

Task 1

For the first task, we evaluated the enthalpy of reaction, equilibrium constant, and equilibrium conversion as a function of temperature. To get the enthalpy of reaction at each temperature, we first calculated $\sum \nu C_p$ by multiplying each heat capacity expression with its stoichiometric coefficient. Then, we integrated $\sum \nu C_p$ from 298 K to T and used Equation 1 from the Appendix to calculate the enthalpy of reaction in Matlab, where $\Delta H_0^\circ(298 \text{ K})$ is -49500 J/mol. Looking at the graph, from 250K to 600K, the heat of reaction is decreasing as the temperature increases (Figure A). As the temperature increases, the heat of reaction becomes more negative as a result of higher temperatures differences when we integrate the sensible heat.

For calculating K_{eq} , we used equation 2 to calculate $\Delta G/RT$ which is also equal to $-\ln K$. The integral terms in that expression were also integrated from 298 K to some temperature T. As we plot the equilibrium constant against temperature, we noticed an immediate steep slope down the graph, and eventually the curve approaches 0 after 325K. (Figure B, blue curve). This is because for an exothermic reaction, increases in temperature favor the reverse reaction; Le Châtelier's principle explains that the reaction will proceed in such a way as to counteract the temperature change to maintain equilibrium. Also expected, equilibrium conversion decreases as temperature increases shown in Figure B. When the equilibrium constant decreases, less product is formed since the reaction favors the reactants side. The equilibrium conversion became near zero after 525 K, indicating that very little methanol and water are produced if we raise the temperature above 525 K.

Task 2

In task 2, we modeled an isothermal PBR at 500, 533, and 566 K and determined the conversion profile against length down the reactor. To solve for this profile, we used the differential expression from Equation 4 in the Appendix. Based on the results from the graph (Figure C), the best temperature to operate the reactor was 500 K. This is because the exiting (equilibrium) conversion is the highest at this temperature, which was 0.0054. The other exiting conversions were 0.0024 and 0.0011 at 533 and 566 K respectively. These lower conversions were expected because as temperature increases, the equilibrium constant decreases, thus lowering the equilibrium conversion. Also, equilibrium is reached faster since the higher temperature curves level off faster than the 500 K curve. That said, it should be noted that final conversion at 500K is only good in the context of an impractical reactor premise. The space-time of the feed in the reactor is ~10 million hours. At shorter, more reasonable space times, the equilibrium constant will be far less relevant since the reaction will not come close to reaching equilibrium.

At this chosen temperature of 500 K, we then varied the mass of catalyst in the reactor and saw how doing so affects conversion across the length of the reactor. Conversion was evaluated using 5 different masses of catalyst: 0.5, 1, 1.5, 2, and 2.5 kg. As shown in the graph (Figure D), increasing the mass of catalyst increases the rate at which equilibrium is achieved. This makes sense because more catalyst would result in a faster reaction. Despite changing the mass of catalyst, all of the curves in Figure D eventually converge to the same equilibrium conversion because the amount of catalyst does not change the reaction equilibrium. The

immediate conversion will increase as a result of a faster reaction, but the equilibrium conversion will not change as the catalyst only affects the kinetics of the reaction.

Task 3

In this part of the assignment, we evaluated the conversion and temperature profiles as they relate to the length of a non-isothermal adiabatic PBR. We solved a system of differential equations to determine these profiles, and these differential equations are shown in the Appendix (Equation 4 and 5); both of those equations are functions of T and X and solving them together would then result X and T down the length of the PBR.

From the graphs, conversion and temperature both increase until a certain point in the reactor. The temperature increases because this is an exothermic reaction, and as we travel further down the reactor, the reaction produces more product, releasing more heat and increasing the conversion. The graph, however, reaches its maximum conversion (about 0.0023) and temperature (533.86 K) about 6 m into the reactor as shown by the plateauing graph in Figure E. Conversion and temperature do not keep increasing because the reaction is limited by equilibrium. Based on these results, a 20-meter reactor is not optimal as the conversion and temperature plateaus at around 6 meters. From these observations, we suggest using a shorter reactor. This would decrease needed space, and it would be a less costly reactor.

Compared to the 533 K isothermal reactor from task 2, the graphs of conversion are very similar in shape, and the maximum conversions are also about the same. In task 2, the max conversion reached was 0.0024 while in task 3, it was 0.0023. In the case of task 2, equilibrium was reached around 6 m into the reactor. The only difference between the 2 is that the end temperature from task 3 was about 533.86 K thus explaining why the maximum conversion is lower.

