7.6 Flow Across Banks of Tubes

Fundamentals of Heat and Mass Transfer, 8th Edition

Table for C and M values

**TABLE 7.5** Constants of Equation 7.58 for the tube bank in cross flow [17]

| Configuration | $Re_{D,\max}$ | $C_1$ | $m$ |
|---|---|---|---|
| Aligned | $10$–$10^2$ | 0.80 | 0.40 |
| Staggered | $10$–$10^2$ | 0.90 | 0.40 |
| Aligned | $10^2$–$10^3$ | Approximate as a single | |
| Staggered | $10^2$–$10^3$ | (isolated) cylinder | |
| Aligned ($S_T/S_L > 0.7$)[a] | $10^3$–$2 \times 10^5$ | 0.27 | 0.63 |
| Staggered ($S_T/S_L < 2$) | $10^3$–$2 \times 10^5$ | $0.35(S_T/S_L)^{1/5}$ | 0.60 |
| Staggered ($S_T/S_L > 2$) | $10^3$–$2 \times 10^5$ | 0.40 | 0.60 |
| Aligned | $2 \times 10^5$–$2 \times 10^6$ | 0.021 | 0.84 |
| Staggered | $2 \times 10^5$–$2 \times 10^6$ | 0.022 | 0.84 |

[a]For $S_T/S_L < 0.7$, heat transfer is inefficient and aligned tubes should not be used.

Table for C and M coeffiecents.

$$\overline{Nu}_D = C_1 \, Re_{D,\max}^m \, Pr^{0.36} \left( \frac{Pr}{Pr_s} \right)^{1/4}$$

$$\left[ \begin{array}{l} N_L \geq 20 \\ 0.7 \lesssim Pr \lesssim 500 \\ 10 \lesssim Re_{D,\max} \lesssim 2 \times 10^6 \end{array} \right]$$

At least 20 tubes(**5 not 20 need correction factor**)

**TABLE 7.6** Correction factor $C_2$ of Equation 7.59 for $N_L < 20$ ($Re_{D,\max} \gtrsim 10^3$) [17]

| $N_L$ | 1 | 2 | 3 | 4 | 5 | 7 | 10 | 13 | 16 |
|---|---|---|---|---|---|---|---|---|---|
| Aligned | 0.70 | 0.80 | 0.86 | 0.90 | 0.92 | 0.95 | 0.97 | 0.98 | 0.99 |

Prandlt number range, Reynolds number range

The Reynolds number $Re_{D,\max}$ for the foregoing correlation is based on the *maximum* *velocity* occurring within the tube bank, $Re_{D,\max} = \rho V_{\max} D/\mu$. For the aligned arrange-t, $V_{\max}$ occurs at the transverse plane $A_1$ of Figure 7.12*a*, and from the mass conserva- requirement for a constant density fluid

