



INSTITUTO
SUPERIOR
TÉCNICO

Lisboa, 09 de Maio de 2011

Manual do Utilizador

Programa TEACH-C

Filipe Soares, nº 57140

Baseado no manual escrito por:

- Prof. V. Semião
- Prof. P. Oliveira
- Prof. M.G. Carvalho

Índice

Introdução	1
Constituição do Programa	3
Funcionamento do Código	7
Verificação do Código	9
1-Temperatura imposta	9
Resultados	27
2-Convecção e simetrias	30
Resultados	31
3-Distribuição de Temperaturas em barra de combustível nuclear.....	32
Resultados	33
4-Fluxo de Calor imposto	36
Resultados:	37
Bibliografia	49
Anexo 1-Código	51
Anexo 2-Listagem de variáveis	73
Anexo 3.....	75
Anexo 4.....	77

Introdução

O programa TEACH-C permite a resolução de problemas de condução de calor sendo que para cada caso é necessário efectuarem-se modificações no código principal do programa. Este manual explica, de uma forma sucinta, o código e as possíveis alterações a fazer neste. Para problemas mais complexos é recomendado a leitura do manual que acompanha o programa, [1] e [6].

Esta versão do programa TEACH-C permite a resolução de problemas de condução de calor bidimensionais com as seguintes variantes:

- Coordenadas cartesianas ou coordenadas cilíndricas (axissimétrico);
- Regime estacionário ou transiente (não estacionário);
- Condutividade térmica (k) uniforme ou variável (com a temperatura ou com o espaço);
- Três tipos de condições de fronteira:
 - Temperatura imposta;
 - Fluxo imposto;
 - Convecção na fronteira do corpo, ou seja, coeficiente de convecção (h) e temperatura do fluido exterior impostos.

Constituição do Programa

Este programa é uma versão do programa TEACH-C escrita em MATLAB em vez de FORTRAN. É constituído por várias funções que serão descritas em seguida.

✓ Programa PRINCIPAL/CONTROL

Este módulo do programa é bastante importante do ponto de vista do utilizador pois é nele que se definem a maioria dos parâmetros:

- Tipo de regime – Estacionário ou Transiente;
- Tipo de coordenadas – Cartesianas ou cilíndricas;
- Tipo de material – condutividade térmica uniforme ou variável;
- Dimensões do corpo e da malha utilizada;
- Condições de fronteira.

✓ Função PROPS

Esta função tem como finalidade a especificação das leis de variação das propriedades do material, nomeadamente a condutividade térmica, k .

✓ Função INIT

Esta função não deverá, em princípio, ser alterada. O seu papel é por um lado inicializar as matrizes a zero e, por outro, calcular todos os valores associados à malha utilizada, como por exemplo:

- Distância entre nós;
- Dimensões dos volumes de controlo;
- Fronteiras dos volumes de controlo.

Na Figura 1 está representada a malha de um volume de controlo genérico e os respectivos parâmetros geométricos.

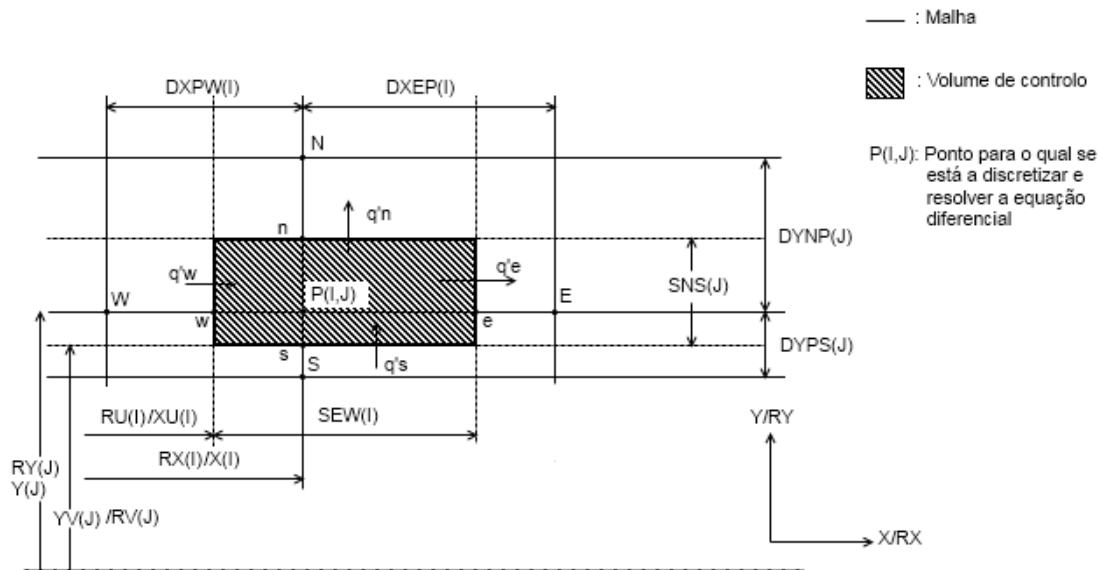


Figura 1 – Parâmetros geométricos da malha.

✓ Função CALCT

Este bloco permite calcular os coeficientes da equação diferencial de calor discretizada.

Uma vez que se utiliza o método implícito, com ou sem sub-relaxação, a equação mencionada é da seguinte forma:

$$\frac{(A_p - S_p)}{k} T_p = A_N T_N + A_S T_S + A_E T_E + A_W T_W + S_U + \frac{1-k}{k} (A_P + S_P)^{n-1} T_P \quad (1)$$

Em que k é o coeficiente de sub-relaxação. A dedução desta equação pode ser encontrada no manual TEACH-C, [1] e [6].

O utilizador só deverá modificar este bloco se o seu problema tiver fontes ou poços. Mais adiante, estas modificações serão explicadas.

✓ Função SOLVE

Esta função tem como finalidade a resolução de um sistema de equações em que a matriz dos coeficientes é uma matriz tri-diagonal.

O método utilizado por esta função integra duas partes: Iteração linha a linha e Ajustamento de bloco.

Iteração linha a linha

Este processo é aplicado a cada linha da malha com o sentido NORTE-SUL sendo o varrimento de todas as linhas “NORTE-SUL” efectuado no sentido OESTE-ESTE. Um varrimento total corresponde a uma iteração.

O objectivo desta parte é eliminar os erros da solução do campo de temperatura, “empurrando-os” para a fronteira, onde serão eliminados pelas condições de fronteira impostas.

Ajustamento de bloco

Em algumas situações os erros (resíduos) referidos anteriormente não são eliminados na fronteira e se o são, é de uma forma demasiado lenta.

Além disso, durante o processo iterativo linha a linha, enquanto se está a resolver a equação (1) para uma dada linha i , admite-se que os valores de T para as linhas $i-1$ e $i+1$ (os últimos calculados), vizinhas de i , são conhecidos. Se por um lado, os valores de $i-1$ são os da iteração actual, os valores para a linha $i+1$ são os da iteração anterior. Ao calcular o resíduo da linha i com os valores anteriores (T_{i-1}, T_i, T_{i+1}), este tenderá a ser nulo. No entanto, ao saltar para a linha seguinte, novos valores de T serão calculados para essa linha, e o resíduo da linha anterior passa a ser finito (não nulo).

O ajustamento de bloco soluciona estes problemas. O processo utilizado neste ajustamento consiste no incremento do valor da temperatura de cada linha i de um valor δT_i (constante para essa linha). Para linhas diferentes o valor de δT_i será diferente de forma a provocar um valor do resíduo nulo para cada linha, garantindo assim que o valor do resíduo de todo o campo é nulo. Este assunto encontra-se explicado com maior detalhe em [1] e em [6].

Esta função foi testada para inúmeros casos e como não depende do tipo de problema a resolver, não é aconselhada a sua modificação.

É ainda importante notar que o ajustamento de blocos é feito após um varrimento completo (uma iteração completa linha a linha), e que a matriz dos coeficientes de δT_i é também tri-diagonal.

✓ **Função PROMOD**

Do ponto de vista do utilizador, este bloco é de extrema relevância, pois é aqui que se definem os tipos de condição de fronteira a que o problema em estudo está sujeito. O utilizador só terá de escolher as condições de fronteira e introduzir os respectivos dados do problema pois esta função já contém as instruções para cada condição de fronteira. Adiante será explicado como foi feito.

✓ **Função PRINT**

Destina-se a produzir o “output” pretendido. Esta função não necessita de ser alterada embora possa sofrer modificações pontuais consoante o que se pretende.

Funcionamento do Código

Em seguida, é apresentado na Figura 2, o fluxograma que explica o modo como o programa resolve o problema de condução de calor assim como o que explica a estrutura do programa e as interações entre as várias funções que o constituem, Figura 3.

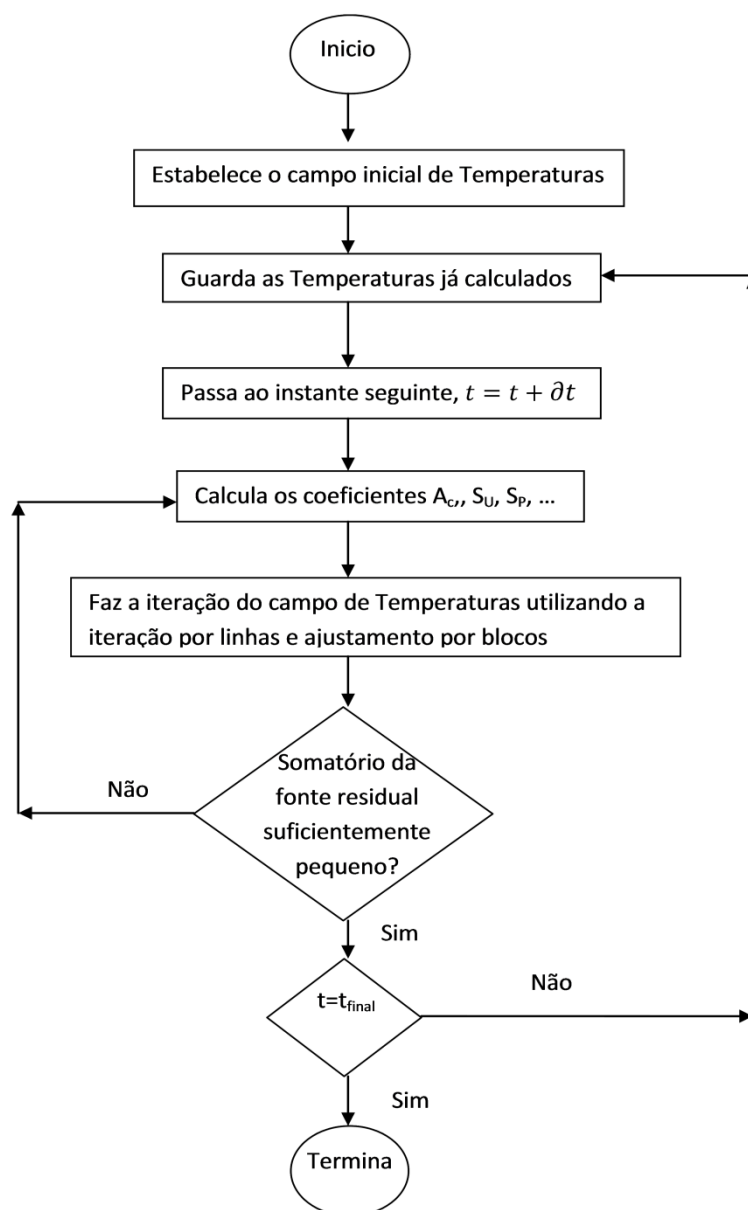


Figura 2 – Fluxograma que mostra o modo como o programa resolve o problema de condução de calor.

Verificação do Código

Em seguida são apresentados os vários testes efectuados ao código de modo a fazer-se a sua verificação.

É dado um maior relevo ao primeiro caso de forma a que o programa e as suas possíveis modificações sejam explicados. No final de cada caso são apresentados os resultados finais obtidos.

1-Temperatura imposta

No Anexo 1 encontra-se o código do programa TEACH-C em que a versão base está preparada para resolver problemas de temperatura imposta como se apresenta em seguida. O programa está preparado para receber outras condições de fronteira sendo necessário modificar apenas certos parâmetros no programa principal e indicar o tipo de condição de fronteira na função PROMOD.

O primeiro caso testado é o seguinte (Figura 4):

Estudo da distribuição de temperaturas da secção de uma barra (com comprimento suficientemente grande para se poder desprezar o efeito dos topos na secção em estudo) inicialmente à temperatura de 0°C . A secção é quadrangular de $1\text{m} \times 1\text{m}$. Subitamente é imposta no topo da secção a temperatura de 100°C mantendo-se a 0°C os restantes lados da secção. Dadas as propriedades do material ($k=14,68 \text{ [W/m.K]}$, $C_v=485,67 \text{ [J/kg.K]}$, $\rho=7800 \text{ [kg/m}^3\text{]}$), pretende-se calcular a evolução de temperaturas no tempo.

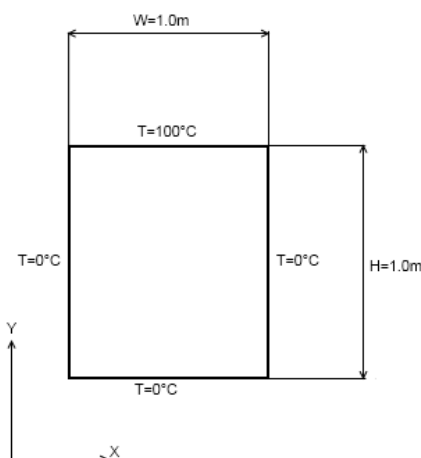


Figura 4 – Figura da secção da barra e das condições de fronteira aplicadas (Temperatura imposta).

De seguida é descrita a forma como se introduziu o problema no programa. Para tal, analisaram-se as funções indicando-se os parâmetros sujeitos a alterações.

- **Programa Principal/CONTROL**

Capítulo 0 – Preliminares

Os parâmetros IT (=22, neste caso) e JT (=22) referem-se às dimensões das matrizes utilizadas nas funções. Como é óbvio, estes valores são iguais ao número de nós utilizados segundo x e y, respectivamente. A sua alteração modifica a dimensão das matrizes.

NI (=22) e NJ (=22) são o número de pontos da malha segundo x e y, respectivamente. Estes valores podem variar mas terão de ser sempre inferiores ou iguais a IT (22) e a JT (22).

Capítulo 1 – Parâmetros e índices de controlo

As variáveis INCYLX e INCYLY funcionam como variáveis lógicas pois permitem definir as coordenadas (cartesianas ou cilíndricas) em que se trabalha. A seguinte tabela estabelece as configurações possíveis.

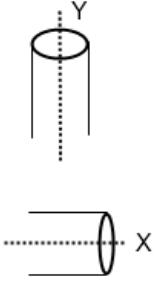
Coordenadas	INCYLX	INCYLY	
Cartesianas	0	0	
Cilíndricas (axissimétrico)	1	0	
	0	1	

Tabela 1 – Configurações das variáveis INCYLX e INCYLY.

É importante referir o facto de as variáveis não poderem ter em simultâneo o valor 1 (o programa irá acusar o erro e termina). Consoante a simetria seja segundo x ou y, tem-se que INCYLY=1 ou INCYLX=1, respectivamente, sendo qualquer dos casos axissimétrico.

A nível de condições de fronteira o código está preparado para receber coordenadas cartesianas e cilíndricas com o eixo de simetria segundo xx (INCYLY=1 e INCYLX=0). No caso de ser necessário que o eixo de simetria esteja alinhado com yy bastará modificar as áreas (ver como estão definidas áreas norte e este na função CALCT).

W (=1.0) e H (=1.0) são as dimensões da secção da barra, nomeadamente largura e altura, conforme mostra a Figura 4. Estes valores podem ser modificados, devendo-se dar as dimensões máximas da secção. Por exemplo, numa secção em I devem ser dadas a largura e altura máximas da secção.

Cálculo das coordenadas da malha

A Figura 5, abaixo, representa a malha no caso em estudo. Esta foi construída de uma forma uniforme para todo o domínio de solução. Antes de se explicar como se procedeu para a construção da malha no programa, chama-se a atenção do utilizador para os seguintes factos:

- Os pontos exteriores da malha (limites) estão fora do domínio de solução. Assim não fará sentido calcular nestes pontos os valores das temperaturas, embora a sua existência seja imprescindível pois, como se verá adiante, são nestes pontos da malha (e portanto nas posições das matrizes do programa) que se guardam os valores impostos pelas condições de fronteira.

