



UNIVERSIDADE DO ESTADO  
DO RIO DE JANEIRO  
Faculdade de Engenharia  
Departamento de Engenharia Mecânica

TRABALHO DE TRANSMISSÃO DE CALOR II SOBRE

---

## Trocadores de calor

**Alunos :**

Daniel Lessa COELHO

Douglas AMARAL

**Prof. :**

Gustavo RABELLO

13 de Março de 2018

Trabalho realizado para a disciplina de Transmissão de Calor II como segunda avaliação.

# Sumário

<b>I</b>	<b>Tipos de Trocadores</b>	<b>1</b>
<b>1</b>	<b>Problema 1 : <i>Cálculo das temperaturas de saída</i></b>	<b>1</b>
	<b>Correntes Paralelas</b>	<b>1</b>
1.1	Esquema e malha	2
1.2	Código em <i>Python</i>	3
1.3	<i>Solução numérica</i>	6
	<b>Contra-Corrente</b>	<b>7</b>
1.4	Esquema e malha	7
1.5	Código em <i>Python</i>	8
1.6	<i>Solução numérica</i>	11
	<b>Comparações</b>	<b>11</b>
<b>2</b>	<b>Problema 2 : <i>Dimensionamento de um Trocador</i></b>	<b>13</b>
2.1	Código em <i>Python</i>	13
2.2	Resultado : <i>analítico vs. numérico</i>	16
<b>3</b>	<b>Dificuldades</b>	<b>16</b>

## Lista de Figuras

1	Esquema de trocador de calor.	2
2	Malha do trocador de calor.	2
3	Distribuição de temperatura de um trocador de calor de correntes paralelas.	6
4	Esquema de trocador de calor contra corrente.	7
5	Malha do trocador de calor contracorrente.	7
6	Distribuição de temperatura de um trocador de calor contra-corrente.	11
7	Distribuição de temperatura nos trocadores paralelo (a) e contra corrente (b).	12

## Parte I

# Tipos de Trocadores

Os trocadores de calor são dispositivos que permitem a troca de calor entre dois fluidos escoando a temperaturas distintas. Essa troca se dá através da parede sólida do tubo que separa os escoamentos e é em função do sentido destes que se classificam os trocadores, concêntricos e de um passe, estudados a seguir : os de corrente paralela e os contra correntes.

## 1 Problema 1 : *Cálculo das temperaturas de saída*

Neste exemplo, deseja-se calcular as temperaturas de saída dos fluido de processo ( $T_{saip}$ ) e serviço ( $T_{saís}$ ) conhecendo algumas caracteíricas do trocador de correntes paralelas :

- Coeficiente global de transfviadoerência de calor  $U = 1200[J]/[s][m^2][K]$
- Comprimento do trocador  $L = 150[m]$
- Diâmetro do tudbo do trocador  $D = 0.015[m]$

Além das características do trocador, sabe-se também :

Água - fluido de serviço;

- Temperatura de entrada  $T_{ents} = 110[K]$
- Capacidade térmica  $cp_s = 4180[J]/[kg][K]$
- Fluxo de massa  $\dot{m}_s = 2.0[kg]/[s]$

Químico - fluido de processo;

- Temperatura de entrada  $T_{entp} = 20[K]$
- Capacidade térmica  $cp_p = 1800[J]/[kg][K]$
- Fluxo de massa  $\dot{m}_p = 3.0[kg]/[s]$

## Correntes Paralelas

Nessa configuração, os fluidos quente e frio entram na mesma extremidade ( $x = 0$ ) do trocador e o deixam, também, na mesma extremidade ( $x = L$ ). Portanto, o escoamento de ambos ocorrem no *mesmo sentido*.

### 1.1 Esquema e malha

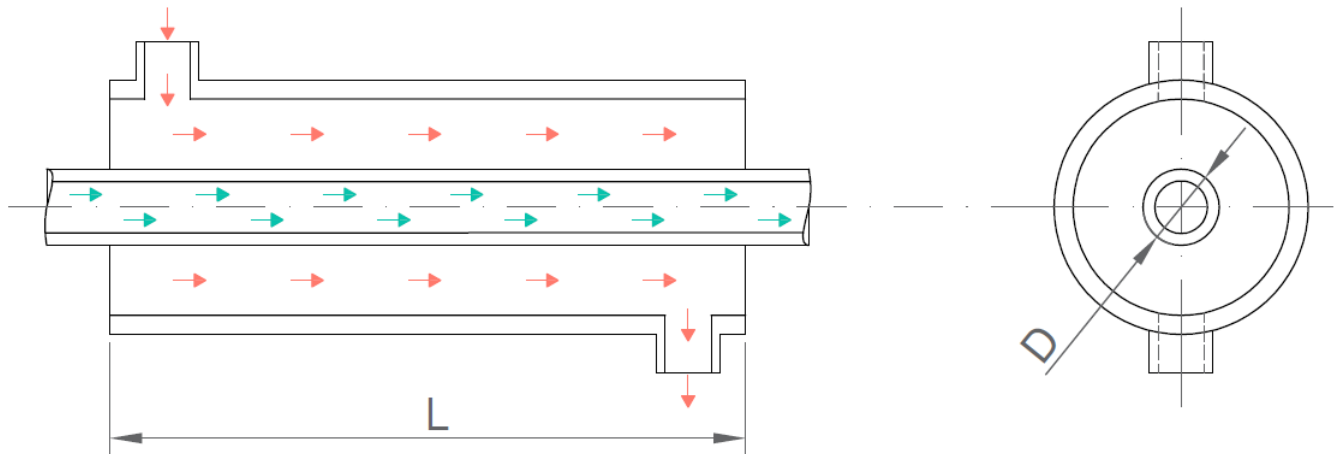


Figura 1 Esquema de trocador de calor.

