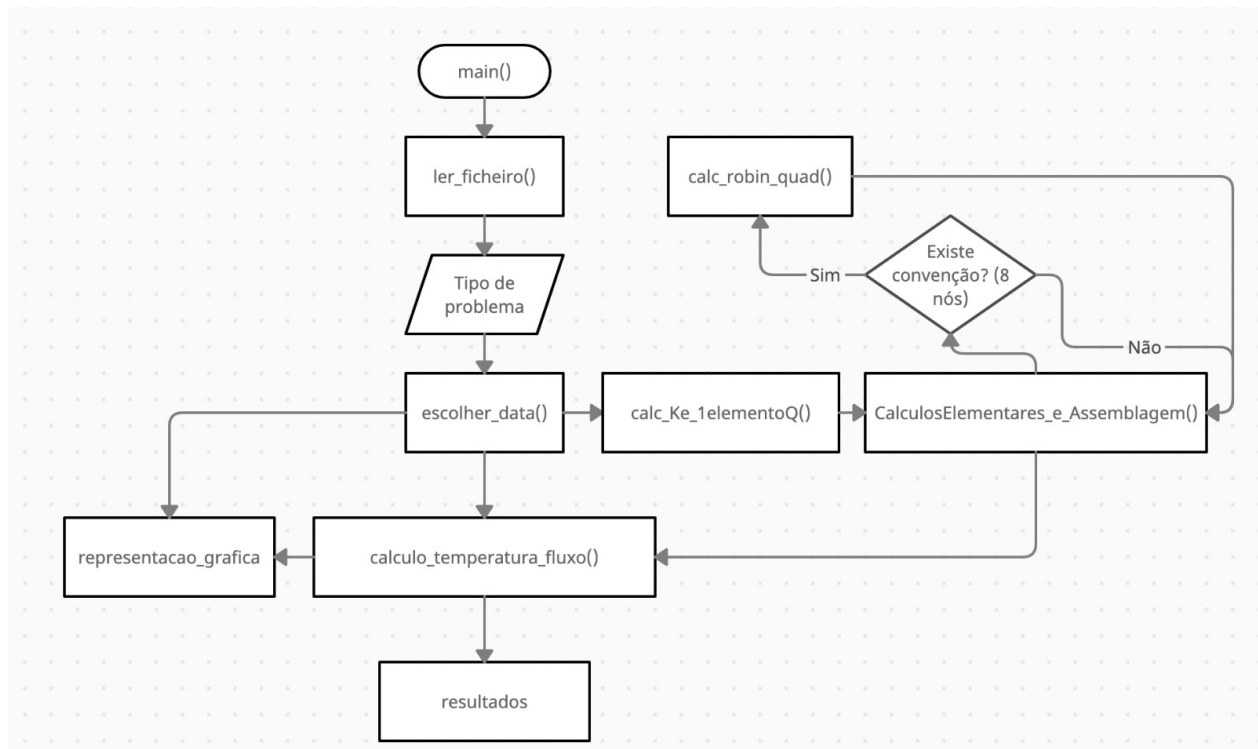


Figura 40: Fluxograma do problema desenvolvido

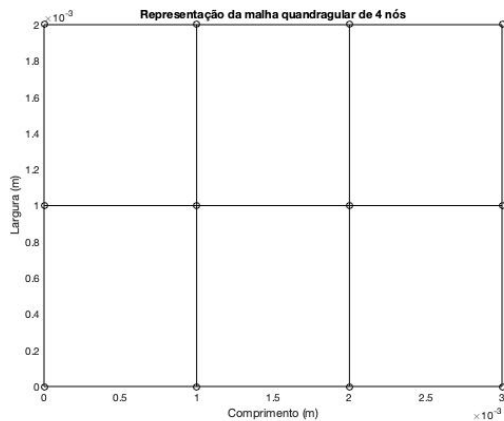


Figura 41: Malha de elementos QUAD-4 para problema simples

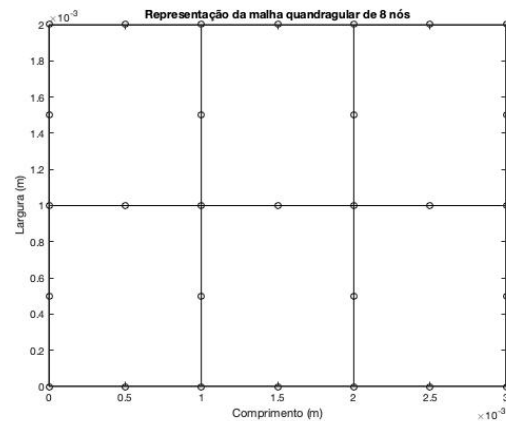


Figura 42: Malha de elementos QUAD-8 para problema simples