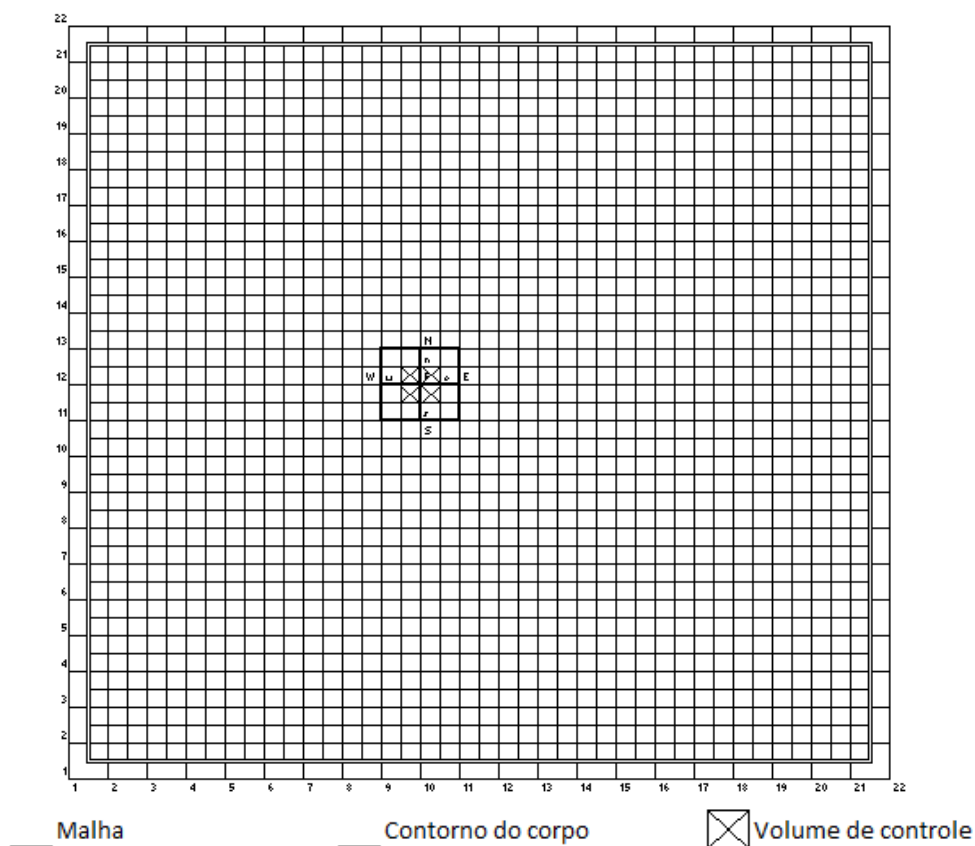


Figura 5 – Malha e volumes de controlo utilizados neste teste.

- A interface de um volume de controlo está sempre entre dois nós (a meia distância). Este aspecto está ilustrado na seguinte figura.

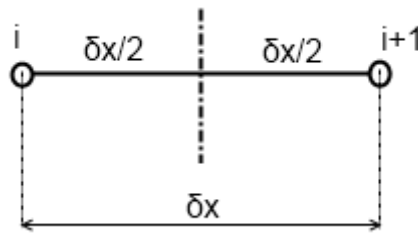


Figura 6 – Interface de um volume de controlo.

O utilizador não se apercebe deste facto mas o programa constrói os volumes de controlo dos nós desta forma. O utilizador, ao construir a malha, precisa apenas de garantir que a fronteira do corpo se situa a meia distância (exactamente) das linhas que a limitam.

Dos dois aspectos explicitados anteriormente resulta que o ponto (1,1) terá coordenadas negativas de valor igual a meia dimensão da malha, isto é:

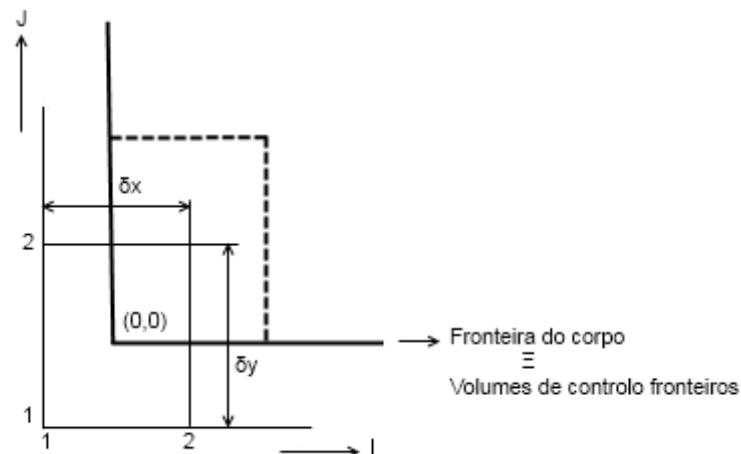


Figura 7 – Fronteira do corpo e volumes de controlo fronteiros.

Em que:

$X(1) = -\Delta x/2$ – abcissa dos pontos da coluna 1;

$Y(1) = -\Delta y/2$ – abcissa dos pontos da coluna 1.

Sendo a malha uniforme e a dimensão do corpo igual a 1, o valor DX (DY) é obtido por:

$$DX = \frac{W}{NIM1 - 1} = \frac{1}{20} = 0.05$$

$$DY = \frac{H}{NJM1 - 1} = \frac{1}{20} = 0.05$$

Em que:

$$NIM1 = NI - 1$$

$$NJM1 = NJ - 1$$

Assim, para este caso, a malha é traçada a partir do seguinte código:

```
%Calcula as abcissas na direcção xx
DX(1)=W/(NIM1-1);
X(1)=-0.5*DX(1);
for I=2:NI;
    DX(I)=DX(1);
    X(I)=X(I-1)+DX(I-1);
end

%Calcula as abcissas na direcção yy
DY(1)=H/(NJM1-1);
Y(1)=-0.5*DY(1);
for J=2:NJ;
    DY(J)=DY(1);
    Y(J)=Y(J-1)+DY(J-1);
End
```

Verificando-se que:

X(1)=-0.025 X(2)=0.025	$X_{\text{Fronteira Este}}=0$
X(21)=0.975 X(22)=1.025	$X_{\text{Fronteira Oeste}}=1$
Y(1)=-0.025 Y(2)=0.025	$Y_{\text{Fronteira Sul}}=0$
Y(21)=0.975 Y(22)=1.025	$Y_{\text{Fronteira Norte}}=1$

A malha utilizada não é a mais indicada. Esta deve ser tal, que em zonas em que os gradientes da propriedade em estudo (neste caso a temperatura) sejam mais intensos esta seja mais fina. Nas zonas em que esses gradientes sejam menos intensos a malha deverá ser mais espaçada. O “output” dos resultados obtidos permitirá definir uma malha mais correcta para o problema em estudo.

Definição do domínio de solução

O domínio de solução pode não abranger todos os nós da malha, ou seja, apesar de I e J estarem contidos nos limites $1 \leq I \leq NI$ e $1 \leq J \leq NJ$ não significa que todos os pontos (I,J) pertençam ao domínio de solução.

Para definir o domínio de solução o programa utiliza dois vectores auxiliares que para uma dada linha I (coluna) estabelecem a fronteira a Norte e a Sul através da linha J limite. Isto significa que para cada linha I os limites de J são:

$$JS(I) \leq J \leq JN(I)$$

Em suma, o domínio de solução fica perfeitamente definido se atendermos aos limites de cada um dos eixos, sendo o domínio de solução o conjunto dos nós interiores.

$$2 \leq I \leq NI-1$$

$$JS(I) \leq J \leq JN(I)$$

Partindo do que já foi referido verifica-se que o domínio de solução para este caso é:

$$2 \leq I \leq 21$$

$$2 \leq J \leq 21$$

No programa estes limites dão-se em:

```
%Estabelece limites do Domínio
for I=1:NI;
    JS(I)=2;
    JN(I)=NJ-1;
end
```

De modo a uma melhor compreensão imagine-se que se tem a seguinte fronteira algures no corpo:

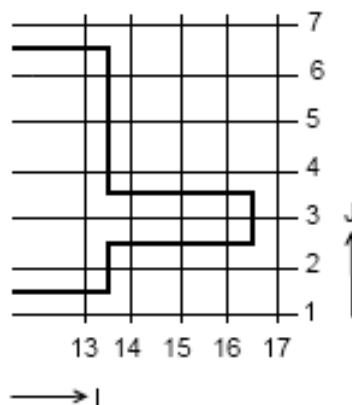


Figura 8 – Fronteira de corpo.

A partir de $I=13$ o domínio de solução seria definido por:

```
I=13;  
JS (I) =2;  
JN (I) =6;  
for I=14:17;  
    JS (I) =3;  
    JN (I) =3;  
end
```

Ponto monitor

Este é o ponto que no processo iterativo permite observar a convergência do processo. Deve ser escolhido um ponto do domínio de solução cuja evolução qualitativa seja previsível. Este ponto é indicado no programa pela sua posição de memória:

IMOM=6

JMON=6

Ao observar-se o “output” verifica-se que no processo iterativo é listado o valor de $T(6,6)$.

Propriedades do Material

As propriedades do material do problema são:

- Condutividade térmica, k [W/m.K]
- Calor específico, C_v [J/kg.K]
- Massa específica, ρ [kg/m³]

As propriedades devem ser especificadas para todos os nós que façam parte do domínio de solução e da fronteira. É de notar que se o ponto (I,J) tiver um dado conjunto de propriedades, o volume de controlo que o envolve também terá as mesmas propriedades.

No caso testado, o valor de k é constante para todos os nós uma vez que o meio é homogéneo. Assim tem-se o seguinte código:

```
%Propriedades do material(Tcond=cond,CV=calor esp.,DENSIT=dens),
%Meio homogéneo
%Aço %%%% Alterável %%%%
for I=1:NI;
    for J=1:NJ;
        TCON(I,J)=14.68;
        CV(I,J)=485.67;
        DENSIT(I,J)=7800;
        if I==1 && J==1;
            BK=TCON(I,J);
        end
    end
end
```

É de referir os seguintes aspectos importantes sobre o material:

- Se o meio não for homogéneo (se existirem valores diferentes de k , ou seja, materiais diferentes) é necessário dar essa informação ao programa alterando ligeiramente o código. Por exemplo, imagine-se o caso em que no meio do corpo existe um material diferente.

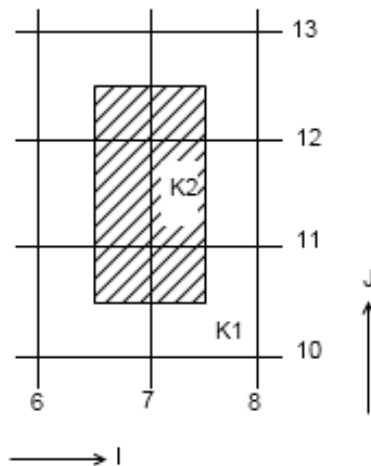


Figura 9 – Corpo constituído por dois materiais.

Além das instruções para os restantes pontos de domínio de solução, o programa deverá conter as seguintes instruções:

```
I=6;
for J=10:13;
    TCON(I,J)=K1;
end

I=7;
for J=11:12;
    if J==11;
        TCON(I,J-1)=K1;
    elseif J==12;
        TCON(I,J+1)=K1;
    end
    TCON(I,J)=K2;
end

I=8;
for J=10:13;
    TCON(I,J)=K1;
end
```

- Outro aspecto a ter em conta é o “if” que aparece no primeiro conjunto de instruções para as propriedades do material. Quando o meio não for homogéneo e houver vários materiais essa instrução deve desaparecer sendo substituída por um conjunto de instruções que procurem o valor de k mínimo de todos os nós. Propõe-se o seguinte código a introduzir no programa, depois de todas as propriedades introduzidas:

```
BK2=GREAT;
for I=I:NI;
    for J=Y:NJ;
        BK1=MIN(BK2,TCON(I,J))
        BK2=BK1
        BK=BK1
    end
end
```

Este valor de k mínimo tem de ser obtido por questões de critério de convergência utilizado pelo programa.

Em suma, se o material for homogéneo bastará substituir os valores das propriedades nas respectivas instruções, caso contrário o utilizador deverá por um lado informar o programa dos sítios exactos (nós) em que o material é diferente (ou seja, terá de dar a distribuição das propriedades dos diferentes nós) e por outro garantir que o valor de “BK” foi calculado e corresponde ao mínimo dos valores de k presentes.

Parâmetros de controlo do programa

Os primeiros parâmetros que aparecem no código são:

```
MAXIT=100  
MAXSTP=20  
NITPRI=50  
NSTPRI=1
```

Isto significa que o programa irá efectuar 20 iterações no tempo (MAXSTP) em que para cada uma destas, fará no máximo 100 iterações (MAXIT). Para cada iteração no tempo (NSTPRI), a cada 50 iterações (NITPRI) o programa irá imprimir o campo de temperaturas. Estes parâmetros devem ser alterados pelo utilizador.

Em seguida aparece o parâmetro URFT com o valor 1. Este parâmetro é o coeficiente de sub-relaxação (“under-relaxation factor”) que diz que o valor da iteração i é apenas uma percentagem da solução anterior. Só pode tomar valores entre 0 e 1, não existindo sub-relaxação quando este toma o valor de 1.

Só é aconselhável utilizar sub-relaxação (desaceleração das variações dos valores da variável) no processo iterativo linha-a-linha para problemas não lineares. A sua utilização evita muitas vezes a divergência da solução iterativa em equações fortemente não lineares (visto que o método lineariza as equações).

O parâmetro SORMAX (0.001) define o valor máximo do resíduo constituindo o critério de paragem. Não deve ser modificado no caso de se trabalhar com precisão simples, evitando assim uma falsa divergência do método uma vez que não se pode tirar à máquina maior precisão do que ela nos pode dar.

DT=50 indica que o intervalo de tempo entre duas iterações é de 50 segundos. Este valor é arbitrado pelo utilizador, não sendo conveniente um valor muito elevado.

Por fim existem mais duas variáveis, INTIME e INPRO, funcionando ambas como lógicas. “INTIME” define o regime do problema, ou seja, se o problema é transiente (INTIME=1) ou estacionário (INTIME=0). No caso estudado esta variável toma o valor de 1.

INPRO (neste caso é igual a 0) indica se o coeficiente de condução de calor, k , varia no espaço ou com a temperatura. A forma como k varia deverá ser indicada na função PROPS, como se irá observar mais adiante.

Capítulo 2 - Operações iniciais

As operações que o utilizador poderá efectuar nesta secção do código são:

Atribuição das condições de fronteira

No caso em estudo, sendo as condições de fronteira do tipo temperatura imposta tem de se definir os seus valores:

TTOP=100

TBOT=0

TLEFT=0

TRIGHT=0

Como já referido, é necessário atribuir valores de temperatura nos nós da malha que representam a fronteira (nós exteriores imediatamente vizinhos das fronteiras).

À primeira vista, seria lógico atribuir estes valores respectivamente às fronteiras norte, sul, oeste e este. No entanto, se atendermos a que o varrimento do processo linha-a-linha é feito da esquerda para a direita (OESTE-ESTE) pelo programa, é vantajoso que a condição de fronteira diferente das outras e da condição inicial seja rapidamente transmitida ao interior do campo de solução, ou seja, varrimento no sentido do fluxo predominante. Assim, de forma a obter-se uma convergência mais rápida deverá ser utilizado o seguinte esquema e código (uma vez que a secção é quadrangular):

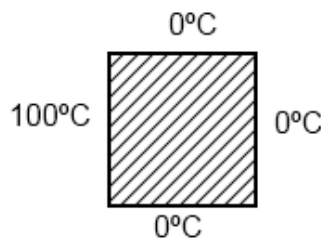


Figura 10 – Condições de Fronteira.

```
%Valores de fronteira
TTOP=0;
TBOT=0;
TLEFT=100;
TRIGHT=0;
```

```
for I=2:NIM1
    T(I,1)=TBOT;
    T(I,NJ)=TTOP;
end
```

```
for J=2:NJM1
    T(1,J)=TLEFT;
    T(NI,J)=TRIGHT;
end
```

Do exemplo anterior verifica-se que as condições de fronteira (e a posição em que se coloca o corpo e malha) devem ser tais que o fluxo de calor predominante fique na direcção do varrimento, ou seja, de oeste para este. Por exemplo, para o seguinte caso:

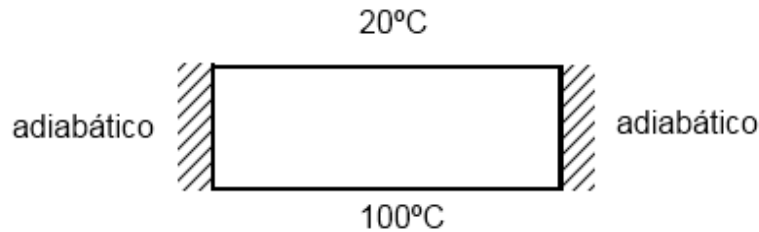


Figura 11 – Condições de fronteira, Temperatura e fluxo imposto.

Na Figura 12, em baixo, é possível observar a forma mais correcta de colocar o corpo (esta colocação refere-se ao traçar da malha) de modo a tornar a convergência mais rápida:

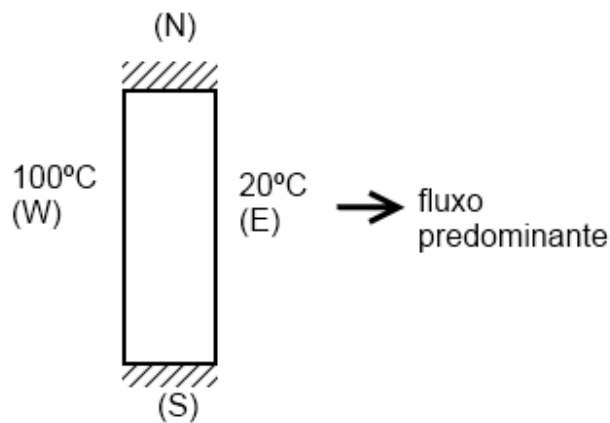


Figura 12 – Corpo colocado de forma a aumentar a velocidade de convergência.

A segunda operação que o utilizador deverá realizar é a inicialização do campo de temperaturas. Para tal é apenas necessário atribuir um valor à variável TINIC.

- **Função PROPS**

Caso a condutividade térmica varie com a temperatura ou posição, deverá ser introduzida uma função que caracterize tal variação. No caso em que k é constante esta função não deverá ser alterada.

- **Função INIT**

Esta função efectua a inicialização das variáveis. Não deverá ser alterada.

- **Função CALCT**

Esta função só deverá ser modificada caso o problema em estudo contenha uma distribuição de fontes/poços. Assim, $SU(J)=0$ e $SP(J)=0$ deverão ser substituídos pelos respectivos valores ou expressões das/os fontes/poços presentes no problema.

É de salientar a necessidade de a fonte/poço ser introduzida/o de uma forma linearizada. A equação da fonte/poço a introduzir é:

$$S = B_p T_p C_p$$

Em que B_p , por questões de convergência do método, deve ser sempre menor ou igual a zero.

