BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

ME F266: STUDY-ORIENTED PROJECT

Modeling and analysis of thermal management system in Fuel Cell vehicles

13th December 2021

Modeling and analysis of thermal management system in Fuel Cell vehicles

Hardik Gupta
f20180632@pilani.bits-pilani.ac.in

December 13, 2021

Contents

Abstract

A thermal management system, which involves coordinating thermal management components such as the coolant circulation pump and radiator fan in a water-cooled proton exchange membrane fuel cell (PEMFC) system, plays an important role in keeping the output performance of the PEMFC in a safe and steady condition. We have modelled here steady and transient condition of the PEM cell. Code is attached at the appendix.

1 Introduction

Due to shortage of fossil fuels and their disadvantage of pollution. Many newer technologies are evolving to provide better solution to this problem. One such solution appears to be the use of a fuel cells. A proton exchange membrane fuel cell (PEMFC) is an electrochemical power generator that offers with high power density, minimal working temperature, rapid reaction, and minimal emissions, among other benefits. It is commonly recognised as a future-proof source of power for transport. When power is created by a PEMFC, equal heat is produced and should be discharged from the PEMFC system; else, the fuel cell system's heat would rise to dangerous levels.

The coolant temperature is among the most essential control factors for a liquid PEMFC unit, as it affects gas transfer, fluid balance, oxidation and reduction activities, and fuel cell efficiency. Increased operating temperatures increase electro catalytic activity and result in greater efficiency, although maintaining water balance in PEMFC is more challenging. As a result, proper temperature control is critical to guarantee that the PEMFC system performs well and lasts a long time. 1 shows the schematic of the fuel cell.

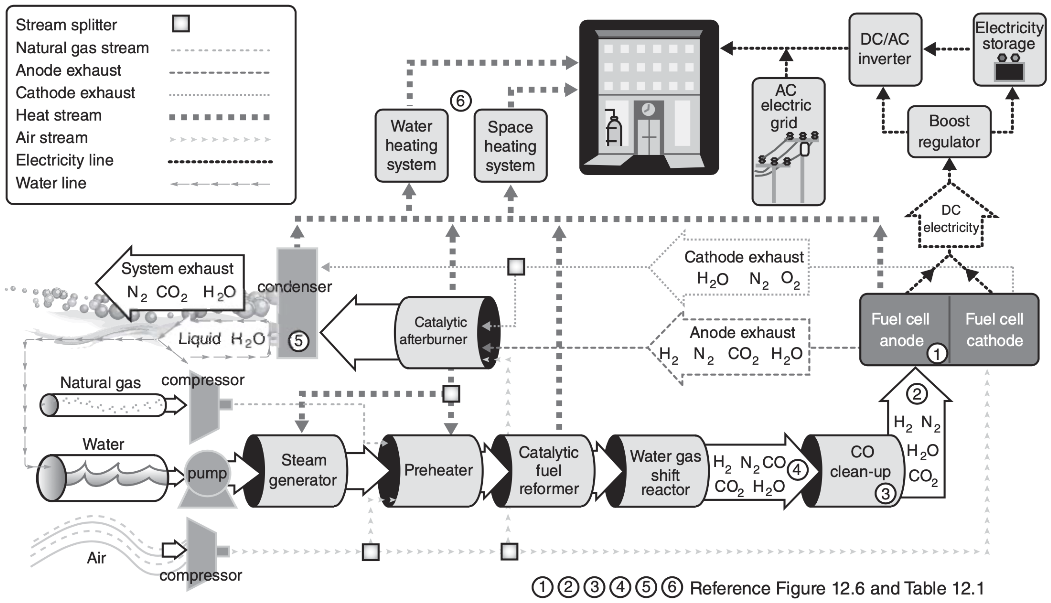

Figure 12.1. Process diagram of CHP fuel cell system. Repeated from Chapter 10 for reference.

