Water Layer Absent

April 23, 2021

1 Case 1: No Water Layer

- Author: **Team G15**
- Attempt: 3

1.1 Analysis

1.1.1 To find

1. Temperature of Roof Surface (T_s)
2. Total heat flux entering the house through the roof, (q_t) when no water layer is present

1.1.2 Nomenclature

- T_s = roof surface temperature (outside)
- T_a = ambient air temperature (outside)
- T_r = room temperature (inside)
- Nu_a = Nusselt number of air
- Ra_a = Rayleigh number of air
- Re_a = Reynolds number of air
- Pr_a = Prandtl number of air
- α_a = thermal diffusivity of air
- k_a = thermal conductivity of air
- h_r = free convection coefficient of room air
- ν_a = dynamic Viscosity of air
- Roof layers:
  - 1: Concrete
  - 2: Brick
  - 3: Lime
- k_i = thermal conductivity of i^{th} roof layer
- L_i = length of i^{th} roof layer
- q_r = radiative heat transfer (per unit area)
- q_c = convective heat transfer (per unit area)
- q_t = net heat transfer into the room (per unit area)
- β = coefficient of thermal expansion
- S = Intensity of Solar Radiation (i.e. solar constant)

1.1.3 Assumptions

- Steady state with room maintained at fixed ambient temperature

1.1.4 Equations

Energy balance,

$$q_t = q_c + q_r$$

Radiation heat transfer,

$$q_r = \tau_s \cdot S - h_r \cdot (T_a - T_s)$$

$$h_r = \epsilon_s \cdot \sigma \cdot \frac{(\overline{T}_s)^4 - (\overline{T}_a - 12)^4}{\overline{T}_a - \overline{T}_s}$$

Convection heat transfer,

$$q_c = h_c \cdot (T_a - T_w)$$

$$h_c = \frac{k_a}{L_s} \cdot Nu_a$$

$$Nu_a = 0.15 \cdot Ra_a^{1/3} + 0.664 \cdot Re_a^{1/2} \cdot Pr_a^{1/3}$$

$$Re_a = \frac{v_a \cdot L_s}{\nu_a}$$

$$Ra_L = \frac{g \cdot \beta \cdot (T_s - T_a) \cdot L_s^3}{\nu_a \cdot \alpha_a}$$

Total heat transfer,

$$q_t = \frac{T_w - T_r}{R_{net}}$$

$$R_{net} = \frac{1}{h_r} + \sum_{i=1}^{3} \frac{L_i}{k_i}$$

1.1.5 Properties

Outside Air

- Mild breeze $v_a = 2.78\ m/s$
- $T_a \in [305, 320]K$
- $T_f = 320K$
- $\beta = \frac{1}{T_f} = 0.0031\ K^{-1}$
- Table A.4, air (T_f):
  - $\nu = 18 \cdot 10^{-6}\ m^2/s$
  - $\alpha = 25 \cdot 10^{-6}\ m^2/s$

- $Pr = 0.702$
- $k = 27.7 \cdot 10^{-3}\ W/m \cdot K$

- $S = 1366\ W/m^2$

Roof

- $L_s = 5\ m$ (approx thickness of water layer)
- $\epsilon_s = 0.9$ (concrete surface)
- $\tau_s = 0.9$
- $t = 0.2\ m$ thick with,
  - Cement $= 5\ cm$
  - Brick $= 10\ cm$
  - Lime $= 5\ cm$
- K_i, Conductivity of each layer,
  - Cement $= 0.72\ W/m \cdot K$
  - Brick $= 0.71\ W/m \cdot K$
  - Lime $= 0.73\ W/m \cdot K$

Inside air

- $T_r = 300K$ (Room Temperature)
- $h_r = 8.4\ W/m^2 \cdot K$

1.1.6 Tools used

- **Python**
- **SymPy** for creating symbolic equations and solving them
- **NumPy**
- **Matplotlib** for plotting results

1.2 Solving (Python Code)

1.2.1 Initialize Values

[1]:
```
import sympy as sp
import numpy as np
```

Outside Air

- Table A.4 used (from reference #2)

[2]:
```
v_a = 2.78 # Velocity (m / s)

# Temperatures
T_f = 320.0 # (K)
beta = 1/T_f # (K)
T_a = np.array([305.0, 310.0, 315.0, 320.0]) # (K)
T_a_avg = 273 + 37 # (K)

# Universal Constants
```

```
sigma = 5.67e-8 # Stefan Boltzmann constant (W / m^2 * K^4)
g = 9.8 # (m^2 / s)
S = 1366 # Solar constant

# Table A.6, air @ T = T_f
nu_a = 18e-6 # dynamic visosity (m^2 / s)
alpha_a = 25e-6 # (m^2 / s)
k_a = 27.7e-3 # thermal conductivity (W / m * K)
Pr_a = 0.702
```