Os dois exemplos que se seguem, ilustram a forma como se introduzem as fontes no programa, acrescentando instruções a $SU(J)=0$ e $SP(J)=0$.

Exemplo 1: Para simular uma fonte cujo comportamento seja $S=4-5T$ (uma vez que a fonte é linear) deve introduzir-se o seguinte código.

```
BP=-5
CP=4
SU (J) =SU (J) +CP
SP (J) =SP (J) +BP
```

Exemplo 2: No caso de uma fonte $S=4-5T^3$, não linear, para a sua linearização deverá ter-se:

$$S = S^* + \left(\frac{dS}{dT}\right)^* (T_p - T_p^*)$$

Em que “*” significa o valor da variável na iteração anterior. Neste caso tem-se:

$$S^* = 4 - 5T_p^{*3} \qquad \left(\frac{dS}{dT}\right)^* = -15T_p^{*2}$$

$$S = 4 - 5T_p^3 - 15T_p^{*2}(T_p - T_p^*) = 4 - 5T_p^3 + 15T_p^{*3} - 15T_p^{*2}T_p$$

$$S = 4 + 10T_p^{*3} - 15T_p^{*2}T_p$$

$$\begin{cases} CP = 4 + 10T_p^{*3} \\ BP = -15T_p^{*2} \end{cases}$$

No programa deve-se escrever:

```
%Termos fonte  
CP=4+10*T(I,J)^3;  
BP=-15*T(I,J)^2;  
SU(J)=SU(J)+CP;  
SP(J)=SP(J)+BP;
```

É de notar que nesta fase do programa o valor actual de T(I,J) ainda não foi calculado, pelo que será utilizado o valor da iteração anterior.

Estas mesmas equações podem ser utilizadas na função PROMOD que como se verá utiliza fontes fictícias para implementação das condições de fronteira.

- **Função SOLVE**

Não se devem efectuar modificações nesta função.

- **Função PRINT**

Esta função só deve ser modificada caso seja pretendido um “output” diferente.

- **Função PROMO**

Esta é a parte do código onde se implementa as condições de fronteira. Para tal deve-se utilizar o seguinte artifício:

Cada condição de fronteira é imposta por meio de uma fonte fictícia, linearizada (semelhante às fontes reais):

$$S = B_p T_p + C_p$$

Assim, consoante as condições de fronteira aplicadas, o B_p e o C_p tomam diferentes formas como é visível na Tabela 2.

Tipo de Condição de Fronteira	B_P	C_P
Temperatura imposta, T_b	$\frac{-k_{BP}a_n}{\delta r_{BP}}$	$\frac{T_B k_{BP} a_n}{\delta r_{BP}}$
Fluxo de calor imposto, q''_b	0	$\dot{q}''_B a_n$
Transferência de calor por convecção (h) com fluido exterior a uma temperatura de referência, T_F	$-\frac{1}{R}$	$\frac{T_F}{R}$

Tabela 2 – Termos da fonte linearizada para cada tipo de condição de fronteira (os índices utilizados são para a fronteira Norte, ver Figura 13).

Sendo,

$$k_{BP} \equiv \frac{1}{2}(k_B + k_P) \quad R \equiv \left[\frac{\delta r_{BP}}{k_{BP}} + \frac{1}{h} \right] a_n$$

a_n – área norte

Deste modo, o problema das condições de fronteira reduz-se ao cálculo dos coeficientes de B_P e de C_P das fontes fictícias (lembra-se o utilizador de que o B_P deverá ser sempre menor ou igual a zero).

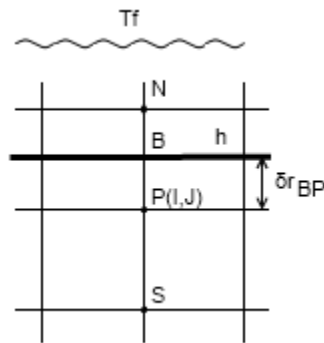


Figura 13 – Fronteira norte do corpo.

O procedimento para cada condição de fronteira é explicado de seguida.

i. Temperatura imposta

Como o método é igual para todas as fronteiras (Norte, Sul, Este, Oeste), só será analisada a fronteira norte.

O primeiro passo é cortar a ligação entre o ponto P e o ponto N, pois de P para N não existe fluxo de calor (existe entre P e B, ver Figura 13). Esta quebra é obtida impondo-se que o coeficiente AN(J) (em que J é a linha correspondente ao ponto P) seja nulo. Atendendo a que:

$$q_N = -\frac{T_P - T_B}{R}$$

$$\frac{1}{R} = \frac{K_{BP}^* A_n}{\delta r_{BP}}$$

Obtém-se,

$$BP = -\frac{1}{R} \qquad CP = \frac{T_B}{R}$$

No programa virá que:

```
%Fronteira Norte
RDYN=RV (NJ) / (YV (NJ) -Y (NJM1) ) ;
AN (NJM1) =0;                               (corte ligação)
DN=GAMH (IL,NJMI) *SEW (IL) *RDYN;          (valor de 1/R)
SU (NJM1) =SU (NJM1) +DN*T (IL,NJ) ;
SP (NJM1) =SP (NJM1) -DN;
```

ii. Fluxo de calor imposto

Se na fronteira houver um fluxo de calor, \ddot{q}_B , imposto tem-se:

$$A_n * \ddot{q}_B = q_N$$

De onde se retira:

$$B_P = 0$$

$$C_P = \ddot{q}_B * A_n$$

Em termos de programa (se na fronteira tivéssemos um fluxo de 20 W/m²) tem-se,

```
%Fronteira Norte
QB=20;
AN (NJM1) =0;
DN=QB*SEW (IL) *RV (NJ) ;
SU (NJM1) =SU (NJM1) +DN;
SP (NJM1) =SP (NJM1) ;
```

iii. Fronteira adiabática

Neste caso existem duas alternativas. A primeira consiste em quebrar a ligação impondo um fluxo nulo na fronteira (caindo no caso anterior, fazendo $Q_B=0$).

Outra alternativa é manter a ligação no ponto fronteiro N (não fazer $AN(NJM1) = 0$). Uma vez que o fluxo na fronteira é nulo, tem-se:

$$\ddot{q} = 0 = K \frac{dT}{dn} \Rightarrow T = cte$$

O que implica $T_B=T_P$. Em termos de programa tem de se escrever:

```
%Fronteira Norte  
T(IL,NJ)=T(IL,NJM1);
```

iv. Fronteira em contacto com um fluido à temperatura T_F com um coeficiente de convecção h

Em primeiro lugar é necessário cortar a ligação ao ponto vizinho N.

$$\ddot{q}_N = -\frac{T_P - T_F}{R} * A_N \qquad R = \frac{\delta r_{BP}}{K_{BP}} + \frac{1}{H}$$

Das equações anteriores obtém-se:

$$B_P = -\frac{a_n}{R} \qquad C_P = \frac{a_n * T_F}{R}$$

O código para implementar a fronteira a norte, supondo que $h=12.5 \text{ W/m}^2\text{K}$ e $T_F=80^\circ\text{C}$, é:

```
%Fronteira Norte  
H=12.5;  
TF=80;  
RDYN=YV(NJ)-Y(NJM1);  
NA(NJM1)=0;  
DN1=RDYN/GAMH(IL,NJM1)+1/H;  
DN=SEW(IL)*RV(NJ)/DN1;  
SU(NJM1)=SU(NJM1)+DN*TF;  
SP(NJM1)=SP(NJM1)-DN;
```

Chama-se a atenção para o facto de se ter de localizar, na Função PROMOD, os nós (ou o nó) onde se está a aplicar a condição de fronteira, no caso de se tratar de condições laterais. Repare-se nos “if” que se encontram no programa antes de se aplicarem as condições oeste e este (que só se aplicam com $IL=2$ e $IL=NIM1$).

v. Temperatura de uma zona do campo interior constante (T=TFIX)

Este é o caso, por exemplo, da temperatura do fluido de arrefecimento interno. O método é idêntico, criação de uma fonte fictícia:

$$S = B_p T_p + C_p$$

Para os nós em causa é necessário:

$$B_p = -GREAT$$

$$C_p = GREAT * TFIX$$

Em que *GREAT* é um número elevado (10^{30}).

Para terminar a explicação deste tipo de condição de fronteira, imagine-se que se pretende que o ponto P(10,10) estivesse sempre à temperatura de 60°C. No programa teria de se acrescentar as seguintes instruções na função PROMOD.

```
%Fronteira Interior
J=10;
TFIX=60;
BP=-GREAT;
CP=GREAT*TFIX;
SU(J)=SU(J)+CP;
SP(J)=SP(J)+BP;
```

Estas instruções permitiriam obter um valor de T(10,10) constante e aproximadamente igual a 60°C.

Resultados

Em seguida apresentam-se os resultados finais. Os resultados completos encontram-se no Anexo 3.

TEACH-C MATLAB:

```
CONDUCTION IN RECTANGULAR BAR WITH PRESCRIBED SURFACE TEMPERATURE

HEIGHT, H [M]----- 1.000
WEIGHT, W [M]----- 1.000
SPECIFIC HEAT, CV [J/KG.K]----- 485.670
THERMAL CONDUCTIVITY, TCON [W/M.K]----- 14.680
DENSIT, DENSIT [KG/M3]----- 7800.00
INITIAL TIME STEP, DT [S]----- 50.0
SOURCE NORMALIZATION FACTOR, SNORM----- 1.468E+003
NUMBER OF NODES IN X DIRECTION, NI----- 22
NUMBER OF NODES IN Y DIRECTION, NJ----- 22

-----TEMPERATURE (°C)-----
I= 1 2 3 4 5 6 7 8 Y
22 1.00E+000 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.0000
21 0.00E+000 4.70E+001 6.88E+001 7.40E+001 7.52E+001 7.55E+001 7.55E+001 7.55E+001 0.9750
20 0.00E+000 1.55E+001 2.97E+001 3.46E+001 3.60E+001 3.64E+001 3.65E+001 3.65E+001 0.9250
19 0.00E+000 4.95E+000 1.09E+001 1.33E+001 1.42E+001 1.44E+001 1.45E+001 1.45E+001 0.8750
18 0.00E+000 1.46E+000 3.46E+000 4.39E+000 4.73E+000 4.83E+000 4.86E+000 4.87E+000 0.8250
17 0.00E+000 3.94E+001 9.71E+001 1.26E+002 1.38E+002 1.41E+002 1.42E+002 1.43E+002 0.7750
16 0.00E+000 9.69E+002 2.45E+003 3.24E+003 3.57E+003 3.69E+003 3.72E+003 3.73E+003 0.7250
15 0.00E+000 2.19E+002 5.64E+002 7.58E+002 8.42E+002 8.72E+002 8.82E+002 8.84E+002 0.6750
14 0.00E+000 4.59E+003 1.20E+004 1.63E+004 1.82E+004 1.90E+004 1.92E+004 1.93E+004 0.6250
13 0.00E+000 8.99E+004 2.37E+005 3.26E+005 3.67E+005 3.83E+005 3.89E+005 3.90E+005 0.5750
12 0.00E+000 1.66E+004 4.42E+004 6.11E+004 6.94E+004 7.27E+004 7.38E+004 7.42E+004 0.5250
11 0.00E+000 2.90E+005 7.78E+005 1.08E+006 1.24E+006 1.30E+006 1.32E+006 1.33E+006 0.4750
10 0.00E+000 4.83E+006 1.30E+007 1.83E+007 2.10E+007 2.22E+007 2.26E+007 2.27E+007 0.4250
9 0.00E+000 7.70E+007 2.09E+008 2.95E+008 3.41E+008 3.60E+008 3.68E+008 3.70E+008 0.3750
8 0.00E+000 1.18E+007 3.22E+007 4.58E+007 5.30E+007 5.63E+007 5.75E+007 5.79E+007 0.3250
7 0.00E+000 1.75E+008 4.78E+008 6.83E+008 7.95E+008 8.46E+008 8.66E+008 8.73E+008 0.2750
6 0.00E+000 2.50E+009 6.87E+009 9.86E+009 1.15E+010 1.23E+010 1.26E+010 1.27E+010 0.2250
5 0.00E+000 3.47E+010 9.56E+010 1.38E+011 1.62E+011 1.73E+011 1.78E+011 1.80E+011 0.1750
4 0.00E+000 4.68E+011 1.29E+012 1.88E+012 2.21E+012 2.37E+012 2.44E+012 2.47E+012 0.1250
3 0.00E+000 6.14E+012 1.70E+013 2.48E+013 2.93E+013 3.16E+013 3.25E+013 3.29E+013 0.0750
2 0.00E+000 6.91E+013 1.92E+014 2.81E+014 3.33E+014 3.59E+014 3.70E+014 3.75E+014 0.0250
1 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.0000

X= 0.0000 0.0250 0.0750 0.1250 0.1750 0.2250 0.2750 0.3250

I= 9 10 11 12 13 14 15 16 Y
22 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.0000
21 7.55E+001 7.55E+001 7.55E+001 7.55E+001 7.55E+001 7.55E+001 7.55E+001 7.55E+001 0.9750
20 3.65E+001 3.65E+001 3.65E+001 3.65E+001 3.65E+001 3.65E+001 3.65E+001 3.65E+001 0.9250
19 1.45E+001 1.45E+001 1.45E+001 1.45E+001 1.45E+001 1.45E+001 1.45E+001 1.45E+001 0.8750
18 4.87E+000 4.87E+000 4.87E+000 4.87E+000 4.87E+000 4.87E+000 4.87E+000 4.87E+000 0.8250
17 1.43E+000 1.43E+000 1.43E+000 1.43E+000 1.43E+000 1.43E+000 1.43E+000 1.43E+000 0.7750
16 3.73E+001 3.73E+001 3.73E+001 3.73E+001 3.73E+001 3.73E+001 3.73E+001 3.73E+001 0.7250
15 8.85E+002 8.85E+002 8.85E+002 8.85E+002 8.85E+002 8.85E+002 8.85E+002 8.85E+002 0.6750
14 1.93E+002 1.93E+002 1.93E+002 1.93E+002 1.93E+002 1.93E+002 1.93E+002 1.93E+002 0.6250
13 3.91E+003 3.91E+003 3.91E+003 3.91E+003 3.91E+003 3.91E+003 3.91E+003 3.91E+003 0.5750
12 7.43E+004 7.43E+004 7.43E+004 7.43E+004 7.43E+004 7.43E+004 7.43E+004 7.43E+004 0.5250
11 1.33E+004 1.33E+004 1.33E+004 1.33E+004 1.33E+004 1.33E+004 1.33E+004 1.33E+004 0.4750
10 2.28E+005 2.28E+005 2.28E+005 2.28E+005 2.28E+005 2.28E+005 2.28E+005 2.28E+005 0.4250
9 3.71E+006 3.71E+006 3.71E+006 3.71E+006 3.71E+006 3.71E+006 3.71E+006 3.71E+006 0.3750
8 5.81E+007 5.81E+007 5.81E+007 5.81E+007 5.81E+007 5.81E+007 5.81E+007 5.81E+007 0.3250
7 8.76E+008 8.76E+008 8.76E+008 8.76E+008 8.76E+008 8.76E+008 8.76E+008 8.76E+008 0.2750
6 1.28E+009 1.28E+009 1.28E+009 1.28E+009 1.28E+009 1.28E+009 1.28E+009 1.28E+009 0.2250
5 1.80E+009 1.80E+009 1.80E+009 1.80E+009 1.80E+009 1.80E+009 1.80E+009 1.80E+009 0.1750
4 2.47E+010 2.47E+010 2.47E+010 2.47E+010 2.47E+010 2.47E+010 2.47E+010 2.47E+010 0.1250
3 3.30E+011 3.30E+011 3.31E+011 3.31E+011 3.30E+011 3.30E+011 3.29E+011 3.25E+011 0.0750
2 3.76E+012 3.77E+012 3.77E+012 3.77E+012 3.76E+012 3.76E+012 3.75E+012 3.70E+012 0.0250
1 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.0000

X= 0.3750 0.4250 0.4750 0.5250 0.5750 0.6250 0.6750 0.7250

I= 17 18 19 20 21 22 Y
22 1.00E+002 1.00E+002 1.00E+002 1.00E+002 1.00E+002 0.00E+000 1.0000
21 7.55E+001 7.52E+001 7.40E+001 6.88E+001 4.70E+001 0.00E+000 0.9750
20 3.64E+001 3.60E+001 3.46E+001 2.97E+001 1.55E+001 0.00E+000 0.9250
19 1.44E+001 1.42E+001 1.33E+001 1.09E+001 4.95E+000 0.00E+000 0.8750
18 4.83E+000 4.73E+000 4.39E+000 3.46E+000 1.46E+000 0.00E+000 0.8250
17 1.41E+000 1.38E+000 1.26E+000 9.71E+001 3.94E+001 0.00E+000 0.7750
16 3.69E+001 3.57E+001 3.24E+001 2.45E+001 9.69E+002 0.00E+000 0.7250
15 8.72E+002 8.42E+002 7.58E+002 5.64E+002 2.19E+002 0.00E+000 0.6750
14 1.90E+002 1.82E+002 1.63E+002 1.20E+002 4.59E+003 0.00E+000 0.6250
13 3.83E+003 3.67E+003 3.26E+003 2.37E+003 8.99E+004 0.00E+000 0.5750
12 7.27E+004 6.94E+004 6.11E+004 4.42E+004 1.66E+004 0.00E+000 0.5250
11 1.30E+004 1.24E+004 1.09E+004 7.78E+005 2.90E+005 0.00E+000 0.4750
10 2.22E+005 2.10E+005 1.83E+005 1.30E+005 4.83E+006 0.00E+000 0.4250
9 3.60E+006 3.41E+006 2.95E+006 2.09E+006 7.70E+007 0.00E+000 0.3750
8 5.63E+007 5.30E+007 4.58E+007 3.22E+007 1.18E+007 0.00E+000 0.3250
7 8.46E+008 7.95E+008 6.83E+008 4.78E+008 1.75E+008 0.00E+000 0.2750
6 1.23E+009 1.15E+009 9.86E+009 6.87E+009 2.50E+009 0.00E+000 0.2250
5 1.73E+009 1.62E+009 1.38E+009 9.56E+010 3.47E+010 0.00E+000 0.1750
4 2.37E+010 2.21E+010 1.88E+010 1.29E+010 4.68E+011 0.00E+000 0.1250
3 3.16E+011 2.93E+011 2.48E+011 1.70E+011 6.14E+012 0.00E+000 0.0750
2 3.59E+012 3.33E+012 2.81E+012 1.92E+012 6.91E+013 0.00E+000 0.0250
1 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.0000

X= 0.7750 0.8250 0.8750 0.9250 0.9750 1.0000
```