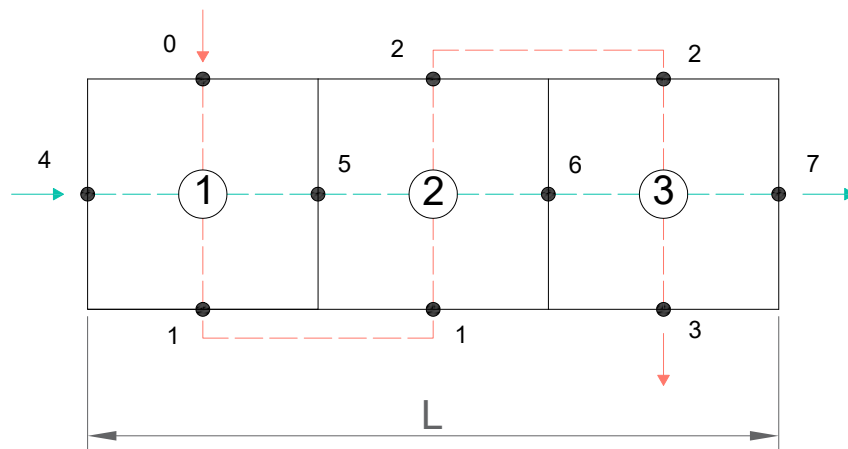


Figura 2 Malha do trocador de calor.