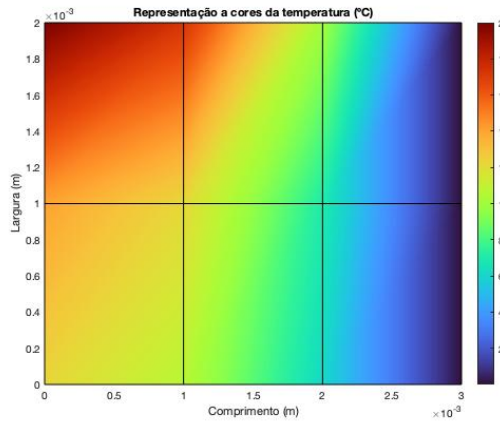


Figura 43: Gradiente de temperatura para problema simples [$^{\circ}\text{C}/\text{mm}$] QUAD-4

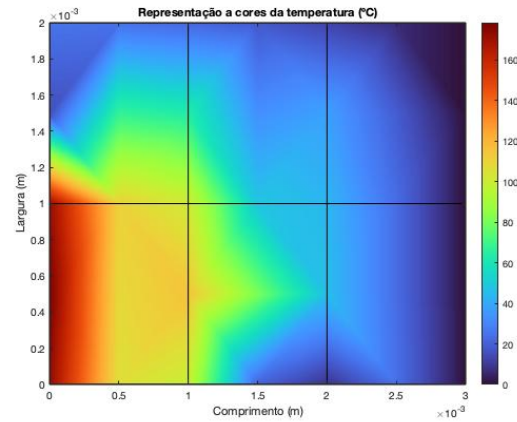


Figura 44: Gradiente de temperatura para problema simples [$^{\circ}\text{C}/\text{mm}$] QUAD-8