Gráfico do campo de temperaturas:

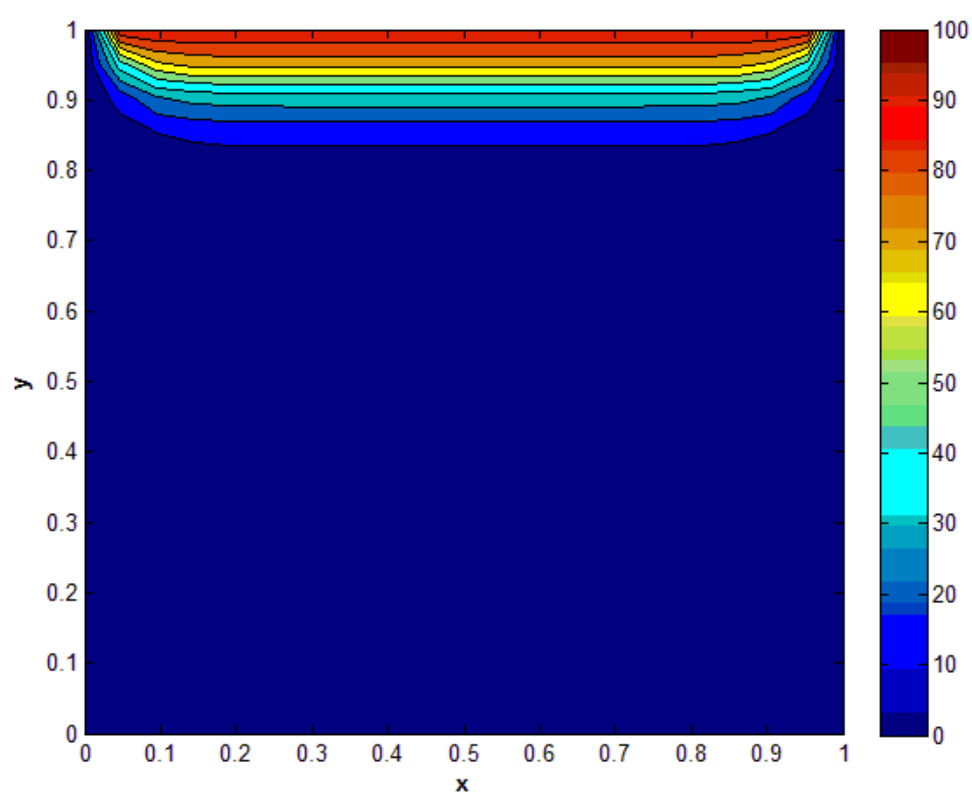


Figura 14 – Campo de Temperaturas final.

TEACH-C FORTRAN:

Verifica-se que os resultados obtidos pela versão em MATLAB e em FORTRAN são idênticos.

[illegible]

2-Convecção e simetrias

Neste problema considera-se uma barra em que em duas fronteiras (Oeste e Sul) é aplicado uma transferência de calor por convecção e nas outras duas são aplicadas condições de fronteira de simetria. Os dados deste problema são os seguintes:

IT	102	TCOND [W/m.K]	52	NITPRI	110
JY	12	CV [J/kg.K]	460	NSTPRI	1
NI	102	DENSIT [kg/m ³]	7850	DT [s]	30
NJ	12	TF [°C]	80	CFN	4
W [m]	1	TINIC [°C]	800	CFS	3
H [m]	0.1	h [W/m ² K]	520	CFW	3
IMON	101	MAXIT	100	CFE	4
JMON	11	MAXSTP	100		

Tabela 3 – Dados do problema.

É necessário efectuarem-se alterações no programa principal mas como estas são bastante simples não serão explicadas. No CD do ANEXO 3 está o código com essas pequenas alterações.

Resultados

TEACH-C MATLAB:

Como a malha tem bastantes nós, o campo de temperaturas é muito grande para introduzir nesta secção. Este ficheiro está no ANEXO 3.

Verifica-se que este campo é igual ao calculado com a versão FORTRAN o que valida o código.

Gráfico de Temperaturas:

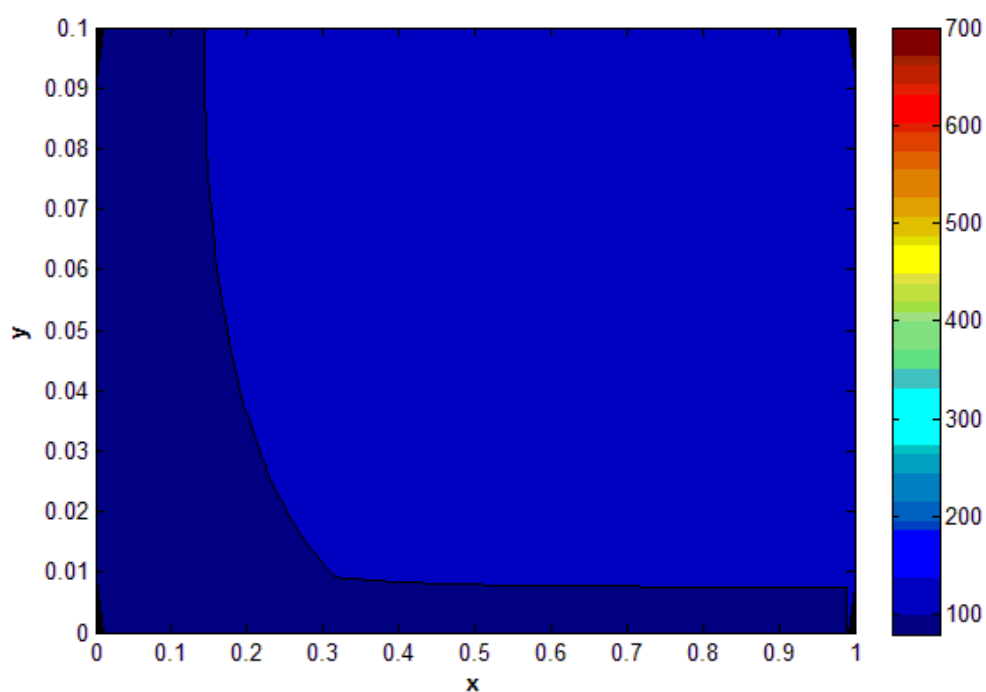


Figura 15 – Campo de Temperaturas final.

TEACH-C FORTRAN:

Pelas razões já apontadas, o campo de Temperaturas não está representado, podendo ser consultado no ANEXO 3.

3-Distribuição de Temperaturas em barra de combustível nuclear

Neste problema considera-se um cilindro (barra de combustível nuclear, Dióxido de Urânio) em regime estacionário que está a produzir energia. Assim, a fronteira Este e Sul contêm planos de simetria e existe uma fonte.

Em seguida apresentam-se somente os dados do problema que são diferentes do caso de temperatura imposta (caso base):

INCYLY	1	IMON	6	CV	334.2
NI	12	JMON	4	DT	0
NJ	8	TCON	9.174	INTIME	0
W	2	DENSIT	1096	CFS	4
H	0.1	TINIC	600	CFE	4
S	5e7				

Tabela 4 – Dados do problema diferentes do caso base.

São necessárias as seguintes modificações no código. No Anexo 3 podem-se visualizar as linhas de código introduzidas para este caso.

Função	Alterações
Principal	Inserir S e W como variável da função CALC
	Normalizar Temperaturas
	INCYLY=1
	Modificar NI, NJ, H, W e S
	Modificar DT e INTIME
	Inicializar campo de temperaturas, TINIC
	Calcular SNORM (factor de normalização da fonte)
	Definir Temperaturas ao longo do eixo de simetria
	Calcular e imprimir as temperaturas adimensionalizadas
	Formatar o “output”
	Formatar Gráfico
PROMOD	Tornar Fronteira Sul e Oeste como planos de simetria
	Inserir a fonte de calor
PRINT	Modificar para geometria radial

Tabela 5 – Modificações a efectuar no código base.

Resultados

TEACH-C MATLAB:

É de referir que o campo final de temperaturas está adimensionalizado.

```

AXISIMETRIC CONDUCTION WITH HEAT SOURCE

RADIUS, H [M]-----> 0.100
WEIGHT, W [M]-----> 2.000
SPECIFIC HEAT, CV [J/KG.K]-----> 334.200
THERMAL CONDUCTIVITY, TCON [W/M.K]-----> 9.174
DENSIT, DENSIT [KG/M3]-----> 1096.00
INITIAL TIME STEP, DT [s]-----> 0.0
SOURCE NORMALIZATION FACTOR, SNORM-----> 5.000E+004
NUMBER OF NODES IN X DIRECTION, NI-----> 12
NUMBER OF NODES IN Y DIRECTION, NJ-----> 8

-----TEMPERATURE (°C)-----
I= 1 2 3 4 5 6 7 8 R
J 8 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.1000
7 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0917
6 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0750
5 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0583
4 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0417
3 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0250
2 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0083
1 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0000

x= 0.0000 0.1000 0.3000 0.5000 0.7000 0.9000 1.1000 1.3000

I= 9 10 11 12 R
J 8 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.1000
7 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0917
6 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0750
5 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0583
4 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0417
3 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0250
2 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0083
1 6.00E+002 6.00E+002 6.00E+002 6.00E+002 0.0000

x= 1.5000 1.7000 1.9000 2.0000

NITER SOURCE T(6,4) TIME(s) DT(s) NSTEP
1 6.537E-002 6.700E+002 0 0 1
2 2.019E-003 6.735E+002 0 0 1
3 7.789E-005 6.737E+002 0 0 1

-----TEMPERATURE (°C)-----
I= 1 2 3 4 5 6 7 8 R
J 8 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.1000
7 0.00E+000 1.31E-002 3.90E-002 6.40E-002 8.74E-002 1.09E-001 1.27E-001 1.43E-001 0.0917
6 0.00E+000 3.50E-002 1.04E-001 1.71E-001 2.33E-001 2.90E-001 3.39E-001 3.80E-001 0.0750
5 0.00E+000 5.25E-002 1.56E-001 2.56E-001 3.49E-001 4.34E-001 5.09E-001 5.70E-001 0.0583
4 0.00E+000 6.56E-002 1.95E-001 3.20E-001 4.37E-001 5.43E-001 6.36E-001 7.13E-001 0.0417
3 0.00E+000 7.43E-002 2.21E-001 3.62E-001 4.95E-001 6.15E-001 7.20E-001 8.08E-001 0.0250
2 0.00E+000 7.87E-002 2.34E-001 3.84E-001 5.24E-001 6.51E-001 7.63E-001 8.55E-001 0.0083
1 0.00E+000 7.87E-002 2.34E-001 3.84E-001 5.24E-001 6.51E-001 7.63E-001 8.55E-001 0.0000

x= 0.0000 0.1000 0.3000 0.5000 0.7000 0.9000 1.1000 1.3000

I= 9 10 11 12 R
J 8 0.00E+000 0.00E+000 0.00E+000 0.00E+000 0.1000
7 1.55E-001 1.63E-001 1.67E-001 1.67E-001 0.0917
6 4.12E-001 4.34E-001 4.45E-001 4.45E-001 0.0750
5 6.18E-001 6.50E-001 6.67E-001 6.67E-001 0.0583
4 7.72E-001 8.13E-001 8.33E-001 8.33E-001 0.0417
3 8.75E-001 9.21E-001 9.44E-001 9.44E-001 0.0250
2 9.27E-001 9.75E-001 1.00E+000 1.00E+000 0.0083
1 9.27E-001 9.75E-001 1.00E+000 1.00E+000 0.0000

x= 1.5000 1.7000 1.9000 2.0000

Fonte normalizada= -90.988069

```

Gráfico do campo de temperaturas:

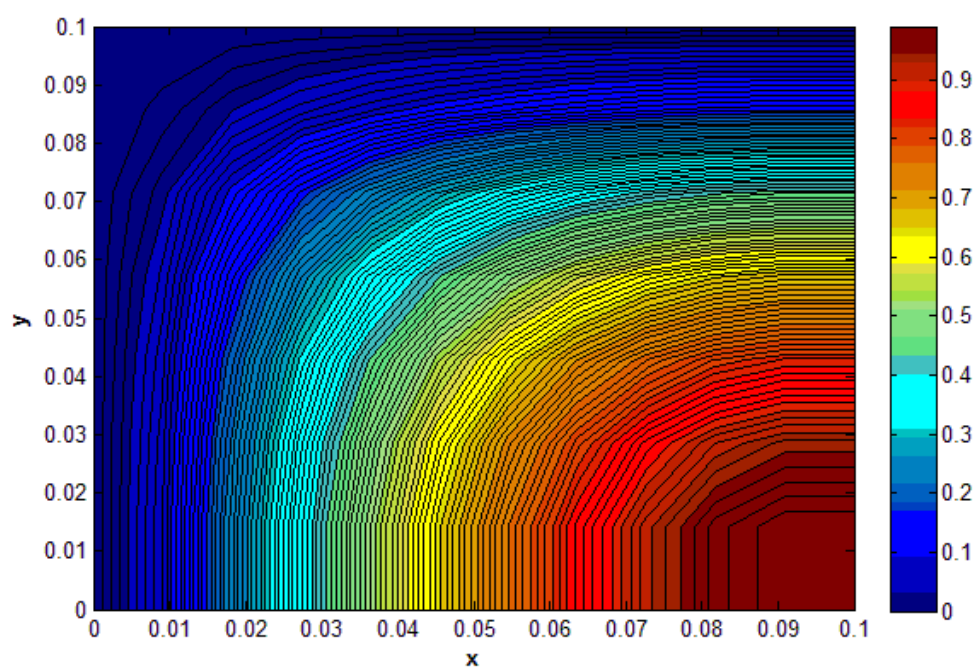


Figura 16 – Campo de temperaturas adimensionalizado.

TEACH-C FORTRAN:

Este teste em FORTRAN foi efectuado num sistemas operativo Linux com o editor de texto VI por isso podem existir problemas na abertura dos ficheiros que estão no Anexo 3. É de referir que apesar de o resíduo não ser igual (ver Anexo 3 para o caso FORTRAN), os resultados finais para o campo de temperaturas coincidem. Isto deve-se ao facto das duas máquinas e dos softwares utilizados (MATLAB e FORTRAN) terem precisões diferentes.

	TEMPERATURE (C)								
I =	1	2	3	4	5	6	7	8	R
J									
8	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.1000
7	0.000E+00	1.310E-02	3.900E-02	6.400E-02	8.740E-02	1.090E-01	1.270E-01	1.430E-01	0.0917
6	0.000E+00	3.500E-02	1.040E-01	1.710E-01	2.330E-01	2.900E-01	3.390E-01	3.800E-01	0.0750
5	0.000E+00	5.250E-02	1.560E-01	2.560E-01	3.490E-01	4.340E-01	5.090E-01	5.700E-01	0.0583
4	0.000E+00	6.560E-02	1.950E-01	3.200E-01	4.370E-01	5.430E-01	6.360E-01	7.130E-01	0.0417
3	0.000E+00	7.430E-02	2.210E-01	3.630E-01	4.950E-01	6.150E-01	7.200E-01	8.080E-01	0.0250
2	0.000E+00	7.870E-02	2.340E-01	3.840E-01	5.240E-01	6.510E-01	7.630E-01	8.550E-01	0.0083
1	0.000E+00	7.870E-02	2.340E-01	3.840E-01	5.240E-01	6.510E-01	7.630E-01	8.550E-01	0.0000
X	0.00	0.10	0.30	0.50	0.70	0.90	1.10	1.30	
I =	9	10	11	12	R				
J									
8	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.1000				
7	1.550E-01	1.630E-01	1.670E-01	1.670E-01	0.0917				
6	4.120E-01	4.340E-01	4.450E-01	4.450E-01	0.0750				
5	6.180E-01	6.500E-01	6.670E-01	6.670E-01	0.0583				
4	7.720E-01	8.130E-01	8.330E-01	8.330E-01	0.0417				
3	8.750E-01	9.210E-01	9.440E-01	9.440E-01	0.0250				
2	9.270E-01	9.750E-01	1.000E+00	1.000E+00	0.0083				
1	9.270E-01	9.750E-01	1.000E+00	1.000E+00	0.0000				
X	1.50	1.70	1.90	2.00					
NORMALIZED SOURCE= -9.099E+01									

4-Fluxo de Calor imposto

O último caso testado é quando o corpo tem na fronteira norte um fluxo de calor imposto. Este caso é muito semelhante ao caso base sendo as diferenças apresentadas na seguinte tabela.

INTIME	0	CFW	1	SNORM	Q''
Q''	1000	CFE	1	MAXIT	3000
CFN	2	IMON	12	SNORMAX	0.0001
CFS	1	JMON	20		

Tabela 6 – Dados do problema.

Em termos de código, para além de se modificar os dados da tabela, apenas é necessário introduzir a variável Q na Função CALC e PROMOD, modificar o cabeçalho dos resultados e o número de cores do gráfico.