## 1.2 Código em Python

```
1 # -*- coding: utf-8 -*-
2 #Trabalho Trocador de Calor
3 # Daniel Coelho & Douglas Lopes
4 #-----#
5 #           Importando as bibliotecas necessarias           #
6 #-----#
7 import matplotlib.pyplot as plt
8 import matplotlib
9 import matplotlib.lines as mlines
10 import sympy
11 import numpy as np
12 #-----#
13 #           Informacoes do problema           #
14 #-----#
15 U=1200.      #[J/s*m**2*K] coeficiente global de transferencia de ←
    calor
16 L=150.      #[m] comprimento do trocador
17 D=0.015    #[m] diametro do tubo
18 #-----#
19 #           Fluxo de massa           #
20 #-----#
21 mff=3.      #[kg/s] Fluido frio
22 mfq=2.      #[kg/s] Fluido quente
23 #-----#
24 #           Capacidade termica           #
25 #-----#
26 cpff=1800. #[J/kg*K] Fluido frio
27 cpfq=4180. #[J/kg*K] Fluido quente
28 #-----#
29 #           Temperaturas           #
30 #-----#
31 Tff=20.    #[K] Fluido frio
32 Tfq=110.   #[K] Fluido quente
33 #-----#
34 #           Informacoes da Malha           #
35 #-----#
36 nel= int(input("Qual o numero de elementos? "))
37 npt = int(nel*2+2)
38 npo=int(npt/2.)
39 #-----#
40 #           Geometrias           #
41 #-----#
42 dl=L*1.0/nel
```

```

43 A=sympy.pi*D*L
44 dA=sympy.pi*D*d1
45 #-----#
46 #           Fluxo de massa * Capacidade Termica           #
47 #-----#
48 mcpf=mff*cpff
49 mcpq=mfq*cpfq
50 #-----#
51 #           Matriz Q           #
52 #-----#
53 Q=np.zeros((npt,1), dtype='float')
54 Q[0]=-(mcpq)*(Tfq)
55 Q[(nel+1)]=-(mcpf)*(Tff)
56 #-----#
57 #           Matriz IEN Paralelo           #
58 #-----#
59 IENP=np.zeros((nel,4), dtype='int')
60 for i in range(1,nel+1):
61     IENP[i-1,0]=i-1
62     IENP[i-1,1]=i
63     IENP[i-1,2]=(npt/2.)-1+i
64     IENP[i-1,3]=(npt/2.) +i
65 #-----#
66 #           Matriz Local           #
67 #-----#
68 klc=np.zeros((4,4), dtype='float')
69 klc[0,0]=-mcpq
70 klc[1,0]=mcpq -(U*dA/2.)
71 klc[1,1]=-U*dA/2.
72 klc[1,2]=U*dA/2.
73 klc[1,3]=U*dA/2.
74 klc[2,2]=-mcpf
75 klc[3,0]=U*dA/2.
76 klc[3,1]=U*dA/2.
77 klc[3,2]=mcpf -(U*dA/2.)
78 klc[3,3]=-(U*dA/2.)
79 #-----#
80 #           Matriz Global           #
81 #-----#
82 K=np.zeros((npt,npt), dtype='float')
83 for elem in range(nel):
84     for i in range(4):
85         igb=IENP[elem-1,i-1]
86         for j in range(4):
87             jgb=IENP[elem-1,j-1]

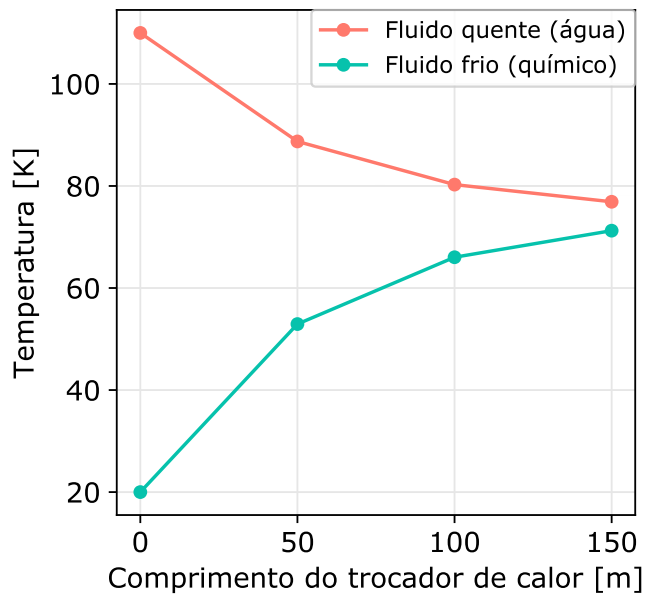
```

```
88         K[igb,jgb]= K[igb,jgb]+ klc[i-1,j-1]
89
90 K[nel,nel]=K[nel,nel]-mcpq
91 K[npt-1,npt-1]=K[npt-1,npt-1]-mcpf
92 #-----#
93 #           Solucao Temperaturas T           #
94 #-----#
95 temp=np.zeros((npt,1), dtype='float')
96 temp=np.linalg.solve(K, Q)
97 print('Solucao Temperaturas T: \n{}\n' .format(temp))
98 #-----#
99 #           Quantidade trocada Q           #
100 #-----#
101 Qtrocada=float(mcpf*(temp[npt-1]-temp[npo]))/1000.
102 print('Quantidade de calor trocada e: '+str(Qtrocada)+' kW')
103 #-----#
104 dx=float(L/nel)
105 x=np.zeros((int(npo),1), dtype='float')
106 for i in range (1,int(npo)):
107     x[i]=x[i-1]+dx
108 o = np.arange(0.0, 82, 1) # markersize params
109 #-----#
110 #           Plotagem           #
111 #-----#
112 fig, ax = plt.subplots(1, 1, figsize←
113     =(4.05511811023622,4.05511811023622))
114 line1, = plt.plot(x,temp[0:npo:+1], linestyle='-',marker='', color=←
115     '#ff796c',markersize=o[5], label='Fluido quente (agua)')
116 line2, = plt.plot(x,temp[npo:npt:+1], linestyle='-',marker='', ←
117     color='#06c2ac', markersize=o[5], label='Fluido frio (quimico)')
118 plt.tick_params(axis='both', labels=12)
119 plt.xlabel('Comprimento do trocador de calor [m]', fontsize=12)
120 plt.ylabel('Temperatura [K]', fontsize=12)
121 plt.legend(title='',loc=0, borderaxespad=0., fontsize=10)
122 plt.grid(b=True, which='both', color='0.9',linestyle='-')
123 plt.show()
124 fig.savefig('nomedafigura.pdf', format='pdf', dpi=1200, bbox_inches←
125     ='tight', transparent=True)
```

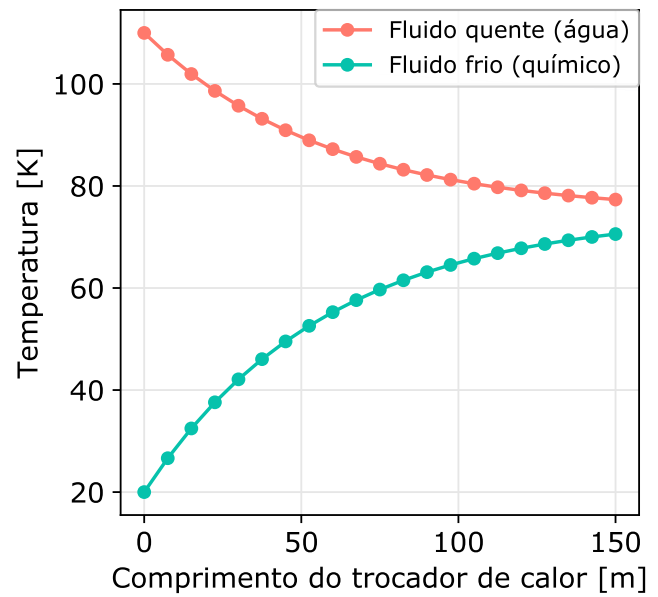


### 1.3 Solução numérica

Abaixo, os gráficos obtidos pela biblioteca *matplotlib* através do python.



(a) 3 elementos.



(b) 20 elementos.

Figura 3 Distribuição de temperatura de um trocador de calor de correntes paralelas.