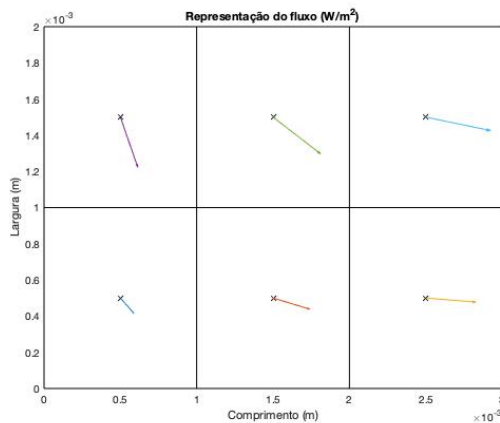


Figura 45: Fluxo de calor para problema simples [W/m^2] QUAD-4

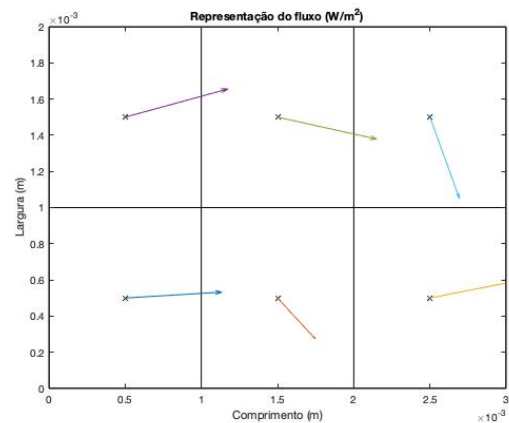


Figura 46: Fluxo de calor para problema simples [W/m^2] QUAD-8

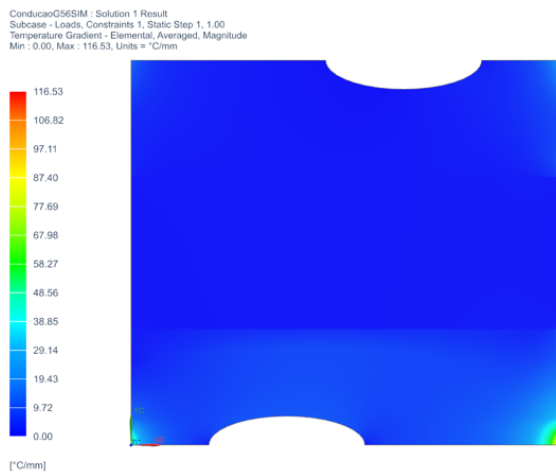


Figura 47: Gradiente de temperatura [$^{\circ}\text{C}/\text{mm}$] - QUAD-4 - Siemens NX

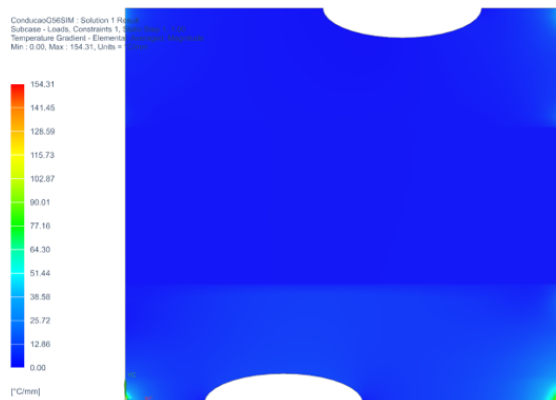


Figura 48: Gradiente de temperatura [$^{\circ}\text{C}/\text{mm}$] - QUAD-8 - Siemens NX