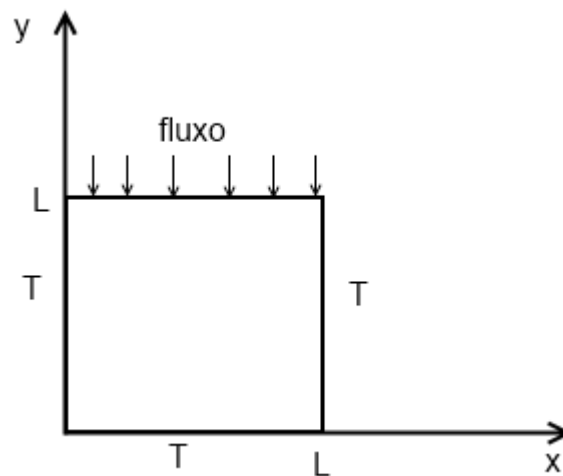


Figura 17 – Corpo e condições de fronteira aplicadas, $T=0^{\circ}\text{C}$.

Resultados:

TEACH-C MATLAB:

-----TEMPERATURE (°C)-----									
I=	1	2	3	4	5	6	7	8	Y
22	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	1.0000
21	0.00E+000	3.65E+000	8.78E+000	1.27E+001	1.57E+001	1.81E+001	2.00E+001	2.14E+001	0.9750
20	0.00E+000	2.42E+000	6.63E+000	1.01E+001	1.29E+001	1.52E+001	1.70E+001	1.84E+001	0.9250
19	0.00E+000	1.81E+000	5.21E+000	8.19E+000	1.07E+001	1.28E+001	1.45E+001	1.58E+001	0.8750
18	0.00E+000	1.44E+000	4.21E+000	6.72E+000	8.92E+000	1.08E+001	1.23E+001	1.35E+001	0.8250
17	0.00E+000	1.17E+000	3.46E+000	5.58E+000	7.47E+000	9.10E+000	1.05E+001	1.15E+001	0.7750
16	0.00E+000	9.71E+001	2.87E+000	4.66E+000	6.28E+000	7.69E+000	8.88E+000	9.84E+000	0.7250
15	0.00E+000	8.11E+001	2.40E+000	3.91E+000	5.29E+000	6.51E+000	7.54E+000	8.38E+000	0.6750
14	0.00E+000	6.80E+001	2.02E+000	3.29E+000	4.46E+000	5.50E+000	6.40E+000	7.12E+000	0.6250
13	0.00E+000	5.71E+001	1.70E+000	2.77E+000	3.76E+000	4.65E+000	5.42E+000	6.05E+000	0.5750
12	0.00E+000	4.80E+001	1.43E+000	2.33E+000	3.17E+000	3.93E+000	4.58E+000	5.11E+000	0.5250
11	0.00E+000	4.02E+001	1.20E+000	1.96E+000	2.66E+000	3.30E+000	3.85E+000	4.31E+000	0.4750
10	0.00E+000	3.36E+001	9.99E+001	1.63E+000	2.23E+000	2.76E+000	3.23E+000	3.61E+000	0.4250
9	0.00E+000	2.78E+001	8.28E+001	1.36E+000	1.85E+000	2.29E+000	2.68E+000	3.00E+000	0.3750
8	0.00E+000	2.28E+001	6.79E+001	1.11E+000	1.52E+000	1.88E+000	2.20E+000	2.47E+000	0.3250
7	0.00E+000	1.84E+001	5.48E+001	8.98E+001	1.22E+000	1.52E+000	1.78E+000	1.99E+000	0.2750
6	0.00E+000	1.45E+001	4.31E+001	7.06E+001	9.63E+001	1.20E+000	1.40E+000	1.57E+000	0.2250
5	0.00E+000	1.09E+001	3.24E+001	5.31E+001	7.25E+001	9.01E+001	1.05E+000	1.18E+000	0.1750
4	0.00E+000	7.59E+002	2.26E+001	3.70E+001	5.05E+001	6.28E+001	7.35E+001	8.23E+001	0.1250
3	0.00E+000	4.48E+002	1.33E+001	2.19E+001	2.98E+001	3.71E+001	4.34E+001	4.86E+001	0.0750
2	0.00E+000	1.48E+002	4.41E+002	7.22E+002	9.86E+002	1.22E+001	1.43E+001	1.61E+001	0.0250
1	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.0000
x=	0.0000	0.0250	0.0750	0.1250	0.1750	0.2250	0.2750	0.3250	
I=	9	10	11	12	13	14	15	16	Y
22	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	1.0000
21	2.25E+001	2.32E+001	2.35E+001	2.35E+001	2.32E+001	2.25E+001	2.14E+001	2.00E+001	0.9750
20	1.94E+001	2.01E+001	2.04E+001	2.04E+001	2.01E+001	1.94E+001	1.84E+001	1.70E+001	0.9250
19	1.68E+001	1.74E+001	1.77E+001	1.77E+001	1.74E+001	1.68E+001	1.58E+001	1.45E+001	0.8750
18	1.44E+001	1.50E+001	1.53E+001	1.53E+001	1.50E+001	1.44E+001	1.35E+001	1.23E+001	0.8250
17	1.23E+001	1.29E+001	1.31E+001	1.31E+001	1.29E+001	1.23E+001	1.15E+001	1.05E+001	0.7750
16	1.06E+001	1.10E+001	1.13E+001	1.13E+001	1.10E+001	1.06E+001	9.84E+000	8.88E+000	0.7250
15	9.01E+000	9.44E+000	9.65E+000	9.65E+000	9.44E+000	9.01E+000	8.38E+000	7.54E+000	0.6750
14	7.68E+000	8.05E+000	8.24E+000	8.24E+000	8.05E+000	7.68E+000	7.12E+000	6.40E+000	0.6250
13	6.52E+000	6.85E+000	7.01E+000	7.01E+000	6.85E+000	6.52E+000	6.05E+000	5.42E+000	0.5750
12	5.52E+000	5.80E+000	5.94E+000	5.94E+000	5.80E+000	5.52E+000	5.11E+000	4.58E+000	0.5250
11	4.66E+000	4.90E+000	5.01E+000	5.01E+000	4.90E+000	4.66E+000	4.31E+000	3.85E+000	0.4750
10	3.91E+000	4.11E+000	4.21E+000	4.21E+000	4.11E+000	3.91E+000	3.61E+000	3.23E+000	0.4250
9	3.25E+000	3.42E+000	3.50E+000	3.50E+000	3.42E+000	3.25E+000	3.00E+000	2.68E+000	0.3750
8	2.67E+000	2.81E+000	2.88E+000	2.88E+000	2.81E+000	2.67E+000	2.47E+000	2.20E+000	0.3250
7	2.16E+000	2.27E+000	2.33E+000	2.33E+000	2.27E+000	2.16E+000	1.99E+000	1.78E+000	0.2750
6	1.70E+000	1.79E+000	1.83E+000	1.83E+000	1.79E+000	1.70E+000	1.57E+000	1.40E+000	0.2250
5	1.28E+000	1.35E+000	1.38E+000	1.38E+000	1.35E+000	1.28E+000	1.18E+000	1.05E+000	0.1750
4	8.92E+001	9.38E+001	9.62E+001	9.62E+001	9.38E+001	8.92E+001	8.23E+001	7.35E+001	0.1250
3	5.26E+001	5.54E+001	5.68E+001	5.68E+001	5.54E+001	5.26E+001	4.86E+001	4.34E+001	0.0750
2	1.74E+001	1.83E+001	1.88E+001	1.88E+001	1.83E+001	1.74E+001	1.61E+001	1.43E+001	0.0250
1	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.0000
x=	0.3750	0.4250	0.4750	0.5250	0.5750	0.6250	0.6750	0.7250	
I=	17	18	19	20	21	22	Y		
22	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	1.0000		
21	1.81E+001	1.57E+001	1.27E+001	8.78E+000	3.65E+000	0.00E+000	0.9750		
20	1.52E+001	1.29E+001	1.01E+001	6.63E+000	2.42E+000	0.00E+000	0.9250		
19	1.28E+001	1.07E+001	8.19E+000	5.21E+000	1.81E+000	0.00E+000	0.8750		
18	1.08E+001	8.92E+000	6.72E+000	4.21E+000	1.44E+000	0.00E+000	0.8250		
17	9.10E+000	7.47E+000	5.58E+000	3.46E+000	1.17E+000	0.00E+000	0.7750		
16	7.69E+000	6.28E+000	4.66E+000	2.87E+000	9.71E+001	0.00E+000	0.7250		
15	6.51E+000	5.29E+000	3.91E+000	2.40E+000	8.11E+001	0.00E+000	0.6750		
14	5.50E+000	4.46E+000	3.29E+000	2.02E+000	6.80E+001	0.00E+000	0.6250		
13	4.65E+000	3.76E+000	2.77E+000	1.70E+000	5.71E+001	0.00E+000	0.5750		
12	3.93E+000	3.17E+000	2.33E+000	1.43E+000	4.80E+001	0.00E+000	0.5250		
11	3.30E+000	2.66E+000	1.96E+000	1.20E+000	4.02E+001	0.00E+000	0.4750		
10	2.76E+000	2.23E+000	1.63E+000	9.99E+001	3.36E+001	0.00E+000	0.4250		
9	2.29E+000	1.85E+000	1.36E+000	8.28E+001	2.79E+001	0.00E+000	0.3750		
8	1.88E+000	1.52E+000	1.11E+000	6.79E+001	2.28E+001	0.00E+000	0.3250		
7	1.52E+000	1.22E+000	8.98E+001	5.48E+001	1.84E+001	0.00E+000	0.2750		
6	1.20E+000	9.63E+001	7.06E+001	4.31E+001	1.45E+001	0.00E+000	0.2250		
5	9.01E+001	7.25E+001	5.31E+001	3.24E+001	1.09E+001	0.00E+000	0.1750		
4	6.28E+001	5.05E+001	3.70E+001	2.26E+001	7.60E+002	0.00E+000	0.1250		
3	3.71E+001	2.98E+001	2.19E+001	1.33E+001	4.48E+002	0.00E+000	0.0750		
2	1.22E+001	9.86E+002	7.22E+002	4.41E+002	1.48E+002	0.00E+000	0.0250		
1	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.00E+000	0.0000		
x=	0.7750	0.8250	0.8750	0.9250	0.9750	1.0000			

Gráfico do campo de temperaturas:

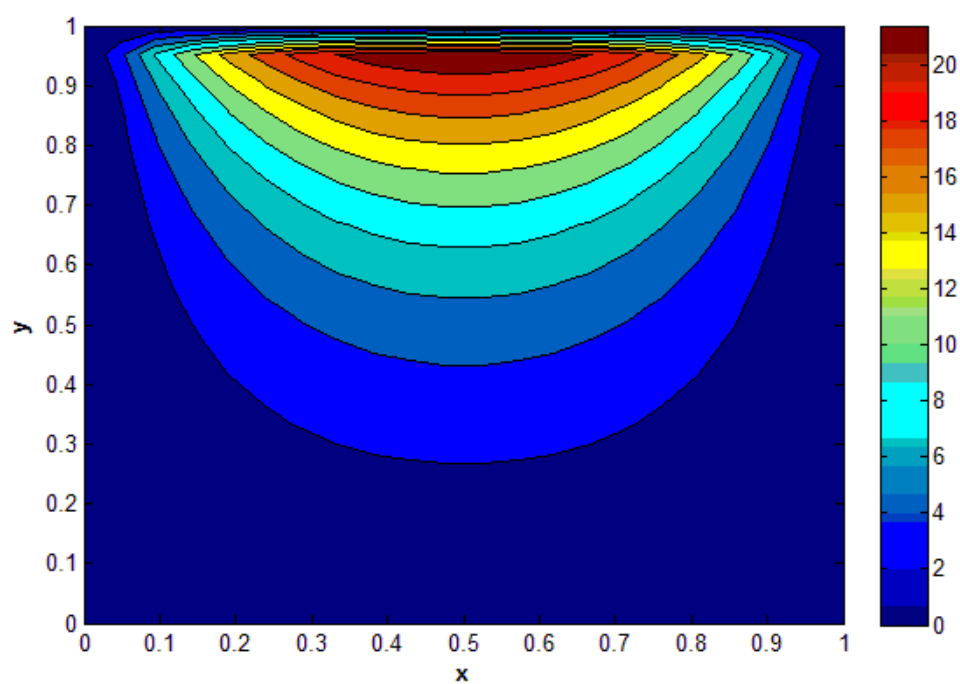


Figura 18 – Campo de temperaturas.

Solução Analítica:

Como não existia solução testada em FORTRAN para o caso em que se tem um fluxo de calor numa das fronteiras, calculou-se uma expressão para o campo de temperaturas do corpo através do método de separação de variáveis em função da posição. Tal permitiu verificar a validade do código.

Para uma placa homogénea, bidimensional com, pelo menos, 3 das 4 condições de fronteira homogéneas, a equação de Laplace diz:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$$

As condições de fronteira são:

$$\begin{array}{llll} T(0, y) = 0 & 0 < y < L & T(x, 0) = 0 & 0 < x < L \\ T(L, y) = 0 & 0 < y < L & \left. \frac{\partial T}{\partial y} \right|_{y=L} = q'' & 0 < x < L \end{array}$$

Como a equação de Laplace é linear e homogénea e três das condições de fronteira são homogéneas pode assumir-se uma solução do tipo,

$$T(x, y) = X(x)Y(y)$$

Assim,

$$Y \frac{d^2 X}{dx^2} + X \frac{d^2 Y}{dy^2} = 0 \Leftrightarrow -\frac{1}{X} \frac{d^2 X}{dx^2} = -\frac{1}{Y} \frac{d^2 Y}{dy^2}$$

Em que o primeiro membro depende apenas de x e o segundo apenas de y. Como X e Y são independentes, então esta igualdade só se verificará se ambos os termos forem iguais a uma constante, γ .

$$\frac{d^2 X}{dx^2} + \gamma X = 0$$

$$\frac{d^2 Y}{dy^2} - \gamma Y = 0$$

Em que $\gamma = \lambda^2 > 0$ para existir solução para o problema:

$$\frac{d^2 X}{dx^2} + \lambda^2 X = 0$$

$$\frac{d^2 Y}{dy^2} - \lambda^2 Y = 0$$

A solução destas equações é:

$$X = A \sin(\lambda x) + B \cos(\lambda x)$$

$$Y = C \sinh(\lambda y) + D \cosh(\lambda y)$$

$$T(x, y) = [A \sin(\lambda x) + B \cos(\lambda x)][C \sinh(\lambda y) + D \cosh(\lambda y)]$$

Aplicando as seguintes condições de fronteira,

$$T(0, y) = B \cos(\lambda y) (C \sinh(\lambda y) + D \cosh(\lambda y)) = 0 \Rightarrow B = 0$$

$$T(x, 0) = A \sin(\lambda y) D = 0 \Rightarrow D = 0 \text{ (Dado que } A \text{ e } B \text{ não podem ser ambos nulos)}$$

Assim obtém-se,

$$T(x, y) = E \sin(\lambda x) \sinh(\lambda y), E = AC$$

Aplicando a outra condição de fronteira de primeira espécie tem-se:

$$T(L, y) = E \sin(\lambda L) \sinh(\lambda y) = 0 \Rightarrow \sin(\lambda L) = 0, \text{ (para evitar } E = 0)$$

$$\lambda_n = \frac{n\pi}{L}, n \neq 0$$

obtendo-se:

$$T(x, y) = \sum_{n=1}^{\infty} E \sin(\lambda_n x) \sinh(\lambda_n y) \quad (2)$$

Derivando a equação das temperaturas segundo y resulta que:

$$\left. \frac{\partial T}{\partial y} \right|_{y=L} = \sum_{n=1}^{\infty} E \lambda_n \sin(\lambda_n x) \cosh(n\pi)$$

E juntando a condição de fronteira de 2ª espécie,

$$\frac{q''}{k} = \sum_{n=1}^{\infty} G \sin\left(\frac{n\pi x}{L}\right) \quad G = E \lambda_n \cosh(n\pi)$$

obtendo-se

$$E = 2 \frac{q'' L}{k} \frac{(-1)^{n+1} + 1}{n^2 \pi^2 \cosh(n\pi)} \quad (3)$$

Introduzindo-se esta ultima equação em (1), obtém-se a equação do campo de temperaturas no corpo quando sobre uma das fronteiras incide um fluxo de calor e nas outras três a temperatura é zero.

$$T(x, y) = \sum_{n=1}^{\infty} 2 \frac{q'' L}{k} \frac{(-1)^{n+1} + 1}{n^2 \pi^2 \cosh(n\pi)} \sin\left(\frac{n\pi x}{L}\right) \sinh\left(\frac{n\pi y}{L}\right)$$

Substituindo L, k, q'' e n e introduzindo esta equação no MATLAB obtém-se o campo de temperaturas em função da posição.

Na figura 18 apresenta-se o gráfico do campo de temperaturas e os respectivos valores do campo para uma série de apenas um termo. É de referir que só foram calculados os pontos interiores do corpo pois com o Programa TEACH-C os contornos deste têm valores pré definidos pelas condições iniciais ou de fronteira.