Roof Layers

[3]:
```
# Temperatures
T_s = sp.symbols('T_s') # Roof surface temp (K)
T_s_avg = 273.0 + 35.0 # (K)

# Surface
L_s = 5 # Dimensions (m)
tau_s = 0.9 # Roof's solar absorbtivity
epsilon_s = 0.9 # Emissivity of roof surface (concrete)

# Layer 1: Concrete
k_1 = 0.72 # (W / m * K)
L_1 = 0.05 # (m)

# Layer 2: Brick
k_2 = 0.71 # (W / m * K)
L_2 = 0.10 # (m)

# Layer 3: Lime
k_3 = 0.73 # (W / m * K)
L_3 = 0.05 # (m)
```

Inside Air

[4]:
```
h_r = 8.4 # (W / m^2 * K)
T_r = 300 # (K)
```

1.2.2 Equations

Radiation Heat

[5]:
```
h_r = epsilon_s * sigma * (T_s_avg**4 - (T_a_avg - 12)**4)/(T_a_avg - T_s_avg)␣
↪# (W / m^2 * K)
q_r = tau_s * S - h_r * (T_a - T_s) # (W / m^2)

# Example at T_a = 310K and T_s = 308K
q_r_test = q_r[1].replace(T_s, 308)
print('Approximate value of q_r = %.2f W/m^2' % (q_r_test))
```

```
Approximate value of q_r = 1172.60 W/m^2
```

Convection Heat

- From below analysis, we can neglect free convection in comparison to forced convection

Free Convection

[6]:
```
Ra_a = (g * beta * (T_a - T_s) * L_s**3) / (nu_a * alpha_a)
Nu_a_fr = 0.15 * Ra_a**(1/3)
h_c_fr = k_a / L_s * Nu_a_fr

# Example at T_a = 310K and T_s = 308K
h_c_fr_test = h_c_fr[1].replace(T_s, 308)
print('Approximate value of free convection coefficient = %.2f W/K*m^2' %
 ↪(h_c_fr_test))
```

```
Approximate value of free convection coefficient = 2.14 W/K*m^2
```

Forced Convection

[7]:
```
Re_a = v_a * L_s / nu_a
Nu_a_fo = 0.664 * Re_a**1/2 * Pr_a**1/3
h_c_fo = k_a / L_s * Nu_a_fo

# Example at T_a = 310K and T_s = 308K
print('Approximate value of forced convection coefficient = %.2f W/K*m^2' %
 ↪(h_c_fo))
```

```
Approximate value of forced convection coefficient = 332.36 W/K*m^2
```

Total Convection

[8]:
```
h_c = h_c_fo # Neglicting free convection
q_c = h_c * (T_a - T_s) # (W / m^2)

# Example at T_a = 310K and T_s = 308K
q_c_test = q_c[1].replace(T_s, 308)
print('Approximate value of q_c = %.2f W/m^2' % (q_c_test))
```

```
Approximate value of q_c = 664.72 W/m^2
```

Total Heat:

[9]:
```
R = 1/h_r + L_1/k_1 + L_2/k_2 + L_3/k_3 # (m^2 * K / W)

q_t = (T_s - T_r) / R # (W / m^2)

# Example at T_a = 310K and T_s = 308K
q_t_test = q_t.replace(T_s, 308)
print('Approximate value of q_t = %.2f W/m^2' % (q_t_test))
```

```
Approximate value of q_t = 25.48 W/m^2
```

1.2.3 Solving

$$q_c + q_r = q_t$$
$$\therefore\ q_c + q_r - q_t = 0$$

Calculate T_s

[10]:
```
eq = q_c + q_r - q_t

n = len(eq)
T_s_calc = np.empty(n, dtype=object)

for i in range(n):
    T_s_calc[i] = round(sp.solve(eq[i], T_s)[0], 2)

for i in range(n):
    print('T_s = %.1f K for T_a = %.1f K' % (T_s_calc[i], T_a[i]))
```

```
T_s = 309.0 K for T_a = 305.0 K
T_s = 313.9 K for T_a = 310.0 K
T_s = 318.9 K for T_a = 315.0 K
T_s = 323.8 K for T_a = 320.0 K
```