Task 4

In Aspen, we modeled an isothermal PBR at 500 K. We chose this temperature because, in task 2, the highest conversion was achieved at this temperature. Compared to task 2, both conversions were very similar to each other; from task 2, the conversion at 500 K was about 0.0054, and in Aspen, the conversion was about 0.0055 (based on the inlet and outlet hydrogen molar flow rates). This discrepancy occurs probably because Aspen used PSRK to model gas behavior whereas in task 2, we used ideal gas law. The value from Aspen is probably more accurate since gases under pressures of 50 bar do not behave ideally.

For the kinetic model, we chose the reaction type as general, set the reaction as reversible, and chose LHHW as the reaction class. In order to fit the rate law to the LHHW model, we rearranged it to this format:

$$r = k * \frac{[KP_{CO2}P_{H2}^3 - P_{MeOH}P_{H2O}]}{KP_{CO2}^{0.5}P_{H2}^{1.5}}$$

All the partial pressures in this rate law were in bars and not MPa, and so the 0.01 term in the project handout becomes 1. The rate is based on weight of catalyst with units of kmol/(kg*hr), and the concentration basis is partial pressure. The variable k in the rate law is called the “kinetic factor”, and the inputs to Aspen are “k” (pre-exponential factor) and “E” (activation energy). The constant k is 8.89×10^{-5} because the units of pressure in the rate is bar, and in the original handout, k has a MPa^2 in the denominator ($\text{MPa}^2 = 100 \text{ bar}^2$); E can be found by multiplying 6163 by 8.314 J/mol*K. The bracketed term is known as the “driving force”, and it has 2 parts, forward and reverse; the reverse part is the term with the minus sign. We input the corresponding

exponents of the partial pressures in the forward and reverse terms, and to input the equilibrium constant, we calculated K at 500 K (which was 3.7704×10^{-5}) since the reactor was isothermal, and in Aspen, it takes in inputs A, B, C, and D to calculate K where $\ln(K) = A + B/T + C\ln(T) + DT$ (T being temperature). However, since K is constant, we only input the parameter A and everything else as 0. For the reverse part, we specified K as 1 (i.e. $A = 0$) since we needed to input some constant. The term in the denominator is known as the “adsorption” term, and the process for inputting the required parameters is similar to before, except the partial pressure components involved in this expression have to be explicitly specified; then the exponents and the equilibrium constants are put in the same way.

After running the original simulation, we changed certain variables to see how conversion would be affected. One of the variables we changed was the reactor pressure, and as pressure increased, the conversion also increased (Figure F). This can be explained with Le Chatelier’s principle; as the pressure increases, the gas phase reaction will shift to produce the least number of gas molecules to counteract the pressure increase, and so, this would be the products side since there are 2 molecules versus 4. With more methanol and water produced, the conversion becomes higher. Another variable we changed was the inlet molar flow rate of CO_2 . As this flow rate increased, the conversion initially increased and then started to decrease (Figure G). We expected for the conversion to increase since we thought that increasing the amount of reactants would shift the reaction to produce more products. In this case, it is partially true; however, it seems that increasing the flow rate past a certain value results in the reactants spending less time in the PBR; thus, there is enough time for the reactants to form the desired products. As shown in the graph, the conversion gets close to 0, meaning the outlet molar flow of hydrogen is nearly the same as the inlet.

Appendix Equations

$$1) \Delta H^\circ = \Delta H_0^\circ + \int_{T_0}^T \Delta C p^\circ dT, \quad \Delta C p^\circ = \sum v_j C p_j^\circ$$

$$2) \frac{\Delta G^\circ}{RT} = -\ln K = \frac{\Delta H_0^\circ}{RT} + \frac{\Delta G_0^\circ - \Delta H_0^\circ}{RT_0} + \frac{1}{RT} \int_{T_0}^T \Delta C p^\circ dT - \frac{1}{R} \int_{T_0}^T \frac{\Delta C p^\circ}{T} dT$$

$$3) K_{fun} = \frac{0.01 \left(\frac{P_{H2,0} + X}{3} \right) \left(\frac{P_{H2O,0} + \frac{P_{H2,0}}{3} X}{1 + \epsilon X} \right)}{\left(\frac{P_{CO2,0} - \frac{P_{H2,0}}{3} X}{1 + \epsilon X} \right) \left(\frac{P_{H2,0}(1-X)}{1 + \epsilon X} \right)^3} - z \text{ where } z = \text{calculated K value at a certain T}$$

$$4) \frac{dX}{dL} = \frac{A(-r_A)}{F_{A0}} \text{ where } A = \text{cross sectional area and } r_A \text{ is normalized by volume}$$

$$5) \frac{dX}{dT} = \frac{FA0 * (dX/dL) * \Delta H_{(T)}}{-FA0 * X * \Delta C p - \sum F_{j0} * C p_j} \text{ where } \Delta H_{(T)} \text{ is evaluated using equation 1}$$

Graphs

Figure A (task 1)