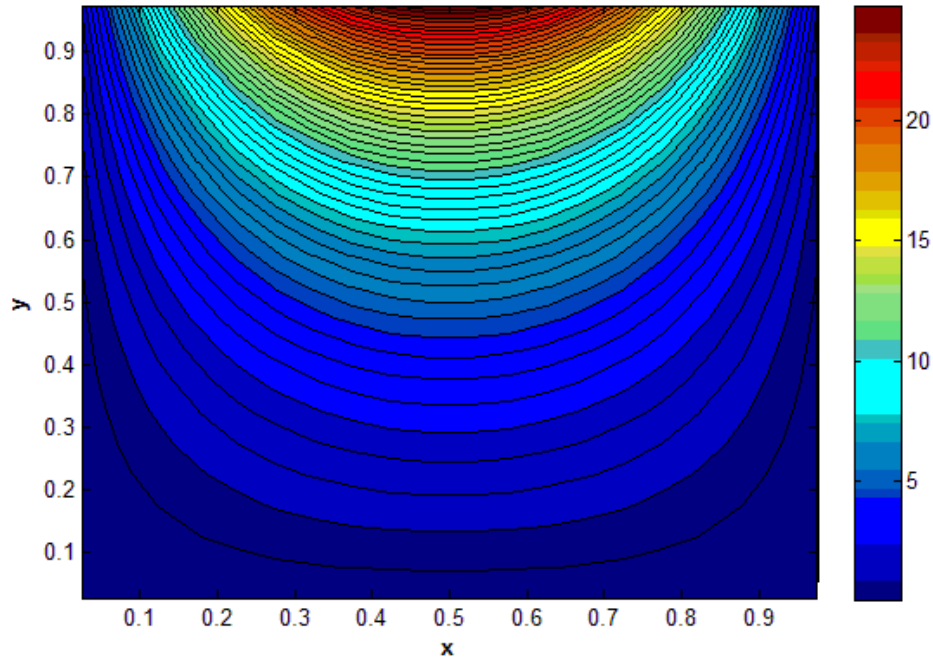


Figura 18 – Campo de temperaturas.

J\i	1	2	3	4	5	6	7	8
20	1,99438	5,93403	9,727566	13,28158	16,50855	19,32903	21,67356	23,48442
19	1,703091	5,067337	8,306808	11,34174	14,0974	16,50593	18,50803	20,05441
18	1,453911	4,325932	7,091435	9,682322	12,0348	14,09094	15,80011	17,12024
17	1,240678	3,691485	6,051395	8,2623	10,26976	12,02434	13,48285	14,60936
16	1,058121	3,148309	5,160975	7,046561	8,758637	10,25505	11,49894	12,4597
15	0,901726	2,682974	4,398159	6,005047	7,46407	8,739303	9,799346	10,6181
14	0,767626	2,283976	3,744087	5,112006	6,354051	7,439637	8,342036	9,039025
13	0,652505	1,941448	3,182586	4,345359	5,401134	6,323915	7,090981	7,683443
12	0,553517	1,646922	2,699774	3,686149	4,581759	5,36455	6,015249	6,517832
11	0,468215	1,393116	2,283714	3,118079	3,875667	4,537823	5,088243	5,513373
10	0,394489	1,173754	1,924118	2,627103	3,2654	3,823292	4,287042	4,645231
9	0,330517	0,983414	1,612095	2,201081	2,735869	3,203292	3,591838	3,891942
8	0,274717	0,817387	1,339931	1,829481	2,273983	2,662492	2,985441	3,234879
7	0,22571	0,671571	1,100896	1,503114	1,86832	2,187521	2,452859	2,657799
6	0,182283	0,542359	0,889081	1,213911	1,50885	1,766637	1,980923	2,146432
5	0,143362	0,426557	0,699248	0,954722	1,186687	1,389432	1,557965	1,688135
4	0,107987	0,321301	0,526704	0,719138	0,893864	1,046581	1,173527	1,271577
3	0,075281	0,22399	0,367183	0,501335	0,623142	0,729606	0,818104	0,886458
2	0,044437	0,132216	0,21674	0,295927	0,367827	0,430671	0,482909	0,523257
1	0,014691	0,043712	0,071656	0,097836	0,121607	0,142383	0,159654	0,172993

J\I	9	10	11	12	13	14	15	16
20	24,71702	25,34099	25,34099	24,71702	2,35E+01	2,17E+01	19,32903	16,50855
19	21,10698	21,63982	21,63982	21,10698	2,01E+01	1,85E+01	16,50593	14,0974
18	18,0188	18,47369	18,47369	18,0188	1,71E+01	1,58E+01	14,09094	12,0348
17	15,37614	15,76431	15,76431	15,37614	1,46E+01	1,35E+01	12,02434	10,26976
16	13,11365	13,4447	13,4447	13,11365	1,25E+01	1,15E+01	10,25505	8,758637
15	11,17539	11,45752	11,45752	11,17539	1,06E+01	9,80E+00	8,739303	7,46407
14	9,513444	9,75361	9,75361	9,513444	9,04E+00	8,34E+00	7,439637	6,354051
13	8,086713	8,290862	8,290862	8,086713	7,68E+00	7,09E+00	6,323915	5,401134
12	6,859924	7,033103	7,033103	6,859924	6,52E+00	6,02E+00	5,36455	4,581759
11	5,802746	5,949236	5,949236	5,802746	5,51E+00	5,09E+00	4,537823	3,875667
10	4,889039	5,012462	5,012462	4,889039	4,65E+00	4,29E+00	3,823292	3,2654
9	4,096212	4,199621	4,199621	4,096212	3,89E+00	3,59E+00	3,203292	2,735869
8	3,404664	3,490614	3,490614	3,404664	3,23E+00	2,99E+00	2,662492	2,273983
7	2,797295	2,867913	2,867913	2,797295	2,66E+00	2,45E+00	2,187521	1,86832
6	2,259089	2,316119	2,316119	2,259089	2,15E+00	1,98E+00	1,766637	1,50885
5	1,776738	1,821591	1,821591	1,776738	1,69E+00	1,56E+00	1,389432	1,186687
4	1,338316	1,372102	1,372102	1,338316	1,27E+00	1,17E+00	1,046581	0,893864
3	0,932984	0,956538	0,956538	0,932984	8,86E-01	8,18E-01	0,729606	0,623142
2	0,55072	0,564623	0,564623	0,55072	5,23E-01	4,83E-01	0,430671	0,367827
1	0,182073	0,186669	0,186669	0,182073	1,73E-01	1,60E-01	0,142383	0,121607

J\I	17	18	19	20
20	13,28158	9,727566	5,93403	1,99438
19	11,34174	8,306808	5,067337	1,703091
18	9,682322	7,091435	4,325932	1,453911
17	8,2623	6,051395	3,691485	1,240678
16	7,046561	5,160975	3,148309	1,058121
15	6,005047	4,398159	2,682974	0,901726
14	5,112006	3,744087	2,283976	0,767626
13	4,345359	3,182586	1,941448	0,652505
12	3,686149	2,699774	1,646922	0,553517
11	3,118079	2,283714	1,393116	0,468215
10	2,627103	1,924118	1,173754	0,394489
9	2,201081	1,612095	0,983414	0,330517
8	1,829481	1,339931	0,817387	0,274717
7	1,503114	1,100896	0,671571	0,22571
6	1,213911	0,889081	0,542359	0,182283
5	0,954722	0,699248	0,426557	0,143362
4	0,719138	0,526704	0,321301	0,107987
3	0,501335	0,367183	0,22399	0,075281
2	0,295927	0,21674	0,132216	0,044437
1	0,097836	0,071656	0,043712	0,014691

Os valores obtidos com a versão em MATLAB do TEACH-C apresentam discrepâncias em relação aos da solução analítica (em alguns casos erro de 7%). Ao aumentar o número de termos da série verificou-se que o campo de temperaturas se aproximava do obtido pelo TEACH-C.

Em seguida é apresentada a variação da Temperatura em função do número de termos da série no ponto de coordenadas $x=0.475$ e $y=0.975$ (ponto de temperatura máxima para a malha utilizada). Verifica-se que com o aumento do número de termos da série o valor de T se aproxima do valor obtido no TEACH-C ($23,5^{\circ}\text{C}$).

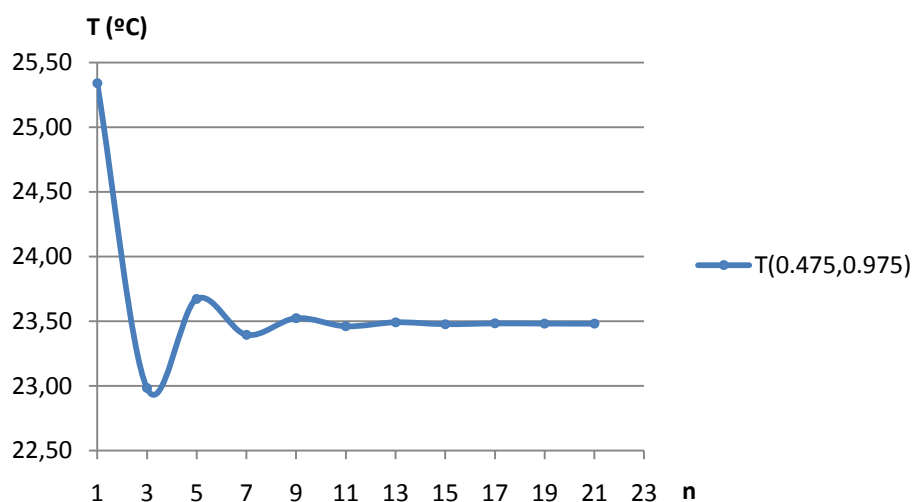


Gráfico 1 – Temperatura no ponto (0.475,0.975) em função do número de termos da série.

O campo obtido para $n=21$ é bastante semelhante ao obtido pelo TEACH-C o que garante a validade do código. Em seguida, apresenta-se este campo e os perfis de temperatura a meio do corpo, ou seja, para $x=0.475$ e $y=0.475$.

J\I	1	2	3	4	5	6	7	8
20	3,336357	8,657296	12,59465	15,64271	18,0705	19,95027	21,41066	22,45314
19	2,295315	6,512342	10,02209	12,87413	15,17076	16,99103	18,39318	19,41486
18	1,762637	5,128284	8,113391	10,65035	12,74935	14,44166	15,75847	16,72496
17	1,413749	4,153943	6,664376	8,866891	10,73441	12,26653	13,47307	14,36581
16	1,158921	3,421609	5,531926	7,422573	9,056324	10,41718	11,50114	12,30973
15	0,961777	2,847192	4,624395	6,238665	7,653042	8,845582	9,804826	10,52567
14	0,80392	2,383763	3,882905	5,257197	6,473323	7,508301	8,34744	8,981967
13	0,674652	2,002556	3,268156	4,435656	5,476141	6,367836	7,095296	7,64815
12	0,567121	1,684546	2,752699	3,742349	4,628818	5,392448	6,018384	6,495961
11	0,476609	1,41637	2,316528	3,15308	3,905145	4,555456	5,090405	5,499798
10	0,399685	1,188165	1,944499	2,648912	3,283845	3,834398	4,288487	4,636795
9	0,33374	0,992361	1,62477	2,214675	2,747401	3,210268	3,592783	3,886693
8	0,276719	0,822949	1,347818	1,837954	2,281186	2,666865	2,986051	3,231613
7	0,226954	0,675028	1,105804	1,508392	1,872814	2,190257	2,453248	2,655766
6	0,183055	0,544506	0,89213	1,217193	1,511648	1,768343	1,981169	2,145169
5	0,143839	0,427883	0,701133	0,956752	1,188419	1,39049	1,558119	1,687354
4	0,108277	0,322109	0,527853	0,720376	0,894921	1,047227	1,173622	1,271101
3	0,075452	0,224464	0,367857	0,502061	0,623763	0,729985	0,81816	0,886179
2	0,044526	0,132463	0,217092	0,296306	0,368151	0,430869	0,482939	0,523111
1	0,014719	0,043788	0,071765	0,097953	0,121707	0,142445	0,159663	0,172948
J\I	9	10	11	12	13	14	15	16
20	23,14585	23,48126	23,48126	23,14585	22,45314	21,41066	19,95027	18,0705
19	20,08324	20,41343	20,41343	20,08324	19,41486	18,39318	16,99103	15,17076
18	17,35949	17,67386	17,67386	17,35949	16,72496	15,75847	14,44166	12,74935
17	14,95495	15,24769	15,24769	14,95495	14,36581	13,47307	12,26653	10,73441
16	12,84623	13,11363	13,11363	12,84623	12,30973	11,50114	10,41718	9,056324
15	11,00638	11,24669	11,24669	11,00638	10,52567	9,804826	8,845582	7,653042
14	9,406991	9,620016	9,620016	9,406991	8,981967	8,34744	7,508301	6,473323
13	8,019827	8,206528	8,206528	8,019827	7,64815	7,095296	6,367836	5,476141
12	6,817974	6,980019	6,980019	6,817974	6,495961	6,018384	5,392448	4,628818
11	5,77647	5,915896	5,915896	5,77647	5,499798	5,090405	4,555456	3,905145
10	4,872597	4,991558	4,991558	4,872597	4,636795	4,288487	3,834398	3,283845
9	4,085932	4,186531	4,186531	4,085932	3,886693	3,592783	3,210268	2,747401
8	3,398241	3,482427	3,482427	3,398241	3,231613	2,986051	2,666865	2,281186
7	2,793287	2,8628	2,8628	2,793287	2,655766	2,453248	2,190257	1,872814
6	2,256594	2,312934	2,312934	2,256594	2,145169	1,981169	1,768343	1,511648
5	1,775193	1,819619	1,819619	1,775193	1,687354	1,558119	1,39049	1,188419
4	1,337374	1,370898	1,370898	1,337374	1,271101	1,173622	1,047227	0,894921
3	0,932431	0,955831	0,955831	0,932431	0,886179	0,81816	0,729985	0,623763
2	0,550432	0,564254	0,564254	0,550432	0,523111	0,482939	0,430869	0,368151
1	0,181983	0,186555	0,186555	0,181983	0,172948	0,159663	0,142445	0,121707

J\I	17	18	19	20
20	15,64271	12,59465	8,657296	3,336357
19	12,87413	10,02209	6,512342	2,295315
18	10,65035	8,113391	5,128284	1,762637
17	8,866891	6,664376	4,153943	1,413749
16	7,422573	5,531926	3,421609	1,158921
15	6,238665	4,624395	2,847192	0,961777
14	5,257197	3,882905	2,383763	0,80392
13	4,435656	3,268156	2,002556	0,674652
12	3,742349	2,752699	1,684546	0,567121
11	3,15308	2,316528	1,41637	0,476609
10	2,648912	1,944499	1,188165	0,399685
9	2,214675	1,62477	0,992361	0,33374
8	1,837954	1,347818	0,822949	0,276719
7	1,508392	1,105804	0,675028	0,226954
6	1,217193	0,89213	0,544506	0,183055
5	0,956752	0,701133	0,427883	0,143839
4	0,720376	0,527853	0,322109	0,108277
3	0,502061	0,367857	0,224464	0,075452
2	0,296306	0,217092	0,132463	0,044526
1	0,097953	0,071765	0,043788	0,014719

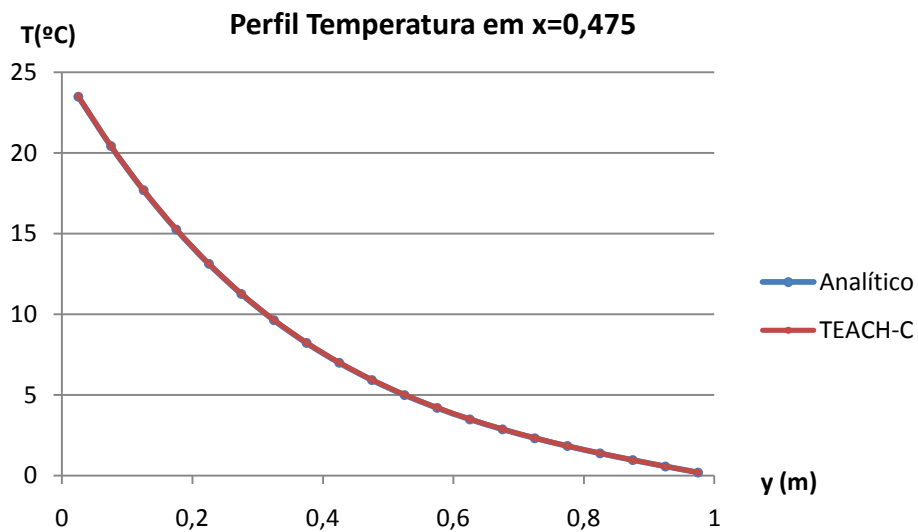


Gráfico 2 – Perfil de Temperaturas em x=0.475m.

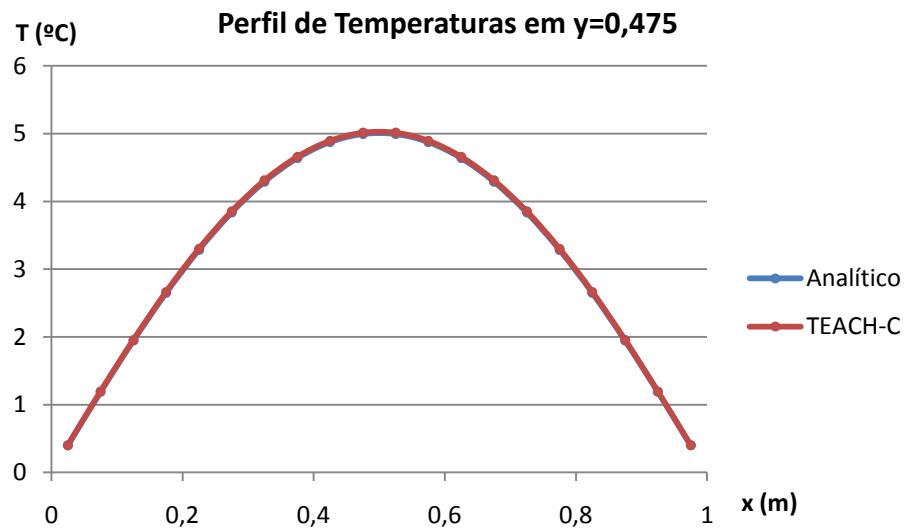


Gráfico 3 – Perfil de temperatura em $y=0.475\text{m}$.