## Contra-Corrente

Nessa configuração, os fluidos quente e frio entram e deixam o trocador em extremidades opostas. Portanto, o escoamento de ambos ocorrem em *sentidos contrários*.

### 1.4 Esquema e malha

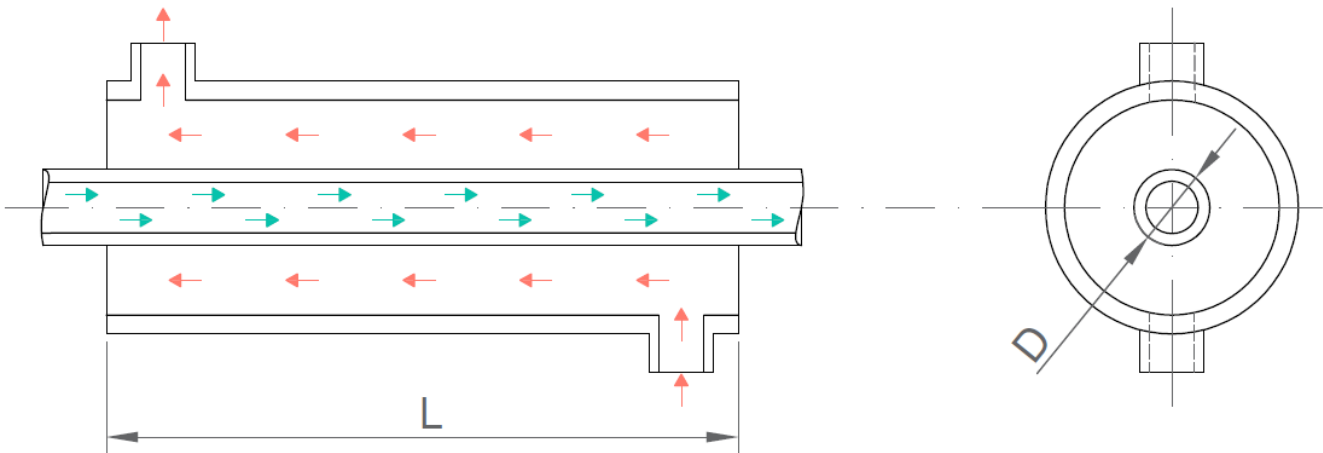


Figura 4 Esquema de trocador de calor contra corrente.

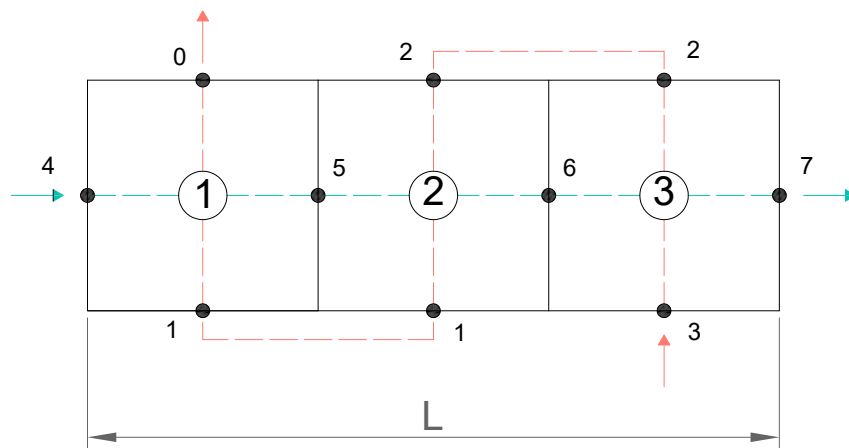


Figura 5 Malha do trocador de calor contracorrente.

## 1.5 Código em Python

```

1 # -*- coding: utf-8 -*-
2 #Trabalho Trocador de Calor
3 # Daniel Coelho & Douglas Lopes
4 #-----#
5 #           Importando as bibliotecas necessarias           #
6 #-----#
7 import matplotlib.pyplot as plt
8 import matplotlib
9 import matplotlib.lines as mlines
10 import sympy
11 import numpy as np
12 #-----#
13 #           Informacoes do problema           #
14 #-----#
15 U=1200.      #[J/s*m**2*K] coeficiente global de transferencia de ←
    calor
16 L=150.      #[m] comprimento do trocador
17 D=0.015     #[m] diametro do tubo
18 #-----#
19 #           Fluxo de massa           #
20 #-----#
21 mff=3.      #[kg/s] Fluido frio
22 mfq=2.      #[kg/s] Fluido quente
23 #-----#
24 #           Capacidade termica           #
25 #-----#
26 cpff=1800.  #[J/kg*K] Fluido frio
27 cpfq=4180.  #[J/kg*K] Fluido quente
28 #-----#
29 #           Temperaturas           #
30 #-----#
31 Tff=20.     #[K] Fluido frio
32 Tfq=110.    #[K] Fluido quente
33 #-----#
34 #           Informacoes da malha           #
35 #-----#
36 nel= int(input("Qual o numero de elementos? "))
37 npt = int(nel*2+2)
38 npo=int(npt/2.)
39 print('Numero de pontos: ' + str(npo))
40 #-----#
41 #           Fluxo de massa * Capacidade termica           #
42 #-----#