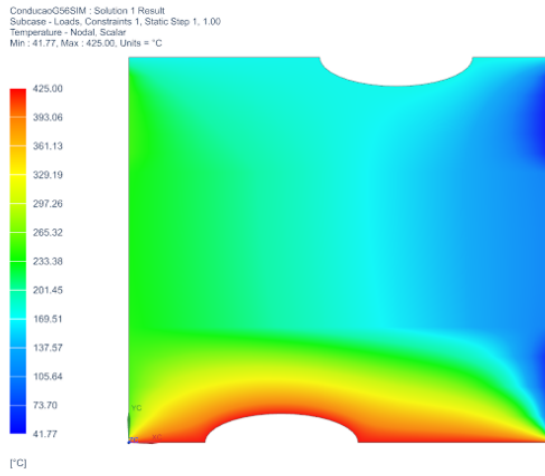


Figura 49: Temperatura nodal [°C] - QUAD-4 - Siemens NX

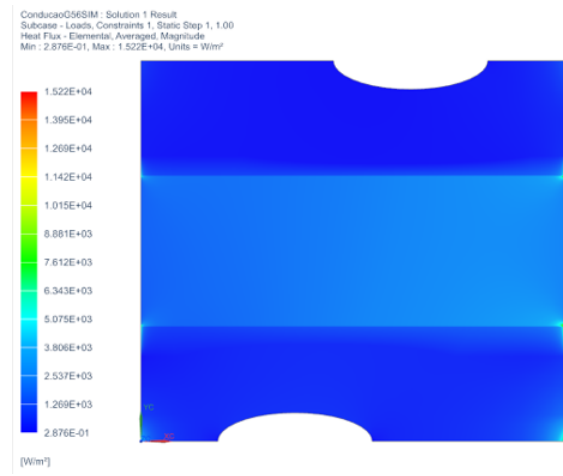


Figura 50: Fluxo de calor [W/m²] - QUAD-4 - Siemens NX

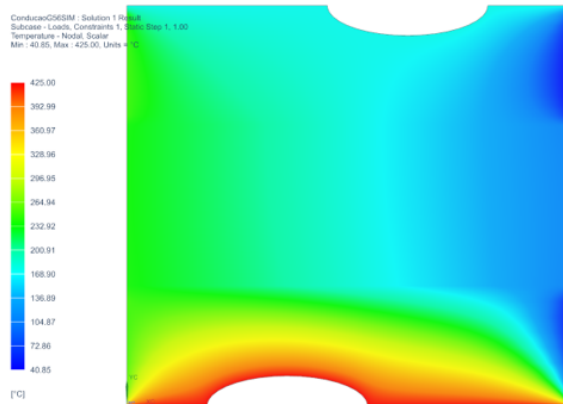


Figura 51: Temperatura nodal [°C] - QUAD-8 - Siemens NX

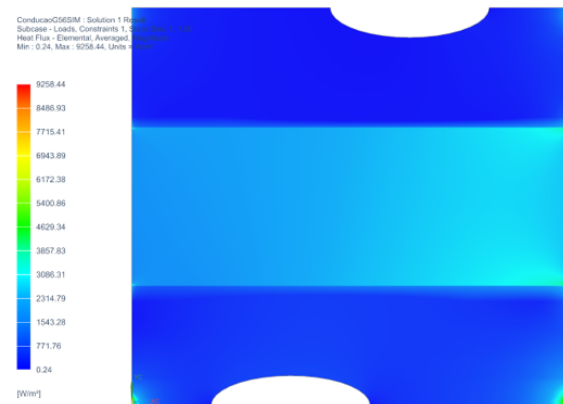


Figura 52: Fluxo de calor [W/m²] - QUAD-8 - Siemens NX

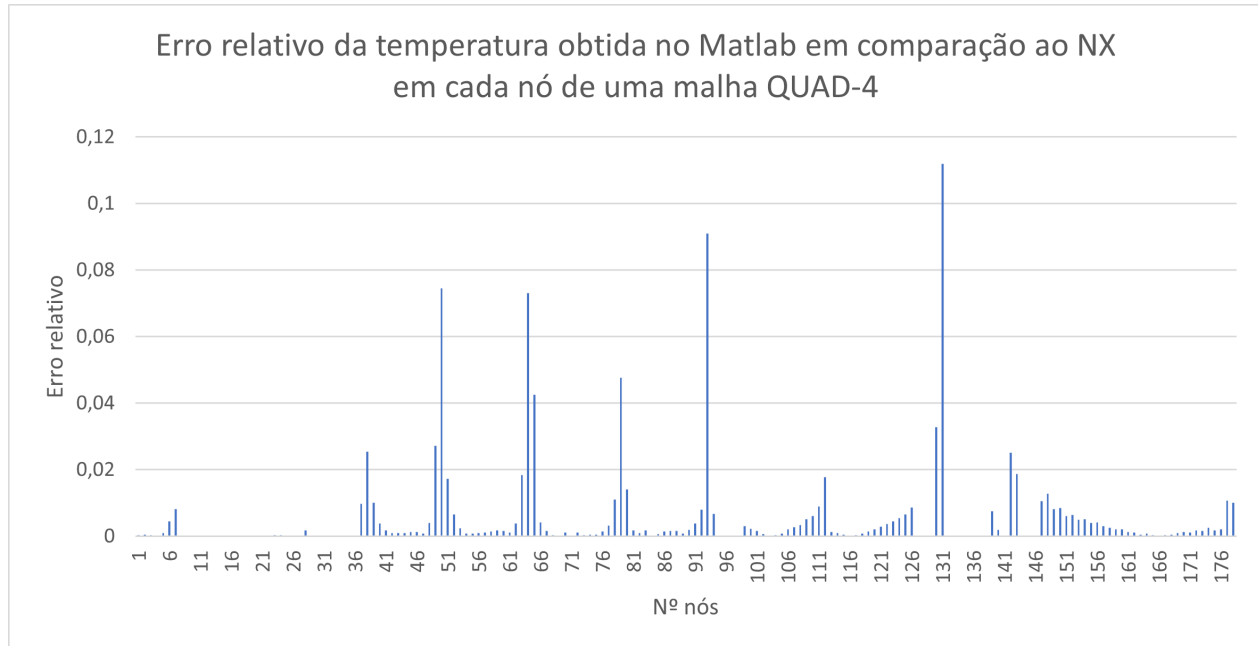


Figura 53: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores da temperatura