Figure 1: Schematic of a fuel cell

2 Theory

The Schematic of The relationship between the losses and gains of heat as well as the temperature of the PEMFC per unit time is determined by the specific heat equation i.e.

$$Q = C * M * \delta T$$

The equation becomes:

$$C_{st} Mst \frac{dT_{st}}{dt} = Q_{gen} - Qdis = (Q_{tot} - P_{st}) - (Q_{gas} + Q_{cl} + Q_{abm})$$

Some of the assumptions are followed for calculations to become easy:

1. PEMFC's stack temperature = Touter of Coolant
2. Reactants are considered to be Ideal
3. Full humidification of the reactants
4. Membranes are fully saturated
5. PEMFC stack pressure difference is neglected

Considering Hydrogen and oxygen reaction:

$$Q_{total} = P_{stack} + Q_{generation}$$

$$\delta = 285.5 kJ/mol$$

Consumption and Production per unit of time is defined as:

$$N = \frac{nI_{st}}{xF}$$

whereas, x depends on specific molecule

The output of the fuel cell can be calculated as:

$$P_{st} = nV_{cell} I_{st}$$

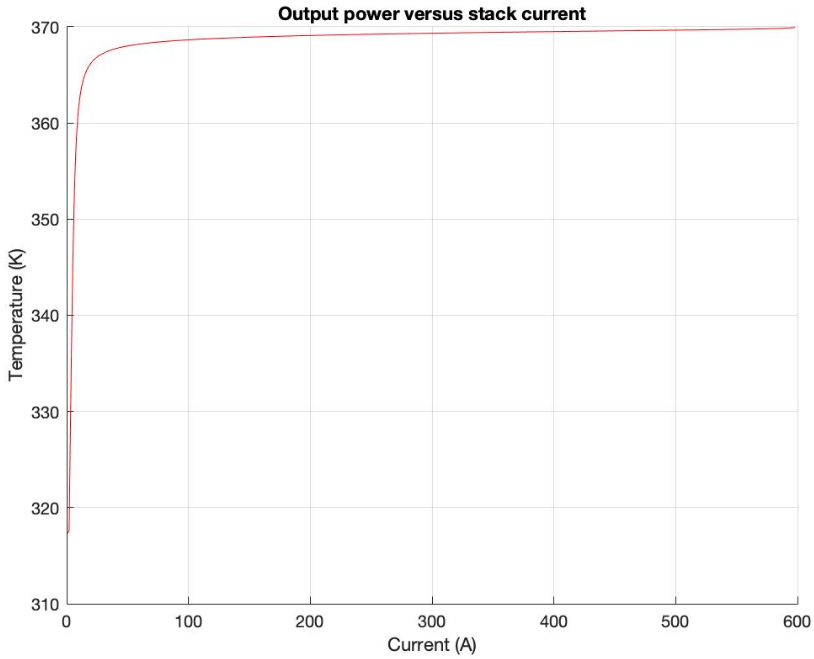

Figure 2: Curve between Current and Temperature in steady state

3 Conclusion

3.1 Steady State

In steady state conditions, the energy that goes in, comes out. As the result, the temperature would not change with time but increases with the current as shown in 2. So, in the modelling process, we are trying to find out the temperature for a particular current where error is zero, or practically very less (here, in our case 0.001). Such temperature values are collected and plotted with the current to provide such a graph.

3.2 Transient State

In transient case, there is a build of energy, where the increase of temperature is recorded with the increase in time. Such is recorded in 4. Although, this is also seen that there is a stright slope line observed in the end, that is, because dtst/dt's slope slips to zero. The graph is shown here 3.