$$V_{\max} = \frac{S_T}{S_T - D} V \qquad (7.60)$$

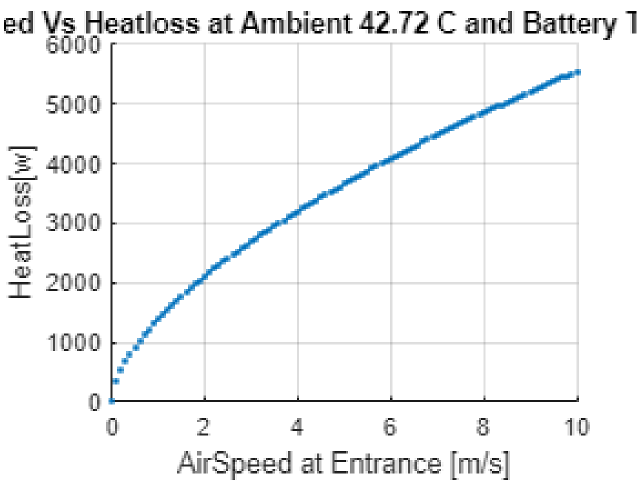
Input vector creation and graphing

```
AirSpeed = 0:0.1:10;
tempLow = 40+ 273.15;
battTemp = 50+ 273.15;
tempHigh = 42.57 + 273.15;
%y = Heatloss(AirSpeed,tempLow,battTemp)
```

```
hold on
ylabel('HeatLoss[w]');
plot(AirSpeed,y,'.')
%yyaxis right
%plot(z,AirSpeed,'.')
xlabel('AirSpeed at Entrance [m/s]');
%ylabel('Ploss');
var2 = battTemp-273.15;
var1 = tempHigh-273;
title_string = sprintf("AirSpeed Vs Heatloss at Ambient %.2f C and Battery Temp
%.2f",var1,var2)
```

```
title_string =
"AirSpeed Vs Heatloss at Ambient 42.72 C and Battery Temp 50.00"
```

```
title(title_string)
grid on;
hold off
```




```
%function HeatLoss = Heatloss(AirSpeed,tempLow,battTemp)
```

Basic

```
%Reynolds Number Calcs
    BattColumns = 5;
    BatteryRows =23;
```

```
    tempHigh = 40 + 273.15;
    DiameterCells = 18.3/1000; % CHANGE THIS BECAUSE NO GAP




    dynViscosity = py.CoolProp.CoolProp.PropsSI('V', 'T', tempLow, 'P|not_imposed',
101325, 'air')
```

```
dynViscosity = 1.9165e-05
```

```
    densityAir = py.CoolProp.CoolProp.PropsSI('D', 'T', tempLow, 'P|not_imposed',
101325, 'air')
```

```
densityAir = 1.1274
```

```
    thermCond = py.CoolProp.CoolProp.PropsSI('L', 'T', tempLow, 'P|not_imposed',
101325, 'air')
```

```
thermCond = 0.0274
```

```
    ConstPNum = py.CoolProp.CoolProp.PropsSI('C', 'T', tempLow, 'P|not_imposed',
101325, 'air')
```

```
ConstPNum = 1.0069e+03
```

```
    thermMassAir = py.CoolProp.CoolProp.PropsSI('C', 'T', tempLow, 'P|not_imposed',
101325, 'air')
```

```
thermMassAir = 1.0069e+03
```

```
    PrandtlAirLow = py.CoolProp.CoolProp.PropsSI('Prandtl', 'T', tempLow, 'P|
not_imposed', 101325, 'air')
```

```
PrandtlAirLow = 0.7055
```

```
    PrandtlAirHigh = py.CoolProp.CoolProp.PropsSI('Prandtl', 'T', tempHigh, 'P|
not_imposed', 101325, 'air')
```

```
PrandtlAirHigh = 0.7055
```

```
    KinViscosity = dynViscosity/densityAir;


% Vert vs Horiz Spacing
    St = 24.3/1000;% spacing between columns
    Sl = 34.3/1000;%center spacing in direction of initial airflow

    Vmax =  St.*AirSpeed./(St-DiameterCells);

    ReynoldsNumber =  Vmax * DiameterCells / KinViscosity % WILL have to update C1
and M Manually before IF statements.
```

ReynoldsNumber = 1×101

$10^4$ ×

```
        0    0.0436    0.0872    0.1308    0.1744    0.2180    0.2616    0.3052 ⋯
```

```
C1 = 0.35*(St/Sl)^(1/5);
mVal = 0.6; % Table
C2 = 0.95; %Table (correction factor for 5)
```

$$\overline{Nu}_D = C_2 C_1 Re_{D,\max}^m Pr^{0.36} \left( \frac{Pr}{Pr_s} \right)^{1/4}$$

$$\bar{h} = \overline{Nu}_D \frac{k}{D}:$$

```
    NusseltD = C2*C1*ReynoldsNumber.^mVal.*(PrandtlAirLow.^0.36).*(PrandtlAirLow./
PrandtlAirHigh).^(0.25);
    Hline = NusseltD*thermCond/DiameterCells;
```

