

# TRANSFERÊNCIA DE CALOR E MECÂNICA DOS SÓLIDOS

## APS 2 : Aplicação do método de diferenças finitas

Alunos(a): Bernardo Cunha Capoferri

Guilherme Dantas Rameh

Henrique Martinelli Frezzatti

Lívia Sayuri Makuta

### DEFINIÇÃO DO PROBLEMA

A função deve receber:

- Dados do material: densidade do material, calor específico, condutividade térmica.
- Dados do ambiente: coeficiente de transferência por convecção, temperatura do fluido.
- Dados do dissipador: temperatura da base, raio, comprimento.
- Dados do modelo numérico:  $\Delta x$ , tempo total da simulação.

A função deve retornar: Um gráfico (*Temperatura  $\times$  posição*) com o resultado numérico. Um gráfico (*Temperatura  $\times$  posição*) com o resultado analítico. Use legendas e indique as unidades usadas.

Considere uma aleta de seção transversal circular e constante, com temperatura fixa na base e exposta ao ambiente em todas as outras superfícies. Podemos esboçar as entradas e saídas de energia em um volume de controle definido ao redor de uma seção como indica a Figura 1.

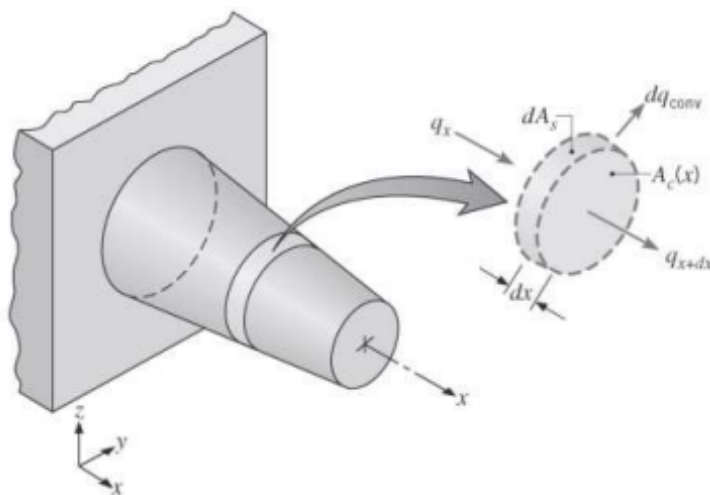


Figura 1: Balanço de energia em um volume de controle (Incropera,2008).

- $q_x$  taxa de transferência de calor por condução em  $x$ .
- $q_{x+dx}$  taxa de transferência de calor por condução em  $x+dx$ .
- $dq_{conv}$  taxa de transferência de calor por convecção através da área superficial do elemento diferencial.

Substituindo as equações de taxa  $q_x$ ,  $q_{x+dx}$  e  $dq_{conv}$  podemos obter a forma geral da equação da energia para uma superfície estendida, em regime transiente como mostrado na figura 2:

$$\frac{d^2T}{dx^2} - \frac{hP}{\kappa A_{tr}} (T - T_{\infty}) = \frac{1}{\alpha} \cdot \frac{dT}{dt}$$

Figura 2 - Forma geral da equação de energia para uma superfície estendida.

Onde  $\alpha$  é a difusividade do material,  $A_{tr}$  é a área da seção transversal da aleta,  $h$  é o coeficiente de transferência de calor por convecção,  $k$  é a condutividade térmica do material da aleta e  $P$  é o perímetro da seção da aleta.

## ROTEIRO DE ATIVIDADES

### Pergunta

1) Aplique os conceitos do método das diferenças finitas na equação diferencial do problema para definir uma equação de diferença algébrica associada aos nós da aleta. Considere para a segunda derivada da temperatura no espaço uma equação de diferença finita centrada e para a primeira derivada no tempo da temperatura uma equação de diferença progressiva. Apresente o desenvolvimento das equações.

### Resposta

Partindo da equação mostrada anteriormente na figura 2, substituímos os termos contínuos, isto é, as derivadas segunda e primeira por uma equação de diferença finita centrada e por uma equação de diferença progressiva, respectivamente. Ambas as equações foram vistas na aula 7.