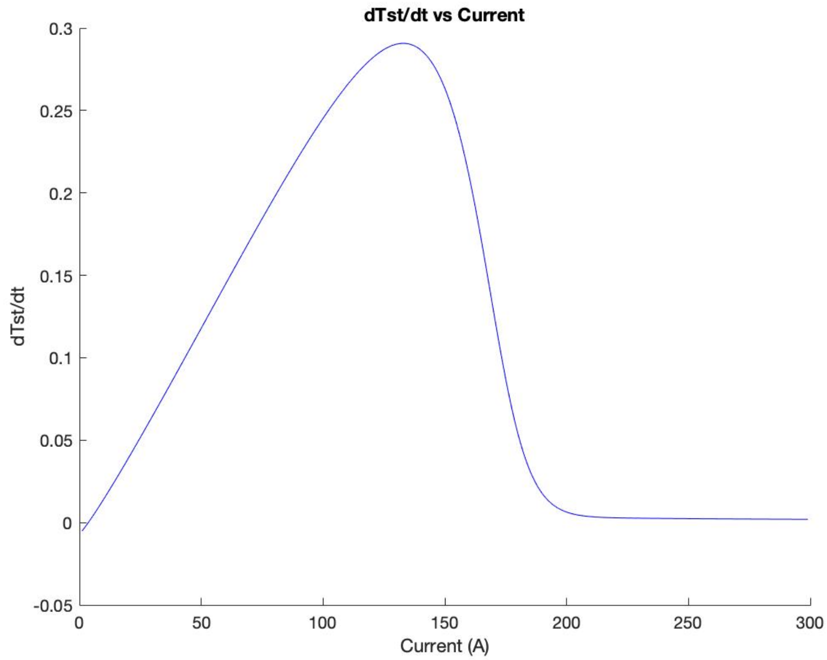

Figure 3: dTst/dt vs Current(time)

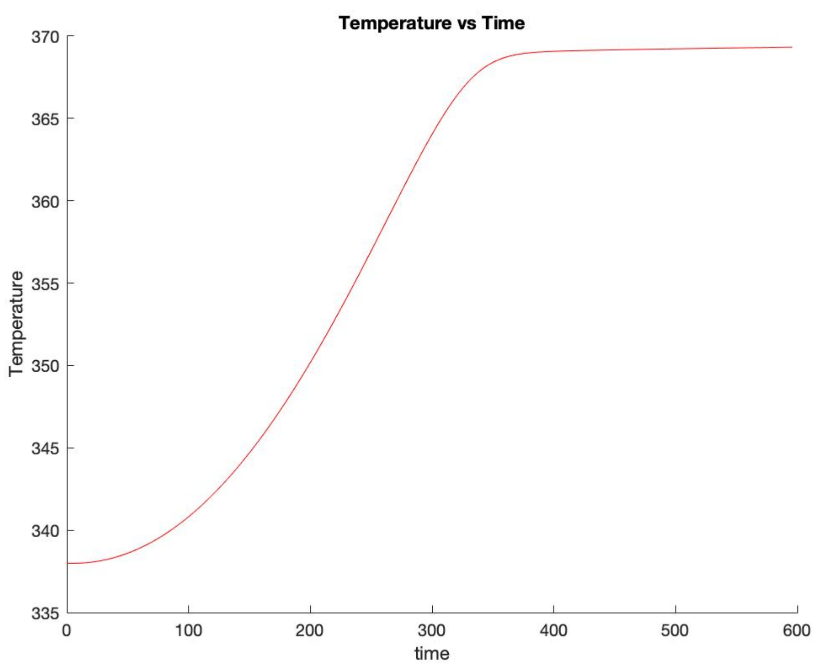

Figure 4: Curve between Temperature and time in transient state

4 References

1. F. Gonzatti et al, *Mathematical and experimental basis to model energy storage systems composed of electrolyzer, metal hydrides and fuel cells*
2. X. Zhao et al, *Thermal management system modeling of a water- cooled proton exchange membrane fuel cell*
3. C. Spigell, *Modelling and Simulation with MATLAB*
4. R. O' Hayre et al, *Fuel Cell Fundamentals*

