Insper

PROJETO 2

IMPLEMENTAÇÃO E ANÁLISE DE TEMPERATURA EM UMA PLACA

Transferência de Calor e Mecânica dos Sólidos

- 28 de março de 2017 -

Gabriela Almeida

Pedro Cunial

Rachel Bottino

Engenharia da Computação - 2017

ÍNDICE

OBJETIVOS

Com base nos conceitos desenvolvidos na disciplina de Transferências de Calor e Mecânica dos Sólidos, os alunos deveriam desenvolver, em grupos, um programa para estudo da temperatura de uma placa aquecida utilizando diferenças finitas.

- O grupo deveria obter resultados para:
    - Distribuição de temperatura com condições de contorno fixa nas bordas.
    - Implementar um métodos explícito para solução da equação do calor.

INTRODUÇÃO

O segundo projeto de Transferências de Calor e Mecânica dos Sólidos consistiu na confecção de um programa de simulação de um modelo térmico em uma placa (superfície bidimensional). Para isso o grupo optou por utilizar a linguagem *python* em sua versão 3.5 (mais recente) com o auxílio da biblioteca numpy para manipulação numérica otimizada assim como o matplotlib para a plotar os gráficos. Para rodar o programa o usuário deve ter instalado em seu computador tal como configurado o mesmo com os pré requisitos descritos anteriormente, os quais podem ser todos obtidos ao instalar o pacote anaconda para python 3.

Com isso o código pode ser obtido em: *https://github.com/pedrocunial/ColdBoard* (você pode clonar o repositorio ou baixar o .zip correspondente a versão da data de acesso. Com isso, no diretório ColdBoard2 temos a versão mais atual do projeto que pode ser executado pelo comando *python 3 run.py*. Os valores da simulação podem ser alterados nas variáveis definidas no início do programa em letras maiúscula.

```
LEFT_TEMP = 75.0
RIGHT_TEMP = 50.0
BOT_TEMP = 0.0
TOP_TEMP = 100.0

FLUX_TOP = None
FLUX_LEFT = None
FLUX_RIGHT = None
FLUX_BOT = 0

T_TOTAL = 10
T_STEP = 0.001

LEN_X = 0.5
LEN_Y = 0.5
LEN_STEP = 0.1

K =1
C = 1
D = 1
ALPHA = K/(C*D)
ERROR = 0.0
```

Sendo que as quatro primeiras variáveis correspondem às temperaturas iniciais das bordas esquerda, direita, inferior e superior, respectivamente. As próximas 4 variáveis são os fluxos das bordas superior, esquerda, direita e inferior, dado que se não houver fluxo essas são definidas como *None*. As duas variáveis seguintes são o tempo total de simulação tamanho e de seus passos. Logo em seguida são definidas as dimensões da placa em metros, assim como o passo de discretização da malha. Por fim são definidas as constantes da placa sendo elas a condutividade térmica, o calor específico e sua densidade. A última variável a ser definida é o erro aceitável de forma que se o erro associado ao valor em certo tempo for menor ou igual a esse o loop de cálculo será quebrado, considerando aquele valor como o ideal.

METODOLOGIA

A transferência de calor em uma placa retangular, cujos lados estão submetidos a diferentes temperaturas, como ilustra a Fig.1, pode ser representada pela expressão geral para a difusão de calor, em coordenadas cartesianas:

$$\rho C_p \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{t}} = \frac{\partial}{\partial x}\left(\kappa \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\kappa \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(\kappa \frac{\partial T}{\partial z}\right) + \dot{q}$$

Aplicando a fórmula acima para as condições do nosso problema, chegamos à equação abaixo:

$$\rho C_p \frac{\partial T}{\partial t} = k\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

$$\frac{\partial T}{\partial t} = \frac{k}{\rho C_p}\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

$$\frac{\partial T}{\partial t} = \alpha\left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2}\right)$$

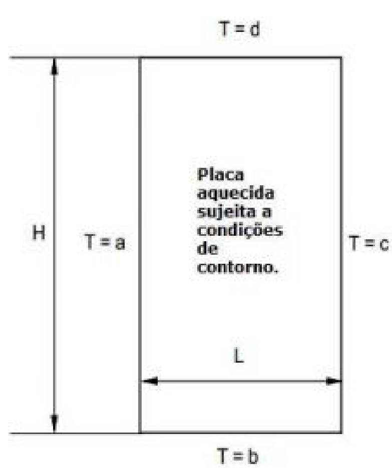


Fig.1

Considerando que todos os pontos da placa estejam a uma temperatura inicial T_0 e sendo esta temperatura diferente da das bordas e que o termo de geração de calor seja nulo, o problema que se coloca é determinar a temperatura em qualquer ponto interno da placa em um dado instante de tempo.