Bibliografia

- [1] – TEACH-C User's Guide and Instructions Manual
- [2] – PATANKAR, S.V.; Numerical Heat Transfer and Fluid Flow; McGraw-Hill
- [3] – DURÃO, D.G.; Folhas de Transmissão de Calor e Massa
- [4] – CARVALHO, M.G.; Computer Simulation of a Furnace; Tese de doutoramento no Imperial College
- [5] – SMITH, G.D.; Numerical Solution of Partial difference Equations
- [6] – GOSMAN, A.D; LAUNDER, B.E.; Reece, G.J.; Computer-Aided Engineering heat transferred and fluid flow; ELLIS HORWOOD LIMITED
- [7] – INCROPERA; DEWITT; BERGMAN; LAVINE; Fundamentals of Heat and Mass Transfer, Sixth Edition; WILEY
- [8] – Folhas de Apoio de Complementos de Transmissão de Calor

Anexo 1-Código

Código Principal:

```
-----%
%-----%
%           Programa TEACH C (versão MATLAB)
%                   CONTROL
%
%   Este programa permite resolver problemas de condução de calor
%           bidimensional com as seguintes variantes:
%   • Coordenadas cartesianas ou coordenadas cilíndricas
%   (axissimétrico)
%   • Regime estacionário ou transiente (não estacionário)
%   • Condutividade térmica (k) uniforme ou variável
%   (com a temperatura ou com o espaço)
%   • Com a utilização dos três tipos de condições de fronteira
%   existentes
%       - Temperatura imposta
%       - Fluxo imposto
%       - Convecção na fronteira do corpo
%                   FS
%                   09/05/2011
%-----%
%-----%

%Apaga variáveis e limpa a consola
clear all
clc

%Utiliza o formato longo (15 dígitos) para maior precisão
format long

%Cria ficheiro para escrever os resultados
fid=fopen('RESULTS.txt','w');

%----- Capitulo 0 - Preliminares -----%

%Define o número de nós da malha segundo x e y
%%%%% Alterável %%%%
IT=22;
JT=22;

%Define o número de pontos da malha segundo x e y
%%%%% Alterável %%%%
NI=22;
NJ=22;

%Constante
GREAT=1.0E30;
```

```

%----- Capitulo 1 - Parâmetros e índices de controlo -----%

%Define o tipo de coordenadas (cartesianas- 0 0;
%cilíndricas vertical 1 0; cilíndricas horizontais 0 1)
%%%% Alterável %%%%
INCYLX=0;
INCYLY=0;

%Variável intermédia
NIM1=NI-1;
NJM1=NJ-1;

%Dimensões totais do domínio de solução, largura e altura [m]
%%%% Alterável %%%%
W=1;
H=1;

%Cálculo as abcissas na direcção xx
DX(1)=W/(NIM1-1);
X(1)=-0.5*DX(1);
for I=2:NI;
    DX(I)=DX(1);
    X(I)=X(I-1)+DX(I-1);

end

%Calcula as abcissas na direcção yy
DY(1)=H/(NJM1-1);
Y(1)=-0.5*DY(1);
for J=2:NJ;
    DY(J)=DY(1);
    Y(J)=Y(J-1)+DY(J-1);

end

%Estabelece limites do Domínio
for I=1:NI;
    JS(I)=2;
    JN(I)=NJ-1;

end

%Estabelece ponto monitor segundo x e y (IMON e JMON)
%%%% Alterável %%%%
IMON=6;
JMON=6;

%Propriedades do material(Tcond=cond,CV=calor esp,DENSIT=dens),
%Meio homogéneo
%Aço %%%% Alterável %%%%
for I=1:NI;
    for J=1:NJ;
        TCON(I,J)=14.68;
        CV(I,J)=485.67;
        DENSIT(I,J)=7800;
        if I==1 && J==1;
            BK=TCON(I,J);
        end
    end
end

```

```

        end
    end
end

%Temperatura inicial
TINIC=0;

%Parâmetros de controlo do programa
%%%% Alterável %%%%
%Número máximo de iterações
MAXIT=10;
%Número máximo de iterações no tempo
MAXSTP=20;
%O output deverá conter os valores de T em intervalos de
NITPRI=110;
%"NITPRI" para "NSTPRI" iterações no tempo
NSTPRI=1;

%Factor de sub relaxação, Máximo residuo e intervalo de tempo (s)
%%%% Alterável %%%%
URFT=1;
SORMAX=0.001;
DT=50;

%Selecciona o Regime---Estacionário->INTIME=0, Transiente->INTIME=1
%%%% Alterável %%%%
INTIME=1;

if INTIME==0;
    MAXSTP=1;

end

%Indica se as propriedades são constantes --- constantes->INPRO=0,
%variáveis->INPRO=1
%%%% Alterável %%%%
INPRO=0;

%----- Capitulo 2 - Operações iniciais -----%

%Calcula dimensões da malha e anula vectores/matrizes
%Chama função INIT
[RX,DXEP,DXPW,SEW,XU,RU,RY,DYPS,DYNP,SNS,YV,RV,AN,AS,AE,AW,SU,SP,GAMH,
TOLD,T,X,Y]=INIT(INCYLX,INCYLY,NI,NJ,NIM1,NJM1,X,Y,TINIC);
TIME=0.0;

%Impõe valores de fronteira e inicializa variável dependente

%Valores de fronteira
%%%% Alterável %%%%
TTOP=100;
TBOT=0;
TLEFT=0;
TRIGHT=0;

for I=2:NIM1
    T(I,1)=TBOT;
    T(I,NJ)=TTOP;
end

```

```

for J=2:NJM1
    T(1,J)=TLEFT;
    T(NI,J)=TRIGHT;
end

%Inicializa variável dependente
%Inicializa campo de propriedades do material
%Chama função PROPS
[GAMH]=PROPS(NI,NJ,TCON,GAMH);

%Cálculo do factor de normalização do resíduo
AK=BK;
SNORM=AK*(TTOP-TBOT)*W/H;
SNORM=abs(SNORM);

%Escreve as especificações do problema
fprintf(fid,'CONDUCTION IN RECTANGULAR BAR WITH PRESCRIBED SURFACE
TEMPERATURE \r\n \r\n');
fprintf(fid,'HEIGHT, H [M]-----=
%10.3f\r\n',H);
fprintf(fid,'WEIGHT, W [M]-----=
%10.3f\r\n',W);
fprintf(fid,'SPECIFIC HEAT, CV [J/KG.K]-----=
%10.3f\r\n',CV(1,1));
fprintf(fid,'THERMAL CONDUCTIVITY, TCON [W/M.K]-----=
%10.3f\r\n',TCON(1,1));
fprintf(fid,'DENSIT, DENSIT [KG/M3] -----=
%10.2f\r\n',DENSIT(1,1));
fprintf(fid,'INITIAL TIME STEP, DT [S] -----=
%10.1f\r\n',DT);
fprintf(fid,'SOURCE NORMALIZATION FACTOR, SNORM -----=
%8.3E\r\n',SNORM);
fprintf(fid,'NUMBER OF NODES IN X DIRECTION, NI -----=
%10d\r\n',NI);
fprintf(fid,'NUMBER OF NODES IN Y DIRECTION, NJ -----=
%10d\r\n',NJ);
fprintf(fid,'\r\n');

%Chama a função PRINT para imprimir o campo de temperaturas inicial
PRINT(1,1,NI,NJ,X,Y,T,fid)

%----- Capitulo 3 - Iterações no tempo e no espaço -----%

%Indica os pontos de controlo
%Imprime o rótulo das informações das iterações no ponto monitor
fprintf(fid,'NITER          SOURCE          T(%d,%d)          TIME(s)          DT(s)
NSTEP',IMON,JMON);
fprintf(fid,'\r\n');

%Iterações no tempo
for NSTEP=1:MAXSTP;
    TIME=TIME+DT;
    for I=1:NI;
        for J=1:NJ;
            TOLD(I,J)=T(I,J);
        end
    end
end

```



```

%Iterações no espaço
for NITER=1:MAXIT;

    %Chama a função CALCT para o cálculo das temperaturas

    [AN, AS, AE, AW, SU, SP, GAMH, CV, DENSIT, TOLD, T, RESORT]=CALCT (NI, NJ, NIM1, NJM1
    , RX, DXEP, SEW, RU, RY, DYNP, SNS, RV, AN, AS, AE, AW, SU, SP, GAMH, CV, DENSIT, TOLD, T
    , URFT, JS, JN, INTIME, DT, Y, X, XU, YV, GREAT) ;

    %Chama a função PROPS no caso de as propriedades variarem
    if INPRO==1
        [GAMH]=PROPS (NI, NJ, TCON, GAMH) ;

    end

    %Atualização de condições de fronteira e fontes se necessário

    %Cálculo do resíduo normalizado
    SOURCE=(RESORT/SNORM) ;

    %Imprime a informação das iterações no ponto monitor
    fprintf(fid, '%5d', NITER);
    fprintf(fid, '%14.1E', SOURCE);
    fprintf(fid, '%14.3E', T (IMON, JMON));
    fprintf(fid, '%11d', TIME);
    fprintf(fid, '%10d', DT);
    fprintf(fid, '%9d', NSTEP);
    fprintf(fid, '\r\n');

    %Imprime Temperaturas em intervalos especificados por NITPRI
    if mod(NITER, NITPRI)==0
        PRINT (1, 1, NI, NJ, X, Y, T, fid)

        if NSTEP~=MAXSTP || SOURCE>SORMAX;
            %Imprime o rótulo das informações das iterações no
ponto monitor
            fprintf(fid, 'NITER          SOURCE          T(%d,%d)
TIME(s)      DT(s)      NSTEP', IMON, JMON);
            fprintf(fid, '\r\n');

            end
        end

        %Testa resíduo do processo iterativo
        if SOURCE<SORMAX;
            break
        end

        %Termina cálculos se a solução não converge (MAXIT e
RESÍDUO<10)
        if NITER>=MAXIT && SOURCE>=10;
            error('myApp:argChk', 'Não Convergiu segundo o critério
especificado \n')

        end
    end
end

```

```

%Termina ciclo no espaço
end

fprintf(fid, '\r\n \r\n \r\n');

%Imprime a solução convergida no intervalo especificado por NSTPRI
if mod(NSTEP, NSTPRI) == 0 && mod(NITER, NITPRI) ~= 0
    PRINT(1, 1, NI, NJ, X, Y, T, fid)
end

if NSTEP ~= MAXSTP
    %Imprime o rótulo das informações das iterações no ponto
monitor
    fprintf(fid, 'NITER          SOURCE          T(%d,%d)          TIME(s)
DT(s)      NSTEP', IMON, JMON);
    fprintf(fid, '\r\n');

end

%Termina ciclo no tempo
end

fclose(fid);

%Desenha gráfico 2D das Temperaturas

%Troca os eixos
for jj=1:NI
    for ii=1:NJ
        THI(jj, ii) = T(ii, NJ+1-jj);
    end
end

%Gráfico da Temperatura final
[X, Y] = meshgrid(0:H/(NI-1):H, W:- (W/(NJ-1)):0);
%Z1=real(Z);
figure(1)
contourf(X, Y, THI);
xlabel('\bfx')
ylabel('\bfy')
zlabel('\bfT')

```

Função CALCT:

```
%-----%
%-----%
%               script: CALCT (versão MATLAB)
%   A finalidade deste bloco é a de calcular os coeficientes da
%       equação diferencial de calor discretizada
%               FS
%               09/05/2011
%-----%
%-----%

function[AN,AS,AE,AW,SU,SP,GAMH,CV,DENSIT,TOLD,T,RESORT]=CALCT(NI,NJ,N
IM1,NJM1,RX,DSEP,SEW,RU,RY,DYNP,SNS,RV,AN,AS,AE,AW,SU,SP,GAMH,CV,DENSI
T,TOLD,T,URFT,JS,JN,INTIME,DT,Y,X,XU,YV,GREAT)

%----- Capitulo 0 - Preliminares -----%

%----- Capitulo 1 - Calcula coeficientes -----%

RESORT=0;

%Ciclo pelas colunas I=cte
for I=2:NIM1;

    %Encontra limites JJ inferior e superior para cada coluna I=cte
    LJS=JS(I);
    LJN=JN(I);

    %Calcula coeficientes para toda a coluna I=cte
    for J=LJS:LJN;

        %Determina areas e volume
        AREAN=RV(J+1)*SEW(I)*RX(I);
        AREAE=RY(J)*SNS(J)*RU(I+1);
        VOL=RY(J)*SNS(J)*SEW(I)*RX(I);

        %Calcula coeficientes de difusão
        GAMN=0.5*(GAMH(I,J)+GAMH(I,J+1));
        GAME=0.5*(GAMH(I,J)+GAMH(I+1,J));
        DN=GAMN*AREAN/DYNP(J);
        DE=GAME*AREAE/DSEP(I);

        %Termos de fonte quando existentes
        %%%% Alterável %%%%
        SU(J)=0;
        SP(J)=0;

        %Coeficientes Transientes
        if INTIME==1;
            DP=VOL*CV(I,J)*DENSIT(I,J)/DT;
            SU(J)=SU(J)+DP*TOLD(I,J);
            SP(J)=SP(J)-DP;
        end

        %Calcula coeficientes
        AN(J)=DN;
```

```

        AS(J)=AN(J-1);
        AW(J)=AE(J);
        AE(J)=DE;

end

IL=I;

%-- Capitulo 2 - Modificações do problema: condições de fronteira ---%

%Chama a Função PROMOD

[AN,AS,AE,AW,SU,SP,T]=PROMOD(NI,NJ,NIM1,NJM1,IL,RV,YV,Y,SNS,SEW,X,XU,A
N,AS,AE,AW,SU,SP,GAMH,T,RY);

%----- Capitulo 3 - Coeficientes finais + Resíduo -----%

for J=LJS:LJN;
    AP(J)=AN(J)+AS(J)+AE(J)+AW(J)-SP(J);
    RESOR=AN(J)*T(I,J+1)+AS(J)*T(I,J-1)+AE(J)*T(I+1,J)+AW(J)*T(I-
1,J)-AP(J)*T(I,J)+SU(J);
    VOL=RY(J)*SEW(I)*SNS(J)*RX(I);

    %Modifica RESOR se condições de fronteira são aplicadas com
recurso
    %a SP=-GREAT
    if -SP(J)>(0.5*GREAT);
        RESOR=RESOR/GREAT;
    end

    %Soma resíduo para esta coluna I=cte
    RESORT=RESORT+abs(RESOR);

    %Sub-relaxação
    AP(J)=AP(J)/URFT;
    SU(J)=SU(J)+(1.0-URFT)*AP(J)*T(I,J);
end

%----- Capitulo 4 - Solução das equações algébricas -----%

%Effectua iteração linha-a-linha
[T]=SOLVE(NI,NJ,IL,LJS,LJN+1,T,AN,AS,AE,AW,AP,SU,SP);

end

%Termina CALCT
end

```

Função INIT:

```
%-----%
%-----%
%               script: INIT (versão MATLAB)
% Inicializa as matrizes a zero e efectua o cálculo de todos os
%           valores associados à malha utilizada
%               FS
%               09/05/2011
%-----%
%-----%

function[RX,DXEP,DXPW,SEW,XU,RU,RY,DYPS,DYNP,SNS,YV,RV,AN,AS,AE,AW,SU,
SP,GAMH,TOLD,T,X,Y]=INIT(INCYLX,INCYLY,NI,NJ,NIM1,NJM1,X,Y,TINIC)

%----- Capitulo 0 - Preliminares -----%

%----- Capitulo 1 - Erro de dados:-----%
%----- Programa encerra se INCYLX=1 e INCYLY=1 -----%

if INCYLX==1 && INCYLY==1;
    error('myApp:argChk','ERRO-Programa vai encerrar
\nINCYLX=INCYLY=1')
else

%----- Capitulo 2 - Cálculo dimensões da malha -----%

    %Impõe RX=X se axissimétrico na direcção xx
    for I=1:NI;
        RX(I)=1.0;
        if INCYLX==1;
            RX(I)=X(I);
        end
    end

    %Impõe RY=Y se assimétrico na direcção yy
    for J=1:NJ;
        RY(J)=1.0;
        if INCYLY==1;
            RY(J)=Y(J);
        end
    end

    %Cálculo da distância entre nos (ver figura 1)
    DXPW(1)=0;
    DXEP(NI)=0;
    for I=1:NIM1;
        DXEP(I)=X(I+1)-X(I);
        DXPW(I+1)=DXEP(I);
    end

    DYPS(1)=0;
    DYNP(NJ)=0;
    for J=1:NJM1;
        DYNP(J)=Y(J+1)-Y(J);
        DYPS(J+1)=DYNP(J);
    end
end
```

```

end

%Cálculo das dimensões do volume de controle
SEW(1)=0;
SEW(NI)=0;
for I=2:NIM1;
    SEW(I)=0.5*(DXEP(I)+DXPW(I));
end

SNS(1)=0;
SNS(NJ)=0;
for J=2:NJM1;
    SNS(J)=0.5*(DYNP(J)+DYPS(J));
end

%Localização das fronteiras dos volumes de controle
XU(1)=0;
RU(1)=0;
for I=2:NI;
    RU(I)=0.5*(RX(I)+RX(I-1));
    XU(I)=0.5*(X(I)+X(I-1));
end

YV(1)=0;
RV(1)=0;
for J=2:NJ;
    RV(J)=0.5*(RY(J)+RY(J-1));
    YV(J)=0.5*(Y(J)+Y(J-1));
end

%Modificação dos valores de fronteira de x e de y
X(1)=XU(2);

if X(1)<((XU(NI)-XU(2))*1.0E-03)
    X(1)=0;
end

X(NI)=XU(NI);
Y(1)=YV(2);

if Y(1)<((YV(NJ)-YV(2))*1.0E-03)
    Y(1)=0.0;
end

Y(NJ)=YV(NJ);