5 Appendix

5.1 Code for Steady-State conditions

Standard Commands

```
clear all
close all
clc
```

Input parameters and constants

```
R         = 8.3145;        % Univeral gas constant J/(mol K)
F         = 96485;         % Farady's contant (A s/mol)
z         = 2;             % number of moving electrons (z = 2)
N_fc      = 370;           % number of fuel cells in the stack
T_fc      = 338;           % Operating temperature of FC (K)
P_fuel    = 1.45;          % absolute supply pressure of fuel (atm)
P_air     = 1.02;          % absolute supply pressure of air (atm)
Lemda_fc  = 5;             % stoichiometric ratio of FC
H_h2      = 285.8;         % change in enthalpy of FC reaction (kJ/mol)
C_H2      = 28.836;        % specific heat of the H2, J/mol/K
C_H2O     = 36.5;          % specific heat of the water vaoour, ...
                           % Cp = 36.5 J/(mol K) at 100C, Cv= 27.5 J/(mol K) at 100C
To        = 30+273.15;     % temperature on standard pressure,K
C_air     = 1;             % specific heat of the air, J/(mol K)
C_N2      = 1.04;          % specific heat of the N2, J/(mol K)
C_H2O_l   = 75.2;          % specific heat of the H2O, J/(mol K)
C_O2      = 29.10;         % specific heat of the O2, J/(mol K)
M_st      = 25;            % Mass of the fuel cell stack (kg)
C_st      = 4000;          % specific heat of the FC stack, J/(kg K)
m_c       = 1;             % mass flow rate of the coolant (kg/s)
h         = 15;            % CHTC for air_Natural convection (W/m2 K)
A_st      = 1;             % Stack outer surface area (m2)
err       = 0.01;          % Error (initialization values = 0.01)
j         = 0;
k         = 0;
dTst_dt   = 0;
```

Fuel Heat Generation Nerst Equation

```
%I_fc = [1 : 598];
for i = 1 : 598

    I_fc(i) = i;
while (err > 0.001)                                                    %while loop for ...
    cycle calc

Roh_fuel(i) = (P_fuel * 10^5) / (T_fc(i) * 4.1242 * 1000);           %density of fuel ...
    (kg/m3)- R_H = 4.1242 kJ/KgK
Roh_Air(i)  = (P_air * 10^5) / (T_fc(i) * 0.2871 * 1000);            %density of air ...
    (kg/m3)- R_air = 0.2871 kJ/KgK
V_fuel(i)   = ((63 * 10^-5) * N_fc * I_fc(i)) / Roh_fuel(i);        %fuel flow rate (l/min)
V_air(i)    = ((214.2 * 10^-4) * N_fc * I_fc(i) * Lemda_fc) / Roh_Air(i); %air flow ...
    rate (l/min)
x = 0.99995;                                                           %percentage of ...
    hydrogen in the fuel (%)
y = 0.21;                                                              %percentage of ...
    oxygen in the oxidant (%)

U_fH(i) = (60000 * R * T_fc(i) * I_fc(i)) / (z * F * P_fuel * 10^5 * V_fuel(i) * x);
U_fO(i) = (60000 * R * T_fc(i) * I_fc(i)) / (2 * z * F* P_air * 10^5 * V_air(i) *  y);
P_H(i)  = (1 - U_fH(i)) * x * P_fuel;
P_O(i)  = (1-U_fO(i)) * y * P_air;
E_Nerst(i) = (1.229) - (8.5 * 10^(-4) * (T_fc(i) - 298.15)) +...
```

```
    (4.308 * 10^(-5) * T_fc(i) * (log(P_H(i)) + 0.5 * log(P_O(i)))); % Pressure ...
        should be in atm
```

Calculation of Losses Activation Loss

```
zeta_1 = -0.944;
zeta_2 = 0.0035;
zeta_3 = 0.000075;
zeta_4 = -0.000208;

C_O(i) = (P_O(i)) / (5.08 * 10^6 * exp(-498 / T_fc(i)));    %Pressure should be in (atm)
V_act(i) = - (zeta_1 + (zeta_2 * T_fc(i)) + (zeta_3 * T_fc(i) * log(C_O(i))) +...
                    (zeta_4 * T_fc(i) * log(I_fc(i))));
```

Ohmic Loss

```
Lemda_m = 10;
L_M = 0.004;              % thickness (cm)
A_M = 237;                % membrane area (cm2)
R_C = 0.0001;             % Rc is a constant value
J(i) = I_fc(i) / A_M;     %current density (A/cm2)
Roh_M(i) = (181.6 * (1 + (0.03 .* J(i)) +...
           (0.062 * ((T_fc(i) / 303.15)^2) .* J(i) .^ 2.5))) /...
           ((Lemda_m - 0.634 - 3 .* J(i)) .* (exp(4.18 * (T_fc(i) - (303.15 / ...
               T_fc(i))))));
R_M(i) = (Roh_M(i) * L_M) / A_M;
V_ohm(i) = I_fc(i) * (R_M(i) + R_C);
```