```

```
43 mcpf=mff*cpff
44 mcpq=mfq*cpfq
45 #-----#
46 #           Matriz Q           #
47 #-----#
48 Q=np.zeros((npt,1), dtype='float')
49 Q[nel]=-(mcpq)*(Tfq)
50 Q[npo]=-(mcpf)*(Tff)
51 #-----#
52 #           Geometrias           #
53 #-----#
54 dl=L*1.0/nel
55 A=sympy.pi*D*L
56 dA=sympy.pi*D*dl
57 #-----#
58 #           Matriz IEN Contra Corrente           #
59 #-----#
60 IENCC=np.zeros((nel,4), dtype='int')
61 for i in range(1,nel+1):
62     IENCC[i-1,0]=i
63     IENCC[i-1,1]=i-1
64     IENCC[i-1,2]=(npt/2.)-1+i
65     IENCC[i-1,3]=(npt/2.) +i
66 #-----#
67 #           Matriz Local           #
68 #-----#
69 klc=np.zeros((4,4), dtype='float')
70 klc[0,0]=-mcpq
71 klc[1,0]=mcpq -(U*dA/2.)
72 klc[1,1]=-U*dA/2.
73 klc[1,2]=U*dA/2.
74 klc[1,3]=U*dA/2.
75 klc[2,2]=-mcpf
76 klc[3,0]=U*dA/2.
77 klc[3,1]=U*dA/2.
78 klc[3,2]=mcpf-(U*dA/2.)
79 klc[3,3]=-(U*dA/2.)
80 #-----#
81 #           Matriz Global           #
82 #-----#
83 K=np.zeros((npt,npt), dtype='float')
84 for elem in range(nel):
85     for i in range(4):
86         igb=IENCC[elem-1,i-1]
87         for j in range(4):
```

```

88         jgb=IENCC[elem-1,j-1]
89         K[igb,jgb]= K[igb,jgb]+ klc[i-1,j-1]
90
91 K[0,0]=K[0,0]-mcpq
92 K[npt-1,npt-1]=K[npt-1,npt-1]-mcpf
93 #-----#
94 #           Solucao do problema           #
95 #-----#
96 temp=np.zeros((npt,1), dtype='float')
97 temp=np.linalg.solve(K, Q)
98 print(temp)
99 #-----#
100 #           Calor Trocado           #
101 #-----#
102 Qtrocada=float(mcpf*(temp[npt-1]-temp[mpo]))/1000.
103 print('Quantidade de calor trocada e: '+str(Qtrrocada)+' kW')
104 #-----#
105 #           Vetor posicao           #
106 #-----#
107 dx=float(L/nel)
108 x=np.zeros((int(npo),1), dtype='float')
109 for i in range (1,int(npo)):
110     x[i]=x[i-1]+dx
111 o = np.arange(0.0, 82, 1)
112 #-----#
113 #           Plot do grafico           #
114 #-----#
115 fig, ax = plt.subplots(1, 1, figsize←
    =(4.05511811023622,4.05511811023622)) # 8,6 padrao
116 line1 = plt.plot(x,temp[0:npo:+1], linestyle='-',marker='', color='←
    #ff796c',markersize=o[5], label='Fluido quente (agua)')
117 line2 = plt.plot(x,temp[mpo:npt:+1], linestyle='-',marker='', color←
    '#06c2ac', markersize=o[5], label='Fluido frio (quimico)')
118 plt.tick_params(axis='both', labelsize=12)
119 plt.xlabel('Comprimento do trocador de calor [m]', fontsize=12)
120 plt.ylabel('Temperatura [K]', fontsize=12)
121 plt.legend(title='',loc=0, borderaxespad=0., fontsize=10)
122 plt.grid(b=True, which='both', color='0.9',linestyle='-')
123 plt.show()
124 fig.savefig('nomedafigura.pdf', format='pdf', dpi=1200, bbox_inches←
    ='tight', transparent=True)