Para resolver, utilizamos o método das diferenças finitas, substituindo as derivadas parciais na equação diferencial parcial, que descreve a condução do calor, por equação de diferença algébrica, que pode ser facilmente executado em computadores.

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i,j-1} - 2T_{i,j} + T_{i,j+1}}{\Delta x^2}$$

$$\frac{\partial^2 T}{\partial y^2} = \frac{T_{i-1,j} - 2T_{i,j} + T_{i+1,j}}{\Delta y^2}$$

Para representar a primeira derivada no tempo, utilizamos a diferença progressiva, que é expressa da seguinte forma:

$$\frac{\partial T}{\partial t} = \frac{T_{i,j}{}^{l+1} - T_{i,j}{}^{l}}{\Delta t}$$

Substituindo essas equações de diferenças algébricas na equação de difusão de calor e considerando que $\Delta x = \Delta y$, obtivemos tal equação:

$$T_{i,j}{}^{l+1} = F_0\left(T^l{}_{i,j+1} + T^l{}_{i,j-1} + T^l{}_{i+1,j} + T^l{}_{i-1,j}\right) + (1 - 4F_0)T^l{}_{i,j}$$

Sendo $F_0 = \alpha\Delta t/\Delta x^2$.

No entanto, se houver fluxo nas bordas, a variação de temperatura nas mesmas pode ser descrita pelas fórmulas:

Borda superior:

$$T_{0,j}{}^{l+1} = F_0(2T^l{}_{i+1,j} - 2\Delta y * \frac{\partial T_{0,j}{}^l}{\partial y} + T^l{}_{i,j+1} + T^l{}_{i,j-1} + (1 - 4F_0)T^l{}_{i,j}$$

Borda inferior:

$$T_{n,j}{}^{l+1} = F_0(2T^l{}_{i-1,j} - 2\Delta y * \frac{\partial T_{n,j}{}^l}{\partial y} + T^l{}_{i,j-1} + T^l{}_{i,j+1} + (1 - 4F_0)T^l{}_{i,j}$$

Borda esquerda:

$$T_{i,0}{}^{l+1} = F_0(2T^l{}_{i,j+1} - 2\Delta x * \frac{\partial T_{i,0}{}^l}{\partial x} + T^l{}_{i-1,j} + T^l{}_{i,j+1} + (1 - 4F_0)T^l{}_{i,j}$$

Borda direita:

$$T_{i,n}{}^{l+1} = F_0(2T^l{}_{i,j-1} - 2\Delta x * \frac{\partial T_{i,n}{}^l}{\partial x} + T^l{}_{i+1,j} + T^l{}_{i-1,j} + (1 - 4F_0)T^l{}_{i,j}$$

RESULTADOS E CONCLUSÕES

Desta maneira, para execução do programa e esperado os seguintes retornos:

- Valor do F_0. Caso esse valor seja maior do que 0.25 o programa irá apontar que o valor dessa variável é maior do que o esperado e, consequentemente, o resultado não convergirá.

```
[0.1
```

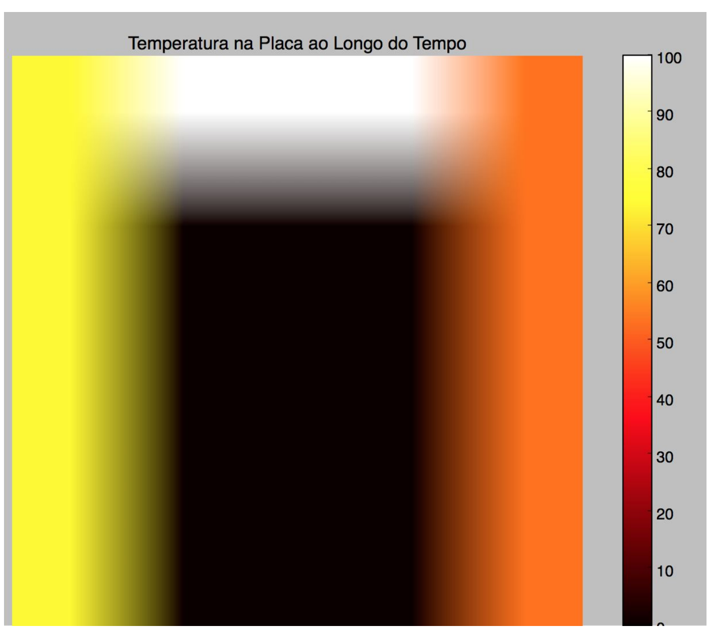
- Matriz de valores para a placa em seu estado inicial:

```
[75.0, 100.0, 100.0, 100.0, 50.0]

[75.0, 0, 0, 0, 50.0]

[75.0, 0, 0, 0, 50.0]

[75.0, 0, 0, 0, 50.0]

[75.0, 0.0, 0.0, 0.0, 50.0]
```