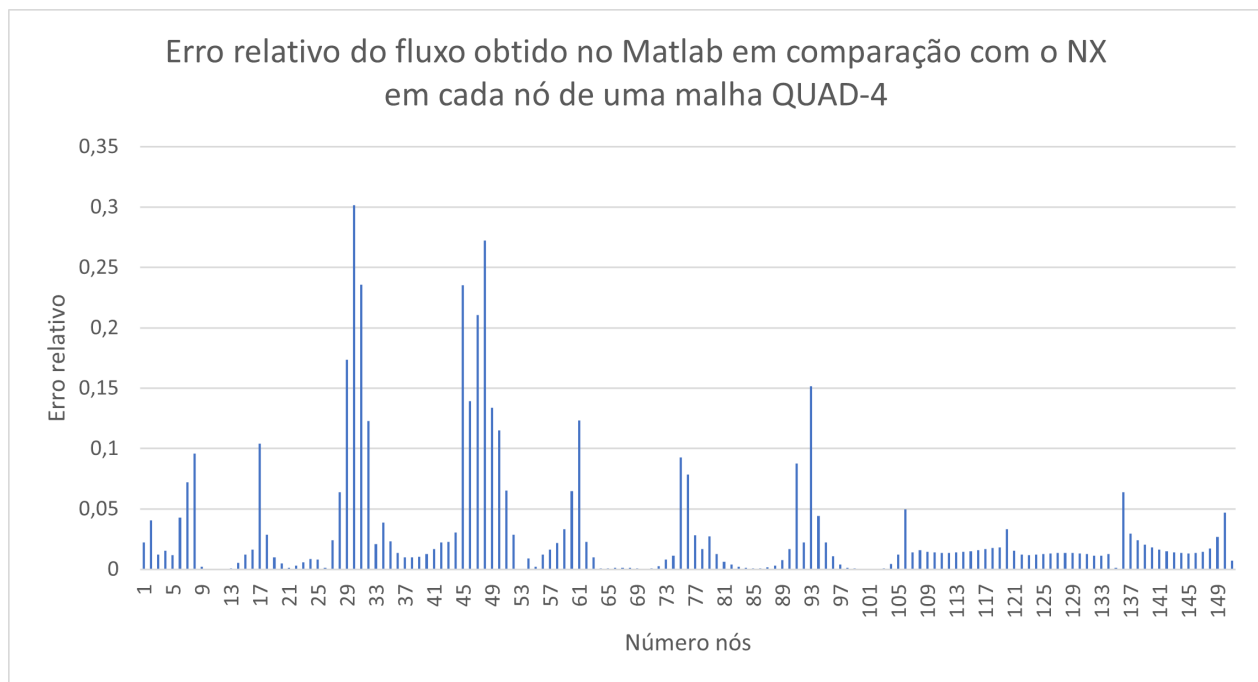


Figura 54: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores de fluxo

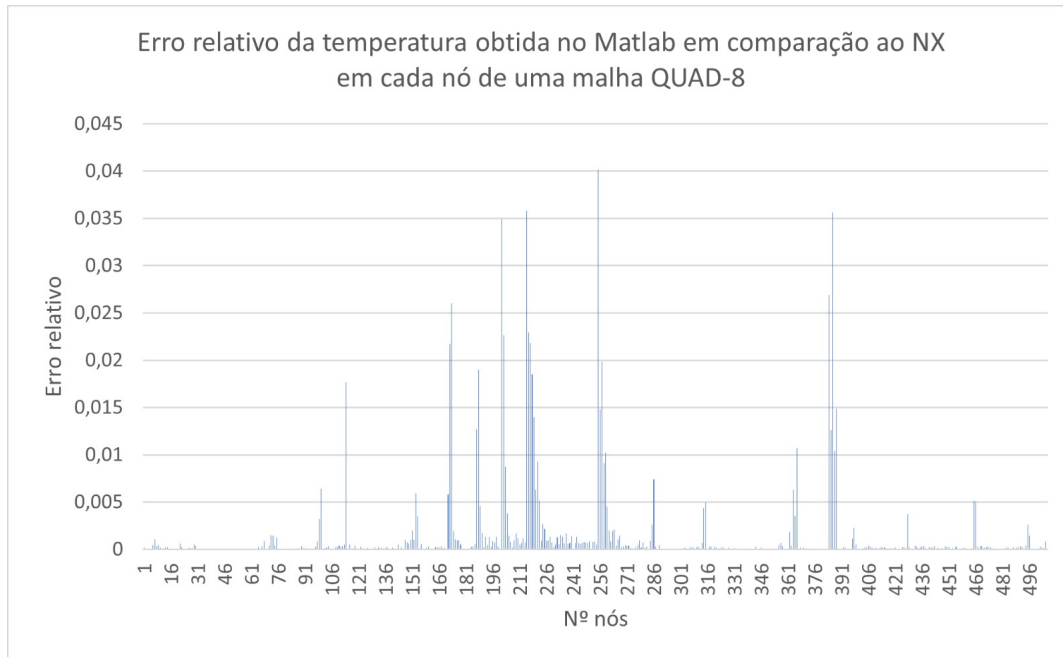


Figura 55: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores da temperatura