Concentration Loss

```
I_max = 600; % Maximum current (A)
J_max = I_max / A_M; % maximum current density (A/cm2)
V_conc(i) = -(((R * T_fc(i)) / (2 * F)) * log(1 - (J(i) / J_max)));
```

Fuel Performance

```
V_fc(i)   = E_Nerst(i) - V_act(i) - V_ohm(i) - V_conc(i);   %VFC output voltage each ...
    fuel cell (V)
V_sfc(i) = N_fc * V_fc(i);          %VS;FC output voltage the fuel cell stack (V)
P_fc(i)   = V_sfc(i) .* I_fc(i);     % unit - W
```

Heat Production term

```
Nrec_ca_O2(i) = N_fc * I_fc(i) / (4 * F);
Nrec_an_H2(i) = N_fc * I_fc(i) / (2 * F);
Ngen_ca_H2O(i) = N_fc * I_fc(i) / (2 * F);
Q_tot(i) = H_h2 * Nrec_an_H2(i) * 1000; % Unit W - (kJ/mol * mol/s *1000)
Q_gen(i) = Q_tot(i) - P_fc(i);
```

Fuel cell Heat Dissipation Model Assumptions

```
Tin_ca = 45 + 273.15; % Temperature of inlet stream at cathode (K)
Tin_an = 35 + 273.15; % Temperature of inlet stream at anode (K)
Tout_ca(i) = T_fc(i);
Tout_an(i) = T_fc(i);
```

Inlet

```
Nin_an_H2(i) = Nrec_an_H2(i);
Nin_ca_air(i) = Lemda_fc * Nrec_ca_O2(i)/0.21;
```

```
P_sat(i) = 1000 * exp(9.3876 - 3826.36/(T_fc(i) - 45.47));
%P_sat(i) = ...
    10^((-1.69*(10^-10)*T_fc(i)^4)+(3.85*(10^-7)*T_fc(i)^3)-(3.39*(10^-4)*T_fc(i)^2)+(0.143*T_fc(i))-20.92)
         %The saturation vapor pressure Psat (Kpa) according to vapor temperature T ...
    (kPa)

Nin_an_H2O(i) = (P_sat(i) / ((P_fuel * 100) - P_sat(i))) * Nin_an_H2(i);  %at ...
    humidification at anode
Nin_ca_H2O(i) = (P_sat(i) / ((P_air * 100) - P_sat(i))) * Nin_ca_air(i);  %at ...
    humidification at cathode
Qin(i)= ((Nin_an_H2(i) * C_H2) + (Nin_an_H2O(i) * C_H2O)) * (Tin_an - To) +...
         ((Nin_ca_air(i) * C_air) + (Nin_ca_H2O(i) * C_H2O)) * (Tin_ca - To);
```

Outlet

```
Nout_an_H2(i) = Nin_an_H2(i) - Nrec_an_H2(i);
Nout_ca_N2(i) = 0.79 * Nin_ca_air(i);
Nout_ca_O2(i) = (0.21 * Nin_ca_air(i)) - Nrec_ca_O2(i);
%P_SAT_OUT AND CHANGE THE NEXT 2 LINES
Nout_an_H2O(i) = (P_sat(i) / ((P_fuel * 100) - P_sat(i))) * Nout_an_H2(i);     %at ...
    Tout_an
Nout_ca_H2O(i) = (P_sat(i) / ((P_air * 100) - P_sat(i))) * (Nout_ca_N2(i) + ...
    Nout_ca_O2(i));     %at Tout_ca
Nout_ca_H2O_l(i) = Nin_an_H2O(i) + Nin_ca_H2O(i) + Ngen_ca_H2O(i) - Nout_an_H2O(i) - ...
    Nout_ca_H2O(i);
Qout(i) = ((Nout_an_H2O(i) * C_H2O) + (Nout_an_H2(i) * C_H2)) * (Tout_an(i) - To) +...
    ((Nout_ca_O2(i) * C_O2) + (Nout_ca_N2(i) * C_N2) + ...
    (Nout_ca_H2O(i) * C_H2O) + (Nout_ca_H2O_l(i) * C_H2O_l)) * (Tout_ca(i) - To);
```

