Heat exchanger - Python

Adam Staniucha

February 1, 2018

Abstract

The main objective of this project was creation of a mini application which calculated parameters of heat exchanger

1 Introduction

Many applications require more and more computing power. We can increase power by using cloud computing

This project did not require huge power, but I used cloud computing to learn how it works.
I also learned how to use:
- AWS Amazon
- Linux
- Jupiter
- EC2
- WinSCP
- PuTTY
- Latex

and the most important think: I learned basics of programming in Python.

2 Project

2.1 Project description

First conception of project:
- calculation of parameters of heat exchanger e.g. mass flow, temperature
- making a chart - T(x),
- determination of the temperature field

Unfortunately, this conception was too hard and I have problems with code application which can do these things.

2.2 Short application

This version of application have a few input data: mass of water and temperatures.
It calculate a mass of air, heat and LMGT.
It is a very easy application wtich use a math library. I want create a better application which calculated a temperatur in every point usuing 'numpy' but the code have same mistakes.

The code of short application:

```
import math

T1=100.0
T2=75.0
t1=5.0
t2=50.0
M1=20
cw=4.1899
cp=1.005


m2=(T1-T2)*M1*cw/((t2-t1)*cp)
Q1=M1*(T1-T2)*cw
Q2=m2*(t2-t1)*cp
LMTD=((T1-t2)-(T2-t1))/(math.log((T1-t2)/(T2-t1)))

print"The LMTD for counter flow configuration is ",round(LMTD,1),"C"
print"The mass of air is ",round(m2,1), "kg"
print"The heat is ",round(Q1,1)
```

The results is easy:

```
The LMTD for counter flow configuration is 59.4 C
The mass of air is 46.3 kg
The heat is 2094.9
```

2.3 Second application

I created a second another application to train usuing Python.
This application is using pump.
NumPy is the fundamental package for scientific computing with Python.
This small application is creates a chart of fanning friction factor from reynolds.
Code of this application:

```
import numpy as np
from scipy.optimize import fsolve
import matplotlib.pyplot as plt

def fanningfrictionfactor(Re):
def fz(f): return 1.0/np.sqrt(f) - (4.0*np.log10(Re*np.sqrt(f))-0.4)
sol, = fsolve(fz, 0.01)
return sol


Re = np.linspace(2200, 10000)
f = fanningfrictionfactor(Re)

plt.plot(Re, f)
plt.xlabel('Re')
plt.ylabel('fanning friction factor')
plt.show();
```

Results of this code is chart:

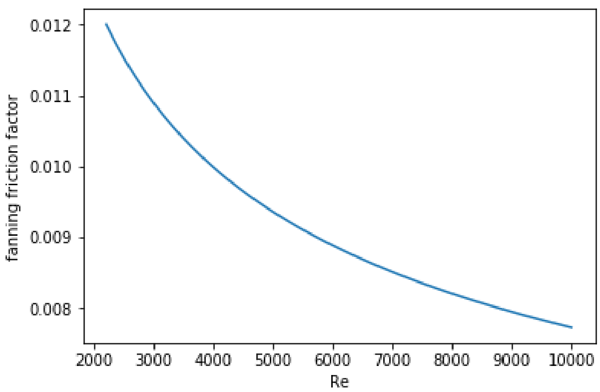

Figure 1: friction factor(Re)

3 Results

Results of this projects are two small application calculated a heat exchanger and fanning friction factor.

References