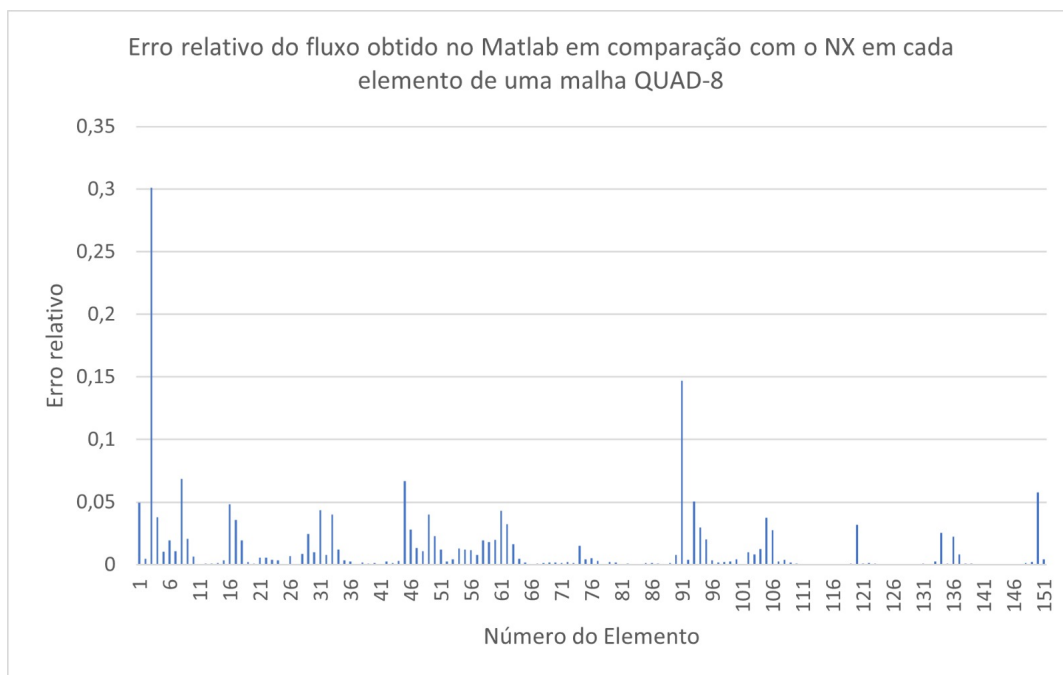


Figura 56: Desvio relativo entre o programa desenvolvido em MATLAB e a simulação em NX para valores de fluxo

9 Referências

[1] J. N. Reddy, An Introduction to the Finite Element Method, 3rd ed., McGraw-Hill Education (2005)