Heat Factors

```
Q_gas(i) = Qout(i) - Qin(i);                          % unit (W)
Q_C(i) = (m_c * C_H2O_l * (T_fc(i) - To))/18;         % unit (W)
Q_amb(i) = h * A_st * (T_fc(i) - To);                 % unit (W)
Q_dis(i) = Q_gas(i) + Q_C(i) + Q_amb(i);              % unit (W)
```

Heat Balance

```
dTst_dt(i) = (Q_gen(i) - Q_dis(i)) / (M_st * C_st);
%For steady state condition, at any current value, rate of change of
%temperature with time should be zero. Hence dT/dt can be used as the
%parameter for to check the validity of formulation.
```

```
err = abs(dTst_dt(i));           %err is the error which need to be corrected to get ...
    the correct value
T_fc(i) = T_fc(i) + dTst_dt(i);    %the error value is added to the periouv temp ...
    value for next iteration
end

err = 0.01;                      % error value need to be reset to a higher value to ...
    start the while loop again
T_fc(i + 1) = T_fc(i);             %next loop run for i=2, which ask for T_fc(2) value, ...
    which need to be assigned before loop begin
Tfc(i)=T_fc(i);                  % however this will result in 1 higher index for T_fc ...
    than I_fc, hence this line is for index correction

end


plot(I_fc,Tfc,'-r')
```

5.2 Code for Transient conditions

Standard Commands

```
clear all
close all
clc
```

Input parameters and constants

```
R         = 8.3145;       % Univeral gas constant J/(mol K)
F         = 96485;        % Farady's contant (A s/mol)
z         = 2;            % number of moving electrons (z = 2)
N_fc      = 370;          % number of fuel cells in the stack
T_fc      = 338;          % Operating temperature of FC (K)
P_fuel    = 1.45;         % absolute supply pressure of fuel (atm)
P_air     = 1.02;         % absolute supply pressure of air (atm)
Lemda_fc  = 5;            % stoichiometric ratio of FC
H_h2      = 285.8;        % change in enthalpy of FC reaction (kJ/mol)
C_H2      = 28.836;       % specific heat of the H2, J/mol/K
C_H2O     = 36.5;         % specific heat of the water vaoour, ...
                          % Cp = 36.5 J/(mol K) at 100C, Cv= 27.5 J/(mol K) at 100C
To        = 30+273.15;    % temperature on standard pressure,K
C_air     = 1;            % specific heat of the air, J/(mol K)
C_N2      = 1.04;         % specific heat of the N2, J/(mol K)
C_H2O_l   = 75.2;         % specific heat of the H2O, J/(mol K)
C_O2      = 29.10;        % specific heat of the O2, J/(mol K)
M_st      = 25;           % Mass of the fuel cell stack (kg)
C_st      = 4000;         % specific heat of the FC stack, J/(kg K)
m_c       = 1;            % mass flow rate of the coolant (kg/s)
h         = 15;           % CHTC for air_Natural convection (W/m2 K)
A_st      = 1;            % Stack outer surface area (m2)
err       = 0.01;         % Error (initialization values = 0.01)
j         = 0;
k         = 0;
dTst_dt   = 0;
t         = 0:2:597
t1 = 0:298
```