Adjusted Delta for warming

$$\Delta T_{\text{lm}} = \frac{(T_s - T_i) - (T_s - T_o)}{\ln\left( \frac{T_s - T_i}{T_s - T_o} \right)}$$

$$\frac{T_s - T_o}{T_s - T_i} = \exp\left( -\frac{\pi D N \bar{h}}{\rho V N_T S_T c_p} \right)$$

$$T_s - T_o = (T_s - T_i) \exp\left( -\frac{\pi D N \bar{h}}{\rho V N_T S_T c_p} \right)$$

```
insideVarTop = (-pi*DiameterCells*(BattColumns*BatteryRows)*Hline);

insideVarBot = (densityAir.*AirSpeed.*(BatteryRows)*St*ConstPNum);
adjDelTso = (battTemp - tempLow) * exp(insideVarTop/insideVarBot);
To = -((battTemp - tempLow) * exp(insideVarTop/insideVarBot))+battTemp -273.15
```

```
To = 42.5796
```

$$\Delta T_{\mathrm{lm}} = \frac{(T_s - T_i) - (T_s - T_o)}{\ln\left(\frac{T_s - T_i}{T_s - T_o}\right)}$$

```
    deltTlm = ((battTemp-tempLow)-(adjDelTso))/(log((battTemp-tempLow)/
(adjDelTso)));
```

$q' = N(\bar{h}\pi D \Delta T_{\mathrm{lm}})$

```
    qPrime = (BatteryRows*BattColumns)*(Hline*pi*DiameterCells*deltTlm);%in KW per m
    qPrimeLadjust = qPrime*0.065*6%Temp dissapated
```

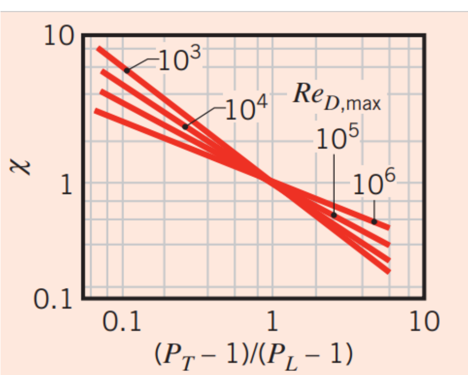
```
qPrimeLadjust = 1×101
10^3 ×
        0    0.3498    0.5301    0.6761    0.8035    0.9186    1.0248    1.1241 ···
```

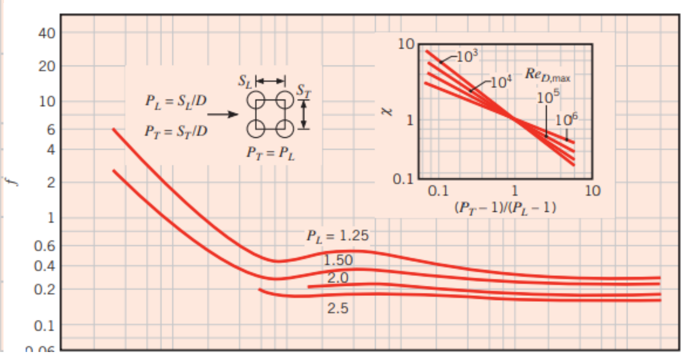
```
    areaPath = St*BattColumns*0.065;
```

pressure loss

$\Delta p = N_L \chi \left(\frac{\rho V_{\max}^2}{2}\right) f$

```
    Pt = St/DiameterCells;
    Pl = Sl/DiameterCells;
    PTPL = (Pt-1)/(Pl-1);
```




```
    fancyX = 1.4;
    fricFactor = 0.22;

    deltPLoss = 5.* fancyX .* (densityAir./2.*Vmax.^2) .* fricFactor;

    HeatLoss = qPrimeLadjust;
```

```
%end
```