```

## 1.6 Solução numérica

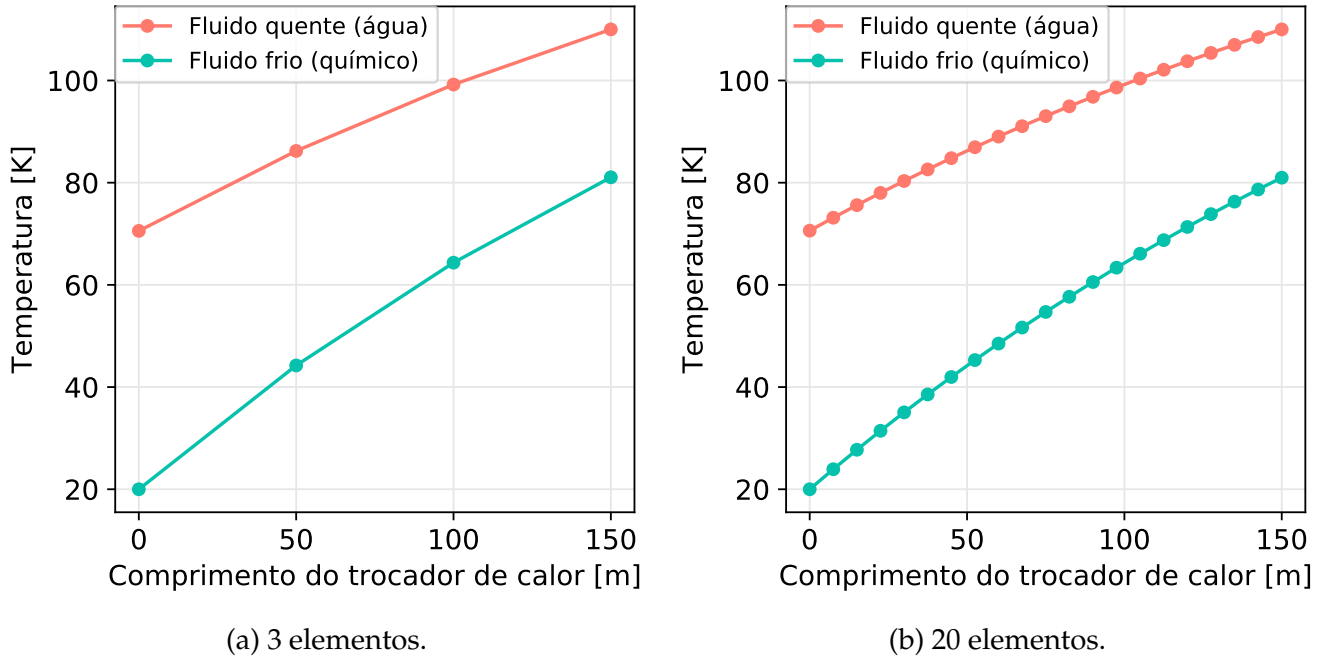


Figura 6 Distribuição de temperatura de um trocador de calor contra-corrente.

## Comparações

Embora o cálculo de  $\Delta T_{lmt\Delta}$  só seja possível se nós conhecermos, previamente, três temperaturas e/ou, equivalentemente, a troca de calor  $Q$  envolvida no processo, podemos comparar a eficiência desses dois tipos de trocadores. As perdas foram desconsideradas nos problemas acima e portanto, a quantidade de calor  $Q_{trocada}$  cedida pelo fluido de trabalho (água) para o fluido de processo (químico) pode ser calculada como :

$$\dot{Q}_{trocada} = \underbrace{\dot{m}_{ff} c_{pff} (T_{out/ff} - T_{ff})}_{\text{Fluido de processo}} = \underbrace{\dot{m}_{fq} c_{pfq} (T_{fq} - T_{out/fq})}_{\text{Fluido de serviço}} \quad (1.1)$$

Onde :

$$T_{out/ff} = \text{temp}[npt-1] \text{ (ambos)}$$

$$T_{out/fq} = \text{temp}[ne1] \text{ (paralelo)} \text{ e } T_{out/fq} = \text{temp}[0] \text{ (contracorrente)}$$

Utilizando a equação 1.1 e as dados explicitados no código para os dois casos, temos :

Paralelo :

$$\dot{Q}_{trocada/P} = 273.020313355 \text{ kW}$$

Contra-corrente :

$$\dot{Q}_{trocada/CC} = 329.28785435 \text{ kW}$$

Eficiência

$$\eta = \frac{\dot{Q}_{\text{trocada/CC}}}{\dot{Q}_{\text{trocada/P}}} - 1 \approx 0.2060928738 \approx 20,6\%$$

Os resultados nos levam a concluir que o trocador contra corrente fornece uma quantidade de calor trocada maior para um mesmo  $L$ , ou seja, é em torno de 20,6% mais eficiente que o trocador de calor de correntes paralelas.

Os valores foram retirados da solução numérica de ambos os casos com discretização em 1000 elementos. Apenas a título de precisão, abaixo está o gráfico :