Fuel Heat Generation Nerst Equation

```
%I_fc = [1 : 598];
for i = 1 : 299
    I_fc(i) = i;
%while (err > 0.001)                                                %while loop for ...
    cycle calc

Roh_fuel(i) = (P_fuel * 10^5) / (T_fc(i) * 4.1242 * 1000);       %density of fuel ...
    (kg/m3)- R_H = 4.1242 kJ/KgK
Roh_Air(i)  = (P_air * 10^5) / (T_fc(i) * 0.2871 * 1000);        %density of air ...
    (kg/m3)- R_air = 0.2871 kJ/KgK
V_fuel(i)   = ((63 * 10^-5) * N_fc * I_fc(i)) / Roh_fuel(i);     %fuel flow rate (l/min)
V_air(i)    = ((214.2 * 10^-4) * N_fc * I_fc(i) * Lemda_fc) / Roh_Air(i); %air flow ...
    rate (l/min)
x = 0.99995;                                                      %percentage of ...
    hydrogen in the fuel (%)
y = 0.21;                                                         %percentage of ...
    oxygen in the oxidant (%)

U_fH(i) = (60000 * R * T_fc(i) * I_fc(i)) / (z * F * P_fuel * 10^5 * V_fuel(i) * x);
U_fO(i) = (60000 * R * T_fc(i) * I_fc(i)) / (2 * z * F* P_air * 10^5 * V_air(i) *  y);
P_H(i)  = (1 - U_fH(i)) * x * P_fuel;
P_O(i)  = (1-U_fO(i)) * y * P_air;
E_Nerst(i) = (1.229) - (8.5 * 10^(-4) * (T_fc(i) - 298.15)) +...
    (4.308 * 10^(-5) * T_fc(i) * (log(P_H(i)) + 0.5 * log(P_O(i)))); % Pressure ...
        should be in atm
```

Calculation of Losses Activation Loss

```
zeta_1 = -0.944;
zeta_2 = 0.0035;
zeta_3 = 0.000075;
zeta_4 = -0.000208;

C_O(i) = (P_O(i)) / (5.08 * 10^6 * exp(-498 / T_fc(i)));    %Pressure should be in (atm)
V_act(i) = - (zeta_1 + (zeta_2 * T_fc(i)) + (zeta_3 * T_fc(i) * log(C_O(i))) +...
                    (zeta_4 * T_fc(i) * log(I_fc(i))));
```

Ohmic Loss

```
Lemda_m = 10;
L_M = 0.004;              % thickness (cm)
A_M = 237;                % membrane area (cm2)
R_C = 0.0001;             % Rc is a constant value
J(i) = I_fc(i) / A_M;     %current density (A/cm2)
Roh_M(i) = (181.6 * (1 + (0.03 .* J(i)) +...
           (0.062 * ((T_fc(i) / 303.15)^2) .* J(i) .^ 2.5))) /...
           ((Lemda_m - 0.634 - 3 .* J(i)) .* (exp(4.18 * (T_fc(i) - (303.15 / ...
               T_fc(i))))));
R_M(i) = (Roh_M(i) * L_M) / A_M;
V_ohm(i) = I_fc(i) * (R_M(i) + R_C);
```

Concentration Loss

```
I_max = 600; % Maximum current (A)
J_max = I_max / A_M; % maximum current density (A/cm2)
V_conc(i) = -(((R * T_fc(i)) / (2 * F)) * log(1 - (J(i) / J_max)));
```

Fuel Performance

```
V_fc(i)  = E_Nerst(i) - V_act(i) - V_ohm(i) - V_conc(i);   %VFC output voltage each ...
    fuel cell (V)
V_sfc(i) = N_fc * V_fc(i);        %VS;FC output voltage the fuel cell stack (V)
P_fc(i)  = V_sfc(i) .* I_fc(i);    % unit - W
```

Heat Production term

```
Nrec_ca_O2(i) = N_fc * I_fc(i) / (4 * F);
Nrec_an_H2(i) = N_fc * I_fc(i) / (2 * F);
Ngen_ca_H2O(i) = N_fc * I_fc(i) / (2 * F);
Q_tot(i) = H_h2 * Nrec_an_H2(i) * 1000; % Unit W - (kJ/mol * mol/s *1000)
Q_gen(i) = Q_tot(i) - P_fc(i);
```

Fuel cell Heat Dissipation Model Assumptions

```
Tin_ca = 45 + 273.15; % Temperature of inlet stream at cathode (K)
Tin_an = 35 + 273.15; % Temperature of inlet stream at anode (K)
Tout_ca(i) = T_fc(i);
Tout_an(i) = T_fc(i);
```

Inlet

```
Nin_an_H2(i) = Nrec_an_H2(i);
Nin_ca_air(i) = Lemda_fc * Nrec_ca_O2(i)/0.21;
P_sat(i) = 1000 * exp(9.3876 - 3826.36/(T_fc(i) - 45.47));
%P_sat(i) = ...
    10^((-1.69*(10^-10)*T_fc(i)^4)+(3.85*(10^-7)*T_fc(i)^3)-(3.39*(10^-4)*T_fc(i)^2)+(0.143*T_fc(i))-20.92)
         %The saturation vapor pressure Psat (Kpa) according to vapor temperature T ...
    (kPa)
```

```

Nin_an_H2O(i) = (P_sat(i) / ((P_fuel * 100) - P_sat(i))) * Nin_an_H2(i);  %at ...
    humidification at anode
Nin_ca_H2O(i) = (P_sat(i) / ((P_air * 100) - P_sat(i))) * Nin_ca_air(i);  %at ...
    humidification at cathode
Qin(i)= ((Nin_an_H2(i) * C_H2) + (Nin_an_H2O(i) * C_H2O)) * (Tin_an - To) +...
         ((Nin_ca_air(i) * C_air) + (Nin_ca_H2O(i) * C_H2O)) * (Tin_ca - To);
```

Outlet

```
Nout_an_H2(i) = Nin_an_H2(i) - Nrec_an_H2(i);
Nout_ca_N2(i) = 0.79 * Nin_ca_air(i);
Nout_ca_O2(i) = (0.21 * Nin_ca_air(i)) - Nrec_ca_O2(i);
%P_SAT_OUT AND CHANGE THE NEXT 2 LINES
Nout_an_H2O(i) = (P_sat(i) / ((P_fuel * 100) - P_sat(i))) * Nout_an_H2(i);     %at ...
    Tout_an
Nout_ca_H2O(i) = (P_sat(i) / ((P_air * 100) - P_sat(i))) * (Nout_ca_N2(i) + ...
    Nout_ca_O2(i));     %at Tout_ca
Nout_ca_H2O_l(i) = Nin_an_H2O(i) + Nin_ca_H2O(i) + Ngen_ca_H2O(i) - Nout_an_H2O(i) - ...
    Nout_ca_H2O(i);
Qout(i) = ((Nout_an_H2O(i) * C_H2O) + (Nout_an_H2(i) * C_H2)) * (Tout_an(i) - To) +...
    ((Nout_ca_O2(i) * C_O2) + (Nout_ca_N2(i) * C_N2) + ...
    (Nout_ca_H2O(i) * C_H2O) + (Nout_ca_H2O_l(i) * C_H2O_l)) * (Tout_ca(i) - To);
```

Heat Factors

```
Q_gas(i) = Qout(i) - Qin(i);                             % unit (W)
Q_C(i) = (m_c * C_H2O_l * (T_fc(i) - To))/18;            % unit (W)
Q_amb(i) = h * A_st * (T_fc(i) - To);                    % unit (W)
Q_dis(i) = Q_gas(i) + Q_C(i) + Q_amb(i);                 % unit (W)
```

Heat Balance

```
dTst_dt(i) = (Q_gen(i) - Q_dis(i)) / (M_st * C_st);
%For steady state condition, at any current value, rate of change of
%temperature with time should be zero. Hence dT/dt can be used as the
%parameter for to check the validity of formulation.
```

```
T_fc(i) = T_fc(i) + dTst_dt(i);    %the error value is added to the periouv temp ...
    value for next iteration
%end
                % error value need to be reset to a higher value to start the while ...
                    loop again
T_fc(i + 1) = T_fc(i);          %next loop run for i=2, which ask for T_fc(2) value, ...
    which need to be assigned before loop begin
Tfc(i)=T_fc(i);              % however this will result in 1 higher index for T_fc ...
    than I_fc, hence this line is for index correction

end

hold on
grid on
title('Comparisons of different current-time relationship vs Time')
xlabel('time')
ylabel('Temperature')
plot(t,Tfc,'-r')
plot(t1,Tfc,'-b')
hold off
```