Calculate q_t

[11]:
```
q_t_calc = np.empty(n, dtype=object)

for i in range(n):
    q_t_calc[i] = q_t.replace(T_s, T_s_calc[i])

for i in range(n):
    print('Heat entering = %.1f W/m^2 for T_a = %.1f K' % (q_t_calc[i], T_a[i]))
```

```
Heat entering = 28.5 W/m^2 for T_a = 305.0 K
Heat entering = 44.3 W/m^2 for T_a = 310.0 K
Heat entering = 60.0 W/m^2 for T_a = 315.0 K
Heat entering = 75.8 W/m^2 for T_a = 320.0 K
```

1.2.4 Plot

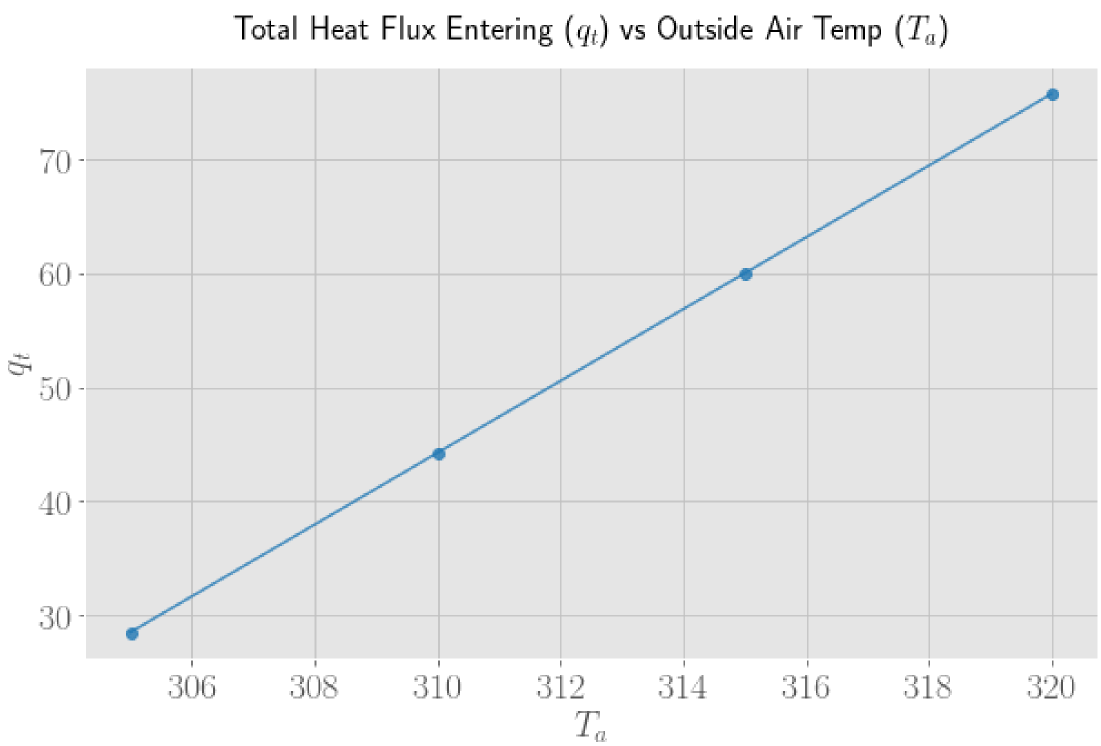
- Total Heat Flux Entering (q_t) vs Outside Air Temp (T_a)

[15]:
```
%matplotlib inline
import matplotlib.pyplot as plt

# Initialize matplotlib
plt.rc('text', usetex=True) # Unnecessary
plt.style.use('ggplot')
plt.rcParams['grid.color'] = '#C0C0C0'
```

```
fig = plt.figure(figsize=(10, 6))
plt.plot(T_a, q_t_calc, color='#1F77B4cc', marker='o')
plt.xticks(fontsize=20)
plt.yticks(fontsize=20)
plt.xlabel('$T_a$', fontsize=20)
plt.ylabel('$q_t$', fontsize=20)
plt.title('Total Heat Flux Entering ($q_t$) vs Outside Air Temp ($T_a$)',␣
 ↪fontsize=18, pad=15)
```

[15]:
```
Text(0.5, 1.0, 'Total Heat Flux Entering ($q_t$) vs Outside Air Temp ($T_a$)')
```



[ ]: