Nuclear Reactor Thermal-Hydraulics

NUGN520 - Homework

By

GUILLAUME L'HER

Department of Nuclear Engineering
COLORADO SCHOOL OF MINES

Homework submitted for the Nuclear Reactor Thermal-Hydraulics class at the Colorado School of Mines.

SPRING 2017

Table of Contents

Page

CHAPTER 1

Reactor Heat Generation

Several exercises from the book written by M. M. El Wakil [1] are tackled in this homework. The problems in this section relate to the seventh and eigth chapter of the book, covering the subject of heat conduction in reactor elements.

1.1 [7-6] -

1.1.1 Problem

A very long fin is rectangular in cross-section $0.48 * 0.24$ *in. It generates* $2 \times 10^6\ Btu.h^{-1}.ft^{-3}$. *The fin base is at* $1000°F$. *It is cooled by a gas at* $600°F$ *with a uniform heat transfer coefficient* $100\ Btu.h^{-1}.ft^{-2}.°F^{-1}$. *Using a network with* $\Delta x = 0.12$ *in., write the necessary set of finite difference equations for the nodal points and solve by any one of the techniques at your command.* k *for the fin material* $= 10\ Btu.h^{-1}.ft^{-1}.°F^{-1}$

1.1.2 Solution

First, one can note that the problem is missing a graph. The considered geometry is given in Figure 1.1.

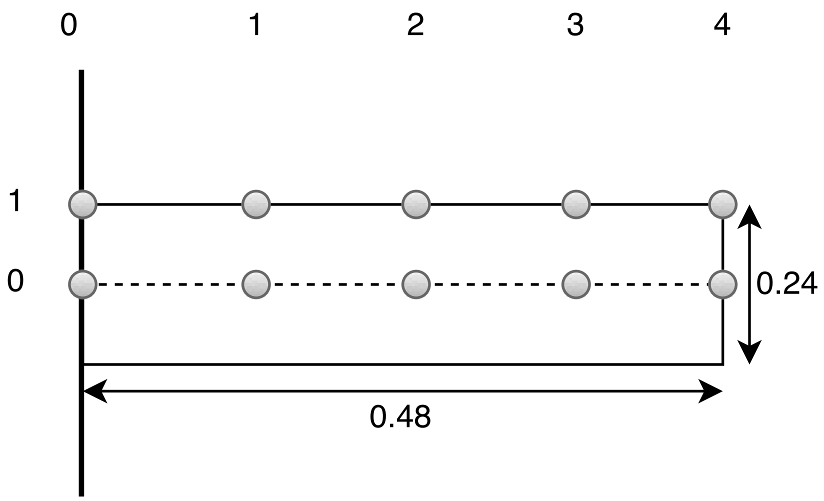

Figure 1.1: Representation of the problem geometry

We can identify four types of nodes: inner, y-bound (next to the upper boundary), x-bound (next to the right boundary) and corner.

The inner node temperature T_n is given by Equation 1.1.

$$T_n = \frac{T_{n-\Delta x} + T_{n+\Delta x} + T_{n-\Delta y} + T_{n+\Delta y}}{4} + \frac{\Delta T_g}{2} \tag{1.1}$$

In our case, we can see that $T_{n-\Delta y} = T_{n+\Delta y}$ by symmetry for the inner nodes. Consequently, we obtain Equation 1.2.

$$T_n = \frac{T_{n-\Delta x} + T_{n+\Delta x} + 2T_{n+\Delta y}}{4} + \frac{\Delta T_g}{2} \tag{1.2}$$

For the x-bound nodes, we can write Equation 1.3.

$$T_n = \frac{T_{n-\Delta x} + T_{n+\Delta y} + Bi_{\Delta x} T_f}{2 + 2Bi_{\Delta x}} + \frac{\Delta T_g}{2 + Bi_{\Delta x}} \tag{1.3}$$

For the y-bound nodes, we can write Equation 1.4

$$T_n = \frac{T_{n-\Delta x} + T_{n+\Delta x} + 2T_{n+\Delta y} + 2Bi_{\Delta x} T_f}{4 + 2Bi_{\Delta x}} + \frac{\Delta T_g}{2 + Bi_{\Delta x}} \tag{1.4}$$

And finally, for the corner node, we can write (as seen in problem 7.1 previously) Equation 1.5.

$$T_n = \frac{T_{n-\Delta x} + T_{n+\Delta y} + 2Bi_{\Delta x} T_f}{2 + 2Bi_{\Delta x}} + \frac{\Delta T_g}{2 + 2Bi_{\Delta x}} \tag{1.5}$$

A few exceptions can be noted. Indeed, the two leftmost points, on the base of the fin, are given to be at $1000°F$.