```

%----- Capitulo 3 - Inicialização das matrizes a zero -----%

```

J=NJ;
I=NI;

AE=zeros(J);
AW=zeros(J);
AN=zeros(J);
AS=zeros(J);
SU=zeros(J);
SP=zeros(J);
GAMH=zeros(I,J);

```

```
TOLD=zeros(I,J);

for ii=1:I;
    for jj=1:J;
        T(ii,jj)=TINIC;
    end
end

end

%Termina INIT
end
```

Função PRINT:

```
%-----%
%-----%
%               script: PRINT (versão MATLAB)
%               Produz o Output
%               Mostra ao utilizador a matriz T/PHI
%               FS
%               09/05/2011
%-----%
%-----%

function []=PRINT (ISTART,JSTART,IEND,JEND,X,Y,PHI,fid)

%----- Capitulo 0 - Preliminares -----%

%----- Capitulo 1 - Inicialização e Cabeçalhos -----%

ISKIP=1;
JSKIP=1;

%%%%% Alterável %%%%%
LINLIM=8;

LINSTA=ISTART;

%Escreve o cabeçalho da matriz
fprintf(fid,'\r\n');
fprintf(fid,'-----TEMPERATURE (°C)---
-----\r\n');

%Define um valor de LINEND inferior a IEND para inicializar o ciclo
while
LINEND=0;

while LINEND<IEND

    %Escreve cabeçalho da matriz
    LINEND=LINSTA+(LINLIM-1)*ISKIP;
    LINEND=min(IEND,LINEND);

    %Escreve a segunda linha do output
    fprintf(fid,'I= ');
    for KI=LINSTA:ISKIP:LINEND
        fprintf(fid,'%10d',KI);

    end
    fprintf(fid,'          Y\r\n');

%----- Capitulo 2 - Escreve a matriz PHI -----%

    fprintf(fid,' J\r\n');
    for JJ=JSTART:JSKIP:JEND;
        J=JSTART+JEND-JJ;
        IS=0;

        for I=LINSTA:ISKIP:LINEND;
```



```

        A=PHI (I,J) ;

        IS=IS+1;

        if abs (A)<(1e-20) ;
            A=0.0;
        end

        STORE (IS) =A;

    end

    %Escreve o valor de J, T e Y
    fprintf(fid,'%3d',J);
    for KT=1:IS
        fprintf(fid,'%10.2E', STORE (KT));
    end
    fprintf(fid,'%10.4f\r\n', Y(J));

end

%Escreve uma linha com os X's
fprintf(fid,'\r\n');
fprintf(fid,'X= ');
for KX=LINSTA:ISKIP:LINEND
    fprintf(fid,'%10.4f',X(KX));
end

fprintf(fid,'\r\n \r\n');

LINSTA=LINEND+ISKIP;

end

fprintf(fid,'\r\n \r\n');

%Termina PRINT
end

```

Função PROMOD:

```
%-----%
%-----%
%               script: PROMOD (versão MATLAB)
%               Define e introduz as Condições de Fronteira
%               -Temperatura Imposta
%               -Fluxo Imposto
%               -Convecção
%               -Simetria
%               FS
%               09/05/2011
%-----%
%-----%
```

```
function [AN,AS,AE,AW,SU,SP,T]=PROMOD (NI,NJ,NIM1,NJM1,IL,RV,YV,Y,SNS,SE
W,X,XU,AN,AS,AE,AW,SU,SP,GAMH,T,RY)
```

```
%----- Capitulo 0 - Preliminares -----%
```

```
%----- Capitulo 1 - Condição de Fronteira -----%
```

```
%O Utilizador define o tipo de condição de fronteira (1-Temperatura,
%2-Fluxo, 3-Convecção e 4-simetria para a fronteira Norte, Sul, Este e
%Oeste (CFN,CFS,CFW,CFE)
```

```
%%%% Alterável %%%%
```

```
CFN=1;
```

```
CFS=1;
```

```
CFW=1;
```

```
CFE=1;
```

```
%Identifica o tipo de condição de fronteira
```

```
I=IL;
```

```
%%FRONTEIRA NORTE-----
```

```
%Temperatura imposta
```

```
if CFN==1;
```

```
RDYN=RV (NJ) / ( YV (NJ) -Y (NJM1) ) ;
```

```
AN (NJM1) =0;
```

```
DN=GAMH (IL,NJM1) *SEW (IL) *RDYN;
```

```
SU (NJM1) =SU (NJM1) +DN*T (IL,NJ) ;
```

```
SP (NJM1) =SP (NJM1) -DN;
```

```
%Fluxo imposto
```

```
elseif CFN==2;
```

```
QN=20; %%%% alterável %%%%
```

```
AN (NJM1) =0;
```

```
DN=QN*SEW (IL) *RV (NJ) ;
```

```
SU (NJM1) =SU (NJM1) +DN;
```

```
SP (NJM1) =SP (NJM1) ;
```

```
%Convecção
```

```

elseif CFN==3;

    HCONV=12.5;          %%%% alterável %%%%
    TF=80;              %%%% alterável %%%%
    RDYN=YV(NJ)-Y(NJM1);
    AN(NJM1)=0;
    DN1=RDYN/GAMH(IL,NJM1)+1/HCONV;
    DN=SEW(IL)*RV(NJ)/DN1;
    SU(NJM1)=SU(NJM1)+DN*TF;
    SP(NJM1)=SP(NJM1)-DN;

%simetria
elseif CFN==4;

    AN(NJM1)=0;
    T(IL,NJ)=T(IL,NJM1);

%Caso o utilizador tenha introduzido incorrectamente a condição de
%fronteira o programa termina
else
    error('myApp:argChk','Condição de fronteira mal introduzida, o
programa vai encerrar')

end
%%-----

%%FRONTEIRA SUL-----
%Temperatura imposta
if CFS==1;

    RDYS=RV(2)/(Y(2)-YV(2));
    AS(2)=0;
    DS=GAMH(IL,2)*SEW(IL)*RDYS;
    SU(2)=SU(2)+DS*T(IL,1);
    SP(2)=SP(2)-DS;

%Fluxo imposto
elseif CFS==2;

    QS=20;              %%%% alterável %%%%
    AS(2)=0;
    DS=QS*SEW(IL)*RV(2);
    SU(2)=SU(2)+DS;
    SP(2)=SP(2);

elseif CFS==3;

    HCONV=12.5;          %%%% alterável %%%%
    TF=80;              %%%% alterável %%%%
    RDYS=Y(2)-YV(2);
    AS(2)=0;
    DS1=RDYS/GAMH(IL,2)+1/HCONV;
    DS=SEW(IL)*RV(2)/DS1;
    SU(2)=SU(2)+DS*TF;

```

```

SP(2)=SP(2)-DS;

%simetria
elseif CFS==4;

    AS(2)=0;
    T(IL,2)=T(IL,2);

%Caso o utilizador tenha introduzido incorrectamente a condição de
%fronteira o programa termina
else

    error('myApp:argChk','Condição de fronteira mal introduzida, o
programa vai encerrar')

end
%%%-----

%%%FRONTEIRA OESTE-----
%Temperatura imposta
if CFW==1;

    if IL==2;
        DXW=X(2)-XU(2);
        for J=2:NJM1;
            AW(J)=0;
            DW=GAMH(IL,J)*SNS(J)*RY(J)/DXW;
            SU(J)=SU(J)+DW*T(1,J);
            SP(J)=SP(J)-DW;
        end
    end

%Fluxo imposto
elseif CFW==2;

    if IL==2;
        QW=20;          %%%% alterável %%%%
        for J=2:NJM1;
            AW(J)=0;
            DW=QW*SNS(J)*RY(J);
            SU(J)=SU(J)+DW;
            SP(J)=SP(J);
        end
    end

%Convecção
elseif CFW==3;

    if IL==2;
        HCONV=12.5;      %%%% alterável %%%%
        TF=80;           %%%% alterável %%%%
        DXW=X(2)-XU(2);

```

```

        for J=2:NJM1
            AW(J)=0;
            DW1=DXW/GAMH(IL,J)+1/HCONV;
            DW=SNS(J)*RY(2)/DW1;
            SU(J)=SU(J)+DW*TF;
            SP(J)=SP(J)-DW;
        end
    end

elseif CFW==4

    if IL==2;

        for J=2:NJM1;
            AW(J)=0;
            T(1,J)=T(2,J);
        end

    end

    %Caso o utilizador tenha introduzido incorrectamente a condição de
    %fronteira o programa termina
    else

        error('myApp:argChk','Condição de fronteira mal introduzida, o
        programa vai encerrar')

    end

    end
    %%%-----

    %%%FRONTEIRA ESTE-----
    %Temperatura imposta
    if CFE==1;

        if IL==NIM1;
            DXE=XU(NI)-X(NIM1);
            for J=2:NJM1;
                AE(J)=0;
                DE=GAMH(IL,J)*SNS(J)*RY(J)/DXE;
                SU(J)=SU(J)+DE*T(NI,J);
                SP(J)=SP(J)-DE;
            end
        end

    end

    %Fluxo imposto
    elseif CFE==2;

        if IL==NIM1;
            QE=0;          %%%% alterável %%%%
            for J=2:NJM1;
                AE(J)=0;
                DE=QE*SNS(J)*RY(J);
                SU(J)=SU(J)+DE;
            end
        end
    end
end

```

```

        SP(J)=SP(J);

    end
end

%Convecção
elseif CFE==3;

    if IL==NIM1;
        HCONV=12.5;      %%%% alterável %%%%
        TF=80;          %%%% alterável %%%%
        DXE=XU(NI)-X(NIM1);
        for J=2:NJM1
            AE(J)=0;
            DE1=DXE/GAMH(IL,J)+1/HCONV;
            DE=SNS(J)*RY(J)/DE1;
            SU(J)=SU(J)+DE*TF;
            SP(J)=SP(J)-DE;
        end
    end

%simetria
elseif CFE==4

    if IL==NIM1;

        for J=2:NJM1;
            AE(J)=0;
            T(NI,J)=T(NIM1,J);
        end

    end

    %Caso o utilizador tenha introduzido incorrectamente a condição de
    %fronteira o programa termina
    else

        error('myApp:argChk','Condição de fronteira mal introduzida, o
        programa vai encerrar')

    end

end
%%%-----

%Termina PROMOD
end

```

Função PROPS:

```

%-----%
%-----%
%               script: PROPS (versão MATLAB)
% Especificação da variação das propriedades do material, nomeadamente
%               a conductividade térmica, K
%               FS
%               09/05/2011
%-----%
%-----%

function[GAMH]=PROPS(NI, NJ, TCON, GAMH)

%-----Capitulo 0 - Preliminares-----%

%----- Capitulo 1 - Actualização das propriedades -----%

for I=1:NI;
    for J=1:NJ;
        GAMH(I,J)=TCON(I,J);
    end
end

%Termina PROPS
end

```

Função SOLVE:

```
%-----%
%-----%
%               script: SOLVE (versão MATLAB)
%       Esta função tem por finalidade a resolução de um sistema de
%       equações em que a matriz dos coeficientes é
%               uma matriz tri-diagonal
%               FS
%               09/05/2011
%-----%
%-----%

function [PHI]=SOLVE (NI,NJ,IL,JSTART,JEND,PHI,AN,AS,AE,AW,AP,SU,SP)

%////////// Algoritmo de THOMA: TDMA \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\%

%----- Capitulo 0 - Preliminares -----%

%----- Capitulo 1 - Processo de iteração linha a linha -----%

JENDM1=JEND-1;
JSTM1=JSTART-1;

%Inicialização dos coeficientes
for J=JSTART:JEND;
    A(J)=0;
    B(J)=0;
    C(J)=0;
    D(J)=0;

end

A(JSTM1)=0;
I=IL;
C(JSTM1)=PHI(I,JSTM1);

for J=JSTART:JENDM1;

    %Assemblagem dos coeficientes TDMA
    A(J)=AN(J);
    B(J)=AS(J);
    C(J)=AE(J)*PHI(I+1,J)+AW(J)*PHI(I-1,J)+SU(J);
    D(J)=AP(J);

    %Calcula os coeficientes da fórmula de recorrência
    DIV=D(J)-B(J)*A(J-1);
    TERM=1.0/DIV;
    A(J)=A(J)*TERM;
    C(J)=(C(J)+B(J)*C(J-1))*TERM;

end

%Obtém novos PHI's por substituição para trás
for JJ=JSTART:JENDM1;
    J=JEND+JSTM1-JJ;
```



```
    PHI (I, J) =A (J) *PHI (I, J+1) +C (J) ;  
  
end  
  
%Termina SOLVE  
end
```


Anexo 2-Listagem de variáveis

Variável	Descrição	Tipo
GREAT	Número grande para evitar indeterminações	Real
URFT	Coefficiente de sub-relaxação ($0 \leq URFT \leq 1$)	
SORMAX	Resíduo máximo para critério de paragem. Não deve ser modificado em caso de se trabalhar com precisão simples	
SNORM	Taxa de transferência de calor característica utilizada para normalizar a fonte energia residual total	
DT	Valor do intervalo de tempo entre duas iterações [s]	
TIME	Intervalo de tempo do problema/instante	
SOURCE	Resíduo	
TTOP	Temperatura no topo da barra rectangular	
TBOT	Temperatura na base da barra rectangular	
TLEFT	Temperatura na face esquerda da barra rectangular	
TRIGHT	Temperatura na face direita da barra rectangular	
TINIC	Temperatura inicial do corpo	
W	Largura do Domínio [m]	
H	Altura do Domínio [m]	
BK	Valor mínimo da condutividade térmica do meio	
AK	Igual a BK	
RESORT	Somatório dos valores absolutos das fontes de energia residual	Inteiro Parâmetro
IT	Número de nós utilizados na malha segundo xx	
JT	Número de nós utilizados na malha segundo yy	
NI	Número de pontos utilizados na malha segundo xx	
NJ	Número de pontos utilizados na malha segundo yy	Real Dimensão NI,NJ
TCOND	Condutividade térmica do meio [W/m.K]	
T(I,J)	Temperatura no nó (I,J)	
TOLD(I,J)	Temperatura na iteração anterior no nó (I,J)	
CV	Calor Específico do meio [J/Kg.K]	
DENSIT	Densidade do meio [Kg/m ³]	
GAMH(I,J)	Condutividade térmica local no nó (I,J)	Real Dimensão NI
X(I)	Coordenada horizontal do nó (I,J)	
DX	Espaço entre dois pontos segundo xx	
RX(I)	Distância radial do nó (I,J) em relação ao eixo de simetria quando INCYLX=1	
DXEP(I)	Distância inter-nó, $\delta x_{EP} = X(I+1) - X(I)$	
DXPW(I)	Distância inter-nó, $\delta x_{PW} = X(I) - X(I-1)$	
SEW(I)	Dimensão axial da célula (I,J)	
XU(I)	Coordenada horizontal da fronteira Oeste da célula (I,J)	
RU(I)	Distância radial entre o eixo de simetria e a fronteira Oeste da célula (I,J) quando INCYLX=1	

Y(J)	Coordenada vertical do nó (I,J)	Real Dimensão NJ
DY	Espaço entre dois pontos segundo yy	
RY(J)	Distância radial do nó (I,J) em relação ao eixo de simetria quando INCYLY=1	
DYPS(J)	Distancia inter-nó, $\delta x_{PS}=Y(J)-Y(J-1)$	
DYNP(J)	Distancia inter-nó, $\delta x_{NP}=Y(J+1)-Y(J)$	
SNS(J)	Dimensão radial da célula (I,J)	
YV(J)	Coordenada vertical da fronteira Sul da célula (I,J)	
RV	Distância radial entre o eixo de simetria e a fronteira Sul da célula (I,J) quando INCYLY=1	
AN(J)	Coeficiente AN da equação das diferenças finitas	
AS(J)	Coeficiente AS da equação das diferenças finitas	
AE(J)	Coeficiente AE da equação das diferenças finitas	
AW(J)	Coeficiente AW da equação das diferenças finitas	
SU	Coeficiente SU da equação das diferenças finitas	
SP	Coeficiente SP da equação das diferenças finitas	
JS	Matriz que contém o número das células do limite inferior do domínio de calculo para cada linha vertical da malha	Inteiro Dimensão NI
JN	Matriz que contém o número das células do limite superior do domínio de calculo para cada linha vertical da malha	
NIM1	Variável intermédia, =NI-1	Inteiro
NJM1	Variável intermédia, =NJ-1	
I	Índice segundo x	
J	Índice segundo y	
IMON	Índice I de um ponto do domínio que no processo iterativo permite testar a convergência do processo	
JMON	Índice J de um ponto do domínio que no processo iterativo permite testar a convergência do processo	
MAXIT	Número máximo de iterações para cada iteração no tempo	
MAXSTP	Número máximo de iterações no tempo	
NSTEP	Contador do número de iterações no tempo	
NITER	Contador do número de iterações no espaço	
NITPRI	Intervalo de interacção no final do qual o programa efectua o "output" para "NSTPRI" iterações no tempo	Inteiro
NSTPRI	Intervalo, em número de iterações no tempo, em que a solução é apresentada ("output")	
INPRO	Variável lógica que indica se a condutibilidade térmica do material varia no espaço ou no tempo (INPRO=0 não varia. INPRO=1 varia)	"Lógico"
INTIME	Variável lógica que indica se o problema é estacionário (INTIME=0) ou transiente (INTIME=1)	
INCYLX	Variável lógica que selecciona sistema de coordenadas cilíndricas em que a direcção radial coincide com x (INCYLX=1)	
INCYLY	Variável lógica que selecciona sistema de coordenadas cilíndricas em que a direcção radial coincide com y (INCYLY=1)	