Com essas mudanças feitas, isolamos a próxima temperatura no tempo, representada por  $T_i^{l+1}$ , para obter no fim um gráfico com os valores da temperatura ao longo do comprimento da aleta. Todos os passos dessa transformação podem ser vistos abaixo na figura 3:

$$\frac{d^2T}{dx^2} - \frac{hP}{\kappa A_{tr}}(T - T_{\infty}) = \frac{1}{\alpha} \cdot \frac{dT}{dt}$$

$$\frac{d^2T}{dx^2} = \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2}$$

$$\frac{dT}{dt} = \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

$$\frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} - \frac{hP}{\kappa A_{tr}}(T - T_{\infty}) = \frac{1}{\alpha} \cdot \frac{T_i^{l+1} - T_i^l}{\Delta t}$$

$$T_i^{l+1} = \left( \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} - \frac{hP}{\kappa A_{tr}}(T - T_{\infty}) \right) \cdot \alpha \cdot \Delta t + T_i^l$$

Figura 3 - Desenvolvimento da transformação feita na equação original para implementar a resolução do problema através de um método numérico.

### Pergunta

2. Apresente um gráfico (*Temperatura × posição*) após obter o regime permanente. Use uma tolerância de  $tol = 10^{-10}$ . Considere que a aleta, de comprimento  $L = 300mm$ , densidade de  $2700 \frac{kg}{m^3}$ , condutividade térmica de  $180 \frac{W}{mK}$  e calor específico  $896 \frac{J}{kgK}$ , tem a temperatura especificada na extremidade de  $T(L) = 25^\circ C$ . A aleta está em contato com uma base a  $T_b = 100^\circ C$  em um ambiente a  $T_{\infty} = 50^\circ C$  e  $h = 50 \frac{W}{m^2 K}$ . Para a condição de estabilidade do procedimento numérico use  $\Delta t \leq \frac{\Delta x^2}{\alpha \cdot (\frac{hP\Delta x^2}{\kappa A_{tr}}) + 2}$

### Resposta

Primeiro começamos definindo os parâmetros passados pelo exercício, e para isso também precisamos importar algumas bibliotecas. Essa etapa pode ser vista na célula abaixo:

In [1]:

```

from math import *
import numpy as np
import matplotlib.pyplot as plt

# tolerância de erro:
tol = 1e-10

# Comprimento da aleta:
L = 300e-3 # [m]

# Raio da aleta:
r = 3e-3 # [m]

# Densidade da aleta:
rho = 2700 # [kg/ m^3]

# Condutividade térmica:
k = 180 # [W/(mK)]

# Calor específico:
cp = 896 # [J/(kgK)]

# Termo alfa:
alfa = k/(rho*cp)

# Temperatura especificada na extremidade:
T_L = 25 # 25 [°C]

# Temperatura da base:
T_b = 100 # 100 [°C]

# Temperatura do ambiente/fluido:
T_inf = 50 # 50 [°C]

# Coeficiente de convecção:
h = 50 # [W/(m^2K)]

# Variação em x:
delta_x = 1e-3 # [m]

#Tempo total da simulação
t_total = 10000 # [s]

```

Observação: definimos o raio da aleta como  $10^{-3}$  metros pois não foi dado pelo exercício e é um valor extremamente pequeno se comparado com o comprimento da aleta, o que nos permite adotar uma resolução unidimensional.

Assim, com todos os parâmetros definidos, montamos uma função chamada **calcula\_T\_numerico** que recebe como entrada o delta x (intervalo adotado entre a posição atual e a próxima posição no eixo x da aleta em metros), o raio [m], o coeficiente de convecção térmica  $[\frac{W}{m^2 K}]$ , a condutividade térmica  $[\frac{W}{mK}]$ , a temperatura da base [°C], a temperatura do ambiente/fluido [°C], o termo alfa (divisão de  $k$  por densidade e calor específico) e o tempo total requerido pela simulação [s] a fim de plotar o gráfico de temperatura [°C] por posição [m].

Primeiro começamos calculando o delta t, já que para manter a estabilidade esse intervalo de tempo deveria ser  $\leq \frac{\Delta x^2}{\alpha \cdot (\frac{hP\Delta x^2}{k A_{Tr}}) + 2}$ . E para garantirmos isso, usamos 99% de  $\frac{\Delta x^2}{\alpha \cdot (\frac{hP\Delta x^2}{k A_{Tr}}) + 2}$ . Além disso, calculamos a área de transferência de calor e o perímetro da área de seção transversal da aleta.