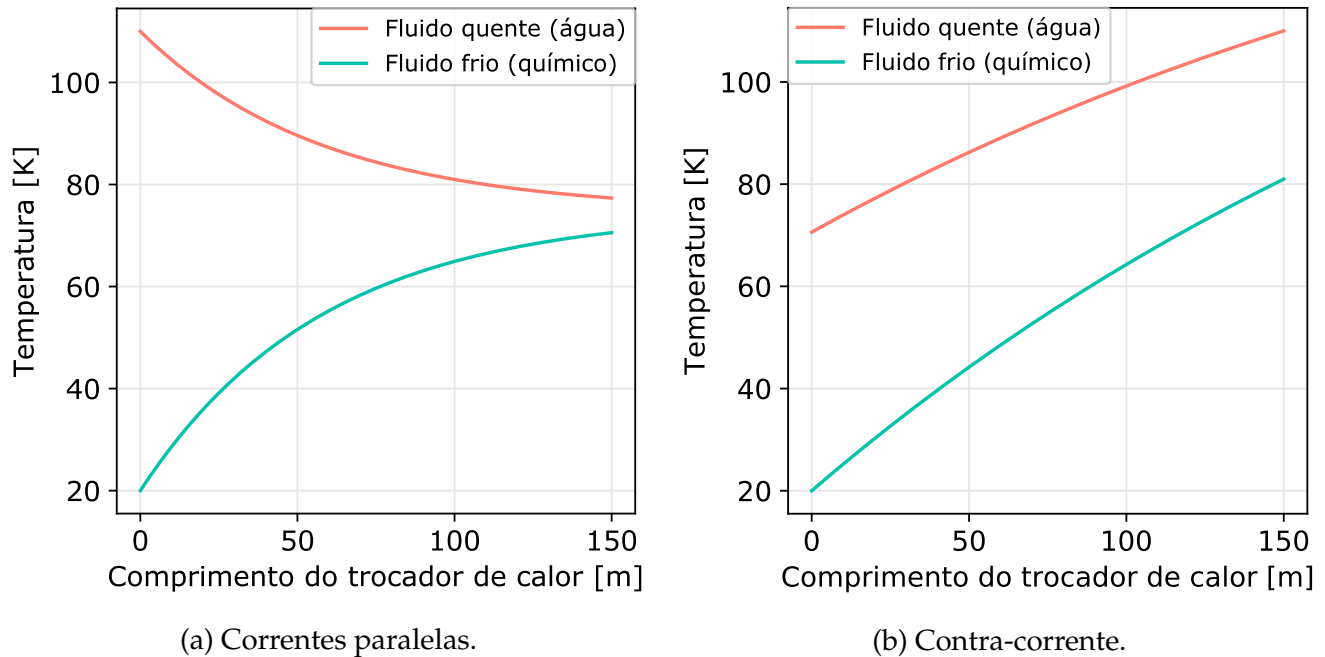


Figura 7 Distribuição de temperatura nos trocadores paralelo (a) e contra corrente (b).

## 2 Problema 2 : Dimensionamento de um Trocador

Um trocador de calor contra corrente casco-tubo (tubo duplo) deve aquecer a água de 20°C a 80°C a uma vazão mássica  $\dot{m} = 1.2\text{kg/s}$ . O aquecimento é obtido por água geotérmica disponível à 160°C com vazão mássica  $\dot{m} = 2\text{kg/s}$ . O tubo interno tem uma parede fina de diâmetro  $D = 1.5\text{cm}$ . Considerando que o coeficiente global de transferência de calor do trocador é  $U = 640\text{W/m}^2\text{K}$ , determine o comprimento  $L$  do trocador de calor necessário para alcançar o aquecimento desejado e obtenha a distribuição de temperatura dos dois fluidos ao longo do trocador de calor através de análise discreta.

- $D = 0.015[\text{m}]$
- $U = 640\text{W/m}^2\text{K}$

Água - fluido de trabalho;

- $T_{entt} = 20^\circ\text{C}$
- $T_{sait} = 80^\circ\text{C}$
- $\dot{m}_t = 1.2[\text{kg}]/[\text{s}]$
- $cp_t = 4180[\text{J}]/[\text{kg}][\text{K}]$

Água geotérmica - fluido de serviço;

- $T_{ents} = 160^\circ\text{C}$
- $cp_s = 4310[\text{J}]/[\text{kg}][\text{K}]$
- $\dot{m}_s = 2.0[\text{kg}]/[\text{s}]$

### 2.1 Código em Python

A equação para o cálculo analítico de  $L$  considerando os dados do problema segue abaixo :

$$Q_{trocada} = UA\Delta T_{lmt\dot{d}} \quad \text{onde} \quad A = \pi DL \quad \Rightarrow \quad L = \frac{Q_{trocada}}{U\pi D\Delta T_{lmt\dot{d}}} \quad (2.1)$$

Na resolução desse exercício, tomamos a liberdade de excluir os comentários que apenas indicariam os *inputs*, funções e matrizes apresentadas nos códigos anteriores. A versão simplificada é:

```

1 # -*- coding: utf-8 -*-
2 #Trabalho Trocador de Calor
3 # Daniel Coelho & Douglas Lopes
4 #import sys
5 #reload(sys)
6 #sys.setdefaultencoding('utf-8')
7 from datetime import datetime
8 start_time = datetime.now()
9 from __future__ import unicode_literals

```



```

10 import matplotlib.pyplot as plt
11 import matplotlib
12 import sympy
13 import numpy as np
14 U=640.
15 L=np.random.randint(100, 200)
16 print('Comprimento L escolhido aleatoriamente para inicio das ←
      iteracoes: {} m' .format(L))
17 D=0.015
18 mff=1.2
19 mfq=2.
20 cpff=4310.
21 cpfq=4180.
22 Tff=20.+273.
23 Tfq=160.+273.
24 Toutff=80.+273.
25 nel= int(input("Qual o numero de elementos? "))
26 npt = int(nel*2+2)
27 npo=int(npt/2.)
28 print('Numero de pontos da malha: ' + str(npt))
29 mcpf=mff*cpff
30 mcpq=mfq*cpfq
31 Q=np.zeros((npt), dtype='float')
32 Q[nel]=-(mcpq)*(Tfq)
33 Q[npo]=-(mcpf)*(Tff)
34 dT = 1.
35 n_it = int(0)
36 m = int(10**4)
37 while abs(dT)>10**(-8) and n_it<m:
38     n_it += 1
39     dl=float(L/nel)
40     A=sympy.pi*D*L
41     dA=sympy.pi*D*dl
42     IENCC=np.zeros((nel,4), dtype='int')
43     for i in range(1,nel+1):
44         IENCC[i-1,0]=i
45         IENCC[i-1,1]=i-1
46         IENCC[i-1,2]=(npt/2.)-1+i
47         IENCC[i-1,3]=(npt/2.) +i
48     klc=np.zeros((4,4), dtype='float')
49     klc[0,0]=-mcpq
50     klc[1,0]=mcpq -(U*dA/2.)
51     klc[1,1]=-U*dA/2.
52     klc[1,2]=U*dA/2.
53     klc[1,3]=U*dA/2.