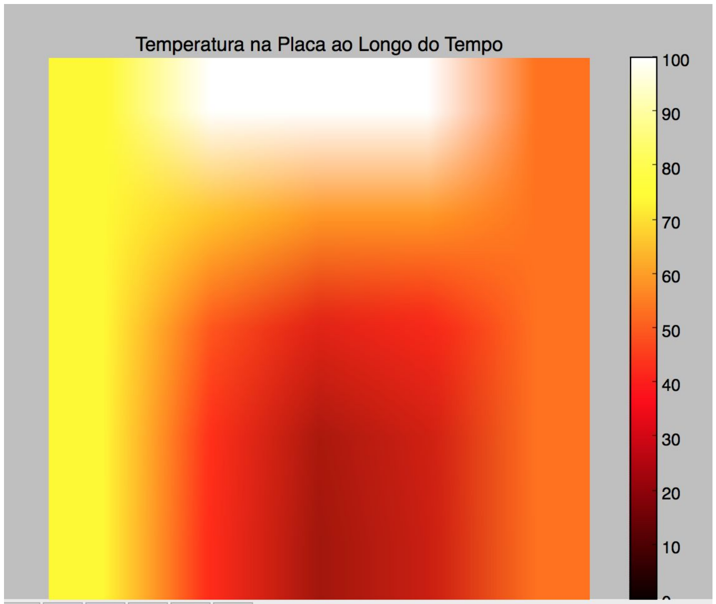
- Matriz de valores finais da placa, após o ensaio:

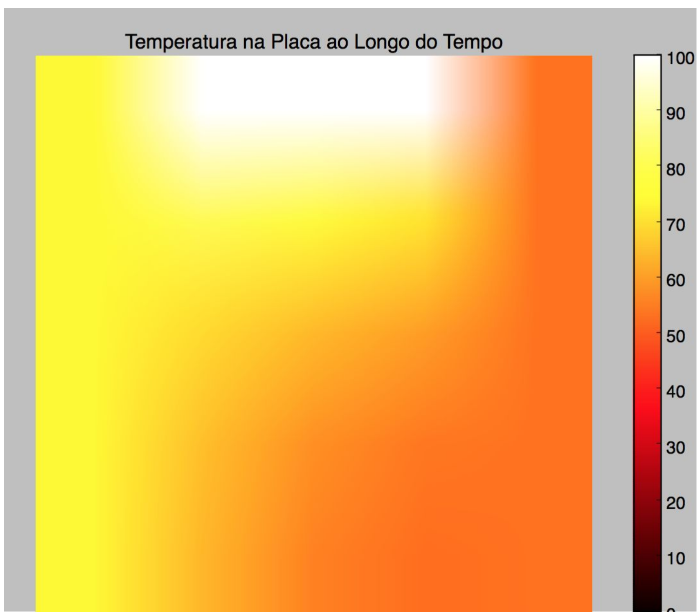
```
[75.0, 100.0, 100.0, 100.0, 50.0]
[75.0, 83.410925947007, 82.62860220845673, 74.26144141092452, 50.0]
[75.0, 76.01510157957131, 72.84204147589543, 64.4171634352414, 50.0]
[75.0, 72.80743889538284, 68.30729868031233, 60.56517085414572, 50.0]
[75.0, 71.90735532164776, 67.01454349582538, 59.53622130102919, 50.0]
```

- Finalmente, um gráfico animado abrirá em uma nova janela mostrando a evolução da temperatura na superfície no decorrer do tempo:



BURDEN,Richard L, FAIRES, J. Douglas; BURDEN, Annete M., **Análise Numérica**, 10ª edição, Cengage Learning, 2015, São Paulo.





REFERÊNCIAS

BURDEN,Richard L, FAIRES, J. Douglas; BURDEN, Annete M., **Análise Numérica**, 10ª edição, Cengage Learning, 2015, São Paulo.

KIUSALAAS, Jaan, **Numerical methods in engineering with Python 3,** Cambridge, 2013, Nova Iorque.s