Feito isso, utilizamos o método das diferenças finitas que é um método aproximado para determinar a temperatura em pontos discretos (nodal) do sistema físico. Para isso começamos preenchendo uma matriz com zeros, sendo essa matriz determinada pelo número de pontos no tempo e número de pontos no eixo x (número total de todos os pequenos intervalos ao longo do comprimento da aleta). Após feito isso, definimos as condições de contorno que se resumem ao fato da temperatura da base ser a mesma e nunca mudar (= 100°C), e também ao fato da temperatura da extremidade da aleta ser a mesma ao longo do tempo (= 25°C).

Depois, para calcular a temperatura em cada um dos pontos selecionados no comprimento da aleta, utilizamos a equação descrita na figura 2 e fomos adicionando essas temperaturas em uma lista. Com a lista de todos os pontos ao longo do comprimento da aleta e das temperaturas associadas a cada um desses pontos, foi possível então plotar o gráfico da temperatura [°C] ao longo do comprimento da aleta [m] e isso tudo ao longo do tempo - que utilizamos 1000 segundos.

In [2]:

```

def calculo_T_numerico(delta_x, r, h, k, T_extremidade, T_b, Tinf, alfa, t_total, tol):
    lista_erro = []
    # Cálculo da área de transferência de calor na aleta.
    Atr = pi*(r**2)

    # Cálculo do perímetro da área de seção transversal da aleta.
    P = 2*pi*r

    # Cálculo do delta t a fim de manter a estabilidade.
    den_delta_t = ((h*P*(delta_x**2))/(k*Atr))+2
    delta_t = ((delta_x**2)/(alfa*den_delta_t))*0.99

    # Número de passos no tempo, isto é, tempo total dividido por delta x.
    n_tempo = len(np.arange(0, t_total, delta_t))

    # Número de pontos ao longo do comprimento da aleta.
    n_x = int(L/delta_x) + 1

    # Montando a matriz de zeros:
    TM = np.zeros((n_tempo, n_x))

    # Condições de contorno:
    # Temperatura na base ao longo do tempo é a mesma.
    TM[:, 0] = T_b

    # Temperatura na extremidade ao longo do tempo é a mesma.
    TM[:, n_x-1] = T_extremidade

    tempo_tol_atingida = 0
    for l in range(0, n_tempo-1):
        for i in range(1, n_x-1):
            termo1 = (TM[l, i+1] - 2*TM[l, i] + TM[l, i-1])/(delta_x**2)
            termo2 = (h*P*(TM[l, i]-Tinf))/(k*Atr)
            TM[l+1, i] = (TM[l, i] + alfa*delta_t*(termo1 - termo2))

        erro_final = np.amax(abs((TM[l+1, 1:-1]-TM[l, 1:-1])/TM[l+1, 1:-1]))
        lista_erro.append(erro_final)

        if tol >= erro_final:
            print('\n\nConvergiu!', l)
            tempo_tol_atingida = l

            break

    instante = n_tempo
    x = np.linspace(0.0, L, n_x)
    T = np.array(TM[tempo_tol_atingida, :])

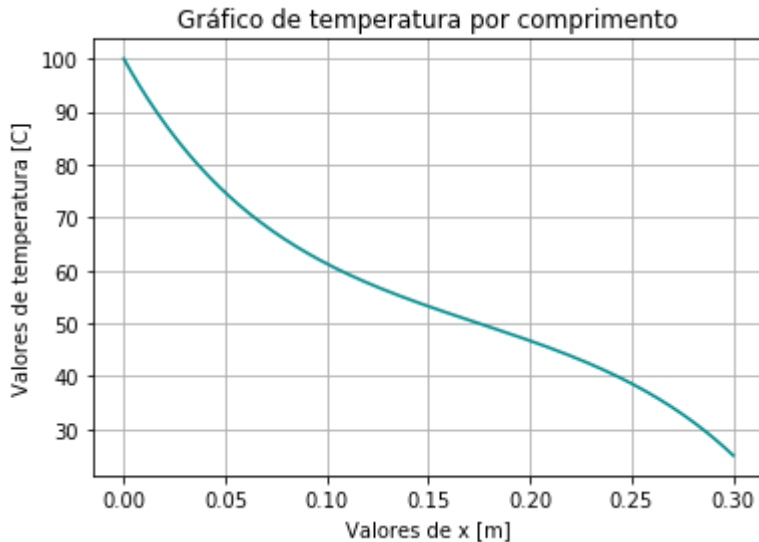
    plt.plot(x, T, color="darkcyan")
    plt.title("Gráfico de temperatura por comprimento")
    plt.xlabel("Valores de x [m]")
    plt.ylabel("Valores de temperatura [C]")
    plt.grid()
    plt.show()