```

```

54 klc[2,2]=-mcpf
55 klc[3,0]=U*dA/2.
56 klc[3,1]=U*dA/2.
57 klc[3,2]=mcpf-(U*dA/2.)
58 klc[3,3]=-(U*dA/2.)
59 K=np.zeros((npt,npt),dtype='float')
60 for elem in range(nel):
61     for i in range(4):
62         igb=IENCC[elem-1,i-1]
63         for j in range(4):
64             jgb=IENCC[elem-1,j-1]
65             K[igb,jgb]= K[igb,jgb]+ klc[i-1,j-1]
66 K[0,0]=K[0,0]-mcpq
67 K[npt-1,npt-1]=K[npt-1,npt-1]-mcpf
68 temp=np.zeros((npt), dtype='float')
69 temp=np.linalg.solve(K, Q)
70 dT = temp[-1] - Toutff
71 L -= dT
72 deltaTum = float(Tfq - Toutff)
73 deltaTdois = float(temp[0] - Tff)
74 deltaTlmtd = (deltaTdois-deltaTum)/(np.log(deltaTdois/deltaTum))
75 Qtrocada=float(mcpf*(temp[npt-1]-temp[npo]))
76 Lanalitico = Qtrocada/(U*np.pi*D*deltaTlmtd)
77 print('Media logaritmica das diferencas de temperatura: {}'.format←
    (deltaTlmtd))
78 print('Temperatura de saida do fluido quente (Toutfq): {} K' .←
    format(temp[0]))
79 print('Numero de iteracoes necessarias: {}'.format(n_it))
80 print('Erro absoluto: {}'.format(dT))
81 print('--- Comprimento do Trocador CC (analitico): {} m ---' .←
    format(Lanalitico))
82 print('--- Comprimento do Trocador CC (numerico): {} m ---' .format←
    (L))
83 print('Quantidade de calor trocada e: '+str(Qtrocada/1000.)+' kW')
84 dx=float(L/nel)
85 x=np.zeros((int(npo),1),dtype='float')
86 for i in range (1,int(npo)):
87     x[i]=x[i-1]+dx
88 o = np.arange(0.0, 82, 1)
89 fig, ax = plt.subplots(1, 1, figsize←
    =(2*4.05511811023622,4.05511811023622))
90 line1 = plt.plot(x,temp[0:npo:+1], linestyle='-',marker='', color='←
    #ff796c',markersize=o[5], label='Fluido quente (agua)')
91 line2 = plt.plot(x,temp[npo:npt:+1], linestyle='-',marker='', color←
    ='#06c2ac', markersize=o[5], label='Fluido frio (quimico)')

```

```
92 plt.tick_params(axis='both', labelsiz=12)
93 plt.xlabel('Comprimento do trocador de calor [m]', fontsize=12)
94 plt.ylabel('Temperatura [K]', fontsize=12)
95 plt.legend(title='', loc=0, borderaxespad=0., fontsize=10)
96 plt.grid(b=True, which='both', color='0.9', linestyle='-')
97 plt.show()
98 fig.savefig('dimensionartcc.pdf', format='pdf', dpi=1200, ↵
    bbox_inches='tight', transparent=True)
99 end_time = datetime.now()
100 print('Duracao: {}'.format(end_time - start_time))
```

## 2.2 Resultado : *analítico vs. numérico*

O *output* abaixo foi obtido para o código acima, que contempla o resultado analítico e o compara com o numérico.

```
Comprimento L escolhido aleatoriamente para inicio das iteracoes: 163 m
Qual o numero de elementos? 1000
Numero de pontos da malha: 2002
Media logaritmica das diferencas de temperatura: 91.9734467186
Temperatura de saida do fluido quente (Toutfq): 125.085846863 K
Numero de iteracoes necessarias: 53
Erro absoluto: 8.70358007887e-09
--- Comprimento do Trocador CC (analitico): 108.498868851 m ---
--- Comprimento do Trocador CC (numerico): 108.498868175 m ---
Quantidade de calor trocada é: 300.960000044 kW
```

## 3 Dificuldades

O maior dos problemas foi a aprendizagem em *python* sem o contato prévio com a linguagem ao longo do curso de engenharia mecânica.

## Referências

- [1] Frank P. Incropera, Lavine, Dewitt, and Bergman. *Fundamentals of Heat and Mass Transfer*. Springer-Verlag, 7<sup>th</sup> edition, 2011.
- [2] M. Necati Ozisik. *Heat Transfer*. McGraw-Hill, 1<sup>st</sup> edition, 1985.
- [3] Prof. Gustavo Anjos. *Transmissão de Calor - Análise Discreta*. PPG-EM (UERJ), 2017.