We can consequently write the matrix A.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.25 & 0 & 1 & -0.5 & -0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-1}{2Bi_{\Delta x}+4} & \frac{-1}{Bi_{\Delta x}+2} & 1 & 0 & \frac{-1}{2Bi_{\Delta x}+4} & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.25 & 0 & 1 & -0.5 & -0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{-1}{2Bi_{\Delta x}+4} & \frac{-1}{Bi_{\Delta x}+2} & 1 & 0 & \frac{-1}{2Bi_{\Delta x}+4} & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.25 & 0 & 1 & -0.5 & -0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{-1}{2Bi_{\Delta x}+4} & \frac{-1}{Bi_{\Delta x}+2} & 1 & 0 & \frac{-1}{2Bi_{\Delta x}+4} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{Bi_{\Delta x}+2} & 0 & 1 & \frac{-1}{Bi_{\Delta x}+2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{2Bi_{\Delta x}+2} & \frac{-1}{2Bi_{\Delta x}+2} & 1 \end{bmatrix}$$

$\frac{\Delta T_g}{2}$ can be calculated using $\Delta T_g = \frac{(\Delta x)^2 q'''}{2k}$. T_f, the temperature of the gas, is known. We can thus obtain b.

$$b = \begin{bmatrix} 1000 \\ 1000 \\ 7.2 \\ 229.5 \\ 7.2 \\ 229.5 \\ 7.2 \\ 229.5 \\ 229.5 \\ 330.5 \end{bmatrix}$$

The equation $A.x = b$ can thus be solved, using Python. The python script is given in 1.1.2.1. We obtain the following results (Equation 1.6):

$$\begin{aligned} T(0,0) &= 1000.0 \\ T(0,1) &= 1000.0 \\ T(1,0) &= 804.0 \\ T(1,1) &= 740.9 \\ T(2,0) &= 705.3 \\ T(2,1) &= 665.0 \\ T(3,0) &= 658.5 \\ T(3,1) &= 635.7 \\ T(4,0) &= 628.3 \\ T(4,1) &= 617.8 \end{aligned} \tag{1.6}$$

1.1.2.1 Python script

```
import numpy as np

m = 10    # size of system is m x m

dx = 0.12
deltax, k, qtriple, h = (dx/10)**2, 10., 2.e6, 100.    # BG units
tb = 1000.
ts = 600.
dtg = (0.5*deltax)*qtriple/k
bi = h * dx / k

amat = np.zeros((m,m))
b = np.zeros(m)

amat[0,0] = 1.
amat[1,1] = 1.
b[0] = tb
b[1] = tb

for i in range(2, m, 2):
    try:
        amat[i,i-2] = -0.25
        amat[i,i] = 1.
        amat[i,i+1] = -0.5
        amat[i,i+2] = -0.25
        b[i] = dtg/2.
    except IndexError:
        break

for i in range(3, m, 2):
    try:
        amat[i,i-2] = -1/(2*bi+4)
        amat[i,i-1] = -1/(bi+2)
        amat[i,i] = 1.
        amat[i,i+2] = -1/(2*bi+4)
        b[i] = (bi*ts + dtg)/(bi+2)
    except IndexError:
        break

amat[m-1,m-1] = 1.
amat[m-1,m-2] = -1/(2*bi+2)
amat[m-1,m-3] = -1/(2*bi+2)
b[m-1] = (2*bi*ts + dtg)/(2*bi+2)


amat[m-2,m-2] = 1.
amat[m-2,m-1] = -1/(bi+2)
amat[m-2,m-4] = -1/(bi+2)
b[m-2] = (bi*ts + dtg)/(bi+2)
print(b)
ysol = np.linalg.solve(amat,b)
```

```
print('Temperature solution:')

temp = []
for i in range(5):
    for j in range(2):
        temp.append("T(%s,%s)" % (i, j))

for i,t in enumerate(temp):
    print("%s = %.1f" % (t, ysol[i]))

# check of solution:
if not np.allclose(np.dot(amat, ysol), b):
    print("Solution does not match!")
```

Bibliography

[1] M. M. El-Wakil, *Nuclear Heat Transport*, American Nuclear Society, 1993.