```

In [3]:

```
calculo_T_numerico(delta_x, r, h, k, T_L, T_b, T_inf, alfa, t_total, tol)
```

Convergiu! 99289



### Pergunta

3) Para validar os resultados obtidos com o software desenvolvido, apresente um gráfico (*Temperatura × posição*) para o comportamento analítico da temperatura na aleta usando as mesmas condições definidas no item anterior.

### Resposta

Para montar um gráfico que mostra o comportamento analítico da temperatura na aleta, começamos analisando qual seria o caso que representa melhor a distribuição de temperatura no objeto estudado. De acordo com as condições de contorno que foram providas pelo exercício, o melhor caso seria o C, isto é, o caso em que a distribuição de temperatura e a transferência de calor através da aleta estão restringidos a uma temperatura especificada na extremidade da aleta. Isso porque foi definido que  $T(L) = 25^{\circ}\text{C}$  ao longo de todos os pontos no intervalo de tempo estabelecido.

Assim, usando a fórmula de distribuição de temperatura apresentada na aula 6 e manipulando-a a fim de calcular a temperatura para todos os pontos ao longo do comprimento da aleta, foi possível montar o gráfico pedido no exercício. A manipulação da fórmula encontra-se abaixo na figura 4:

$$\frac{T - T_{\infty}}{T_b - T_{\infty}} = \frac{\left(\frac{\theta_L}{\theta_b}\right) \cdot \sinh(m \cdot x) + \sinh(m(L - x))}{\sinh(m \cdot L)}$$

$$\theta_L = T(L) - T_{\infty}$$

$$\theta_b = T(0) - T_{\infty}$$

$$T = \frac{\left(\frac{T(L) - T_{\infty}}{T(0) - T_{\infty}}\right) \cdot \sinh(m \cdot x) + \sinh(m(L - x))}{\sinh(m \cdot L)} \cdot (T_b - T_{\infty}) + T_{\infty}$$

Figura 4 - Manipulação da fórmula de distribuição de temperatura apresentada na aula 6.



In [4]:

```

# Resolução analítica

def calculo_T_analitico(delta_x, r, h, k, T_L, T_b, T_inf, alfa, t_total):
    # Cálculo da área de transferência de calor na aleta.
    Atr = pi*(r**2)

    # Cálculo do perímetro da área de seção transversal da aleta.
    P = 2*pi*r

    # Cálculo do termo m
    m = sqrt((h*P)/(k*Atr))

    # Cálculo do delta t a fim de manter a estabilidade.
    den_delta_t = ((h*P*(delta_x**2))/(k*Atr))+2
    delta_t = ((delta_x**2)/(alfa*den_delta_t))*0.99

    # Número de passos no tempo, isto é, tempo total dividido por delta x.
    n_passos = len(np.arange(0, t_total, delta_t))

    # Número de pontos ao longo do comprimento da aleta.
    lista_x = np.linspace(0, L, n_passos)
    listaTemp = []

    #  $T[i] = (((T[L] - T_{inf})/(T_b - T_{inf}))*sinh(m*x) + sinh(m*(L-x)))/(sinh(m*L)))*(T_b$ 

    for x in lista_x:
        teta_l_teta_b = ((T_L - T_inf)/(T_b - T_inf))
        numerador = teta_l_teta_b*sinh(m*x) + sinh(m*(L-x))
        denominador = sinh(m*L)
        fracao = (numerador/denominador)*(T_b - T_inf) + T_inf
        listaTemp.append(fracao)

    plt.plot(lista_x, listaTemp, color="red")
    plt.title("Gráfico de T(x) por x")
    plt.xlabel("Valores de x [de 0 até L em metros]")
    plt.ylabel("Valores de T(x) em Celsius")
    plt.grid()

```

In [5]:

```
calculo_T_analitico(delta_x, r, h, k, T_L, T_b, T_inf, alfa, t_total)
```