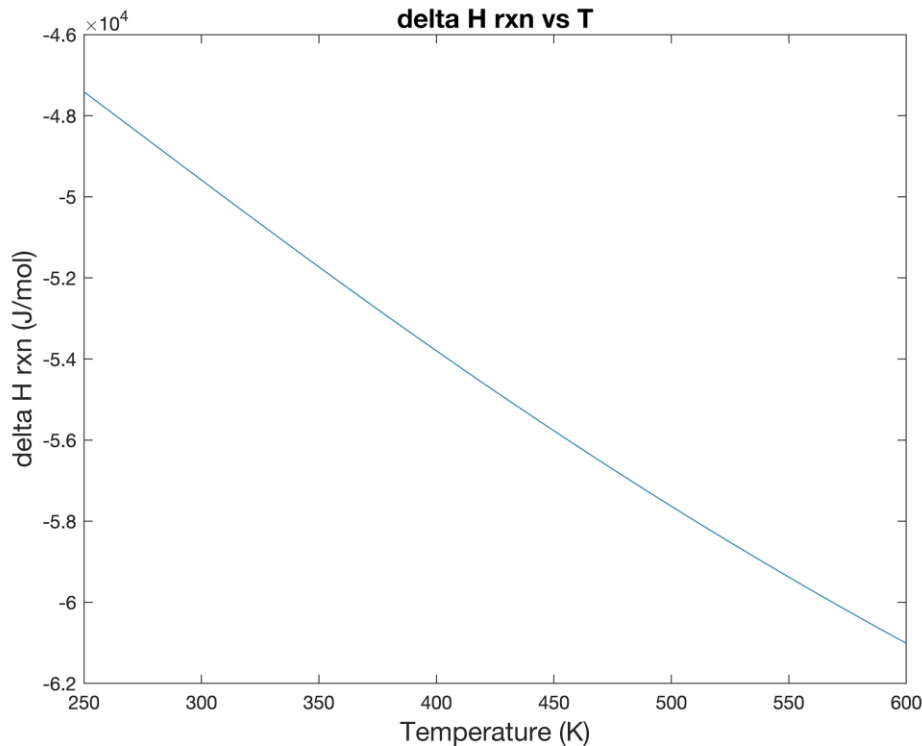


Figure B (task 1)

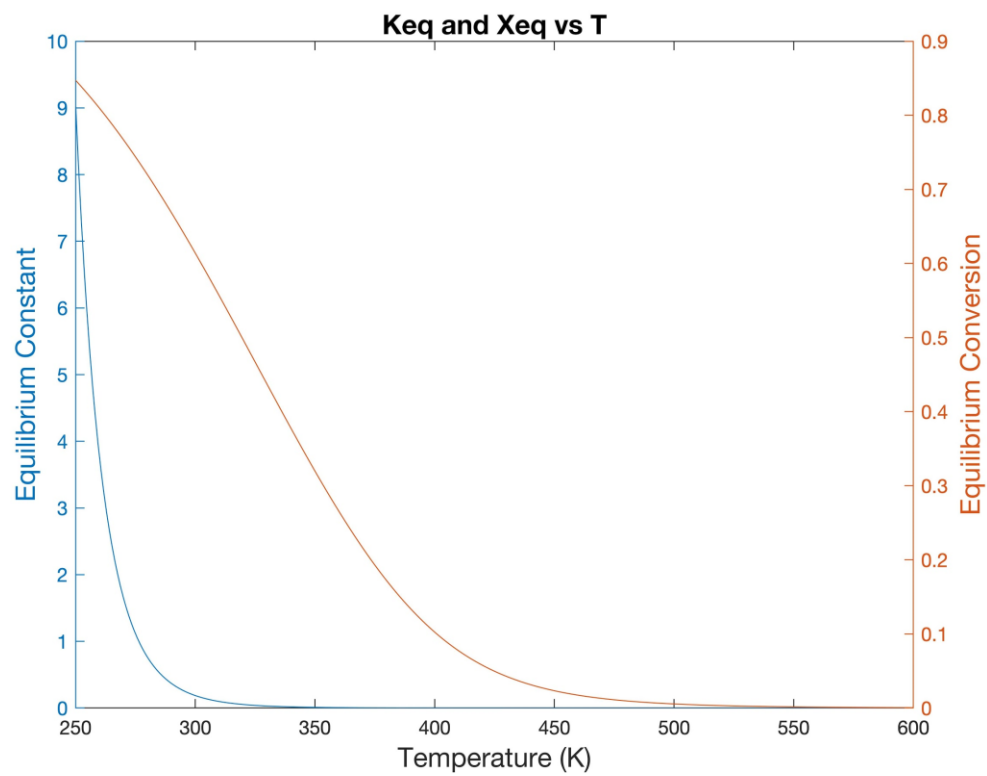


Figure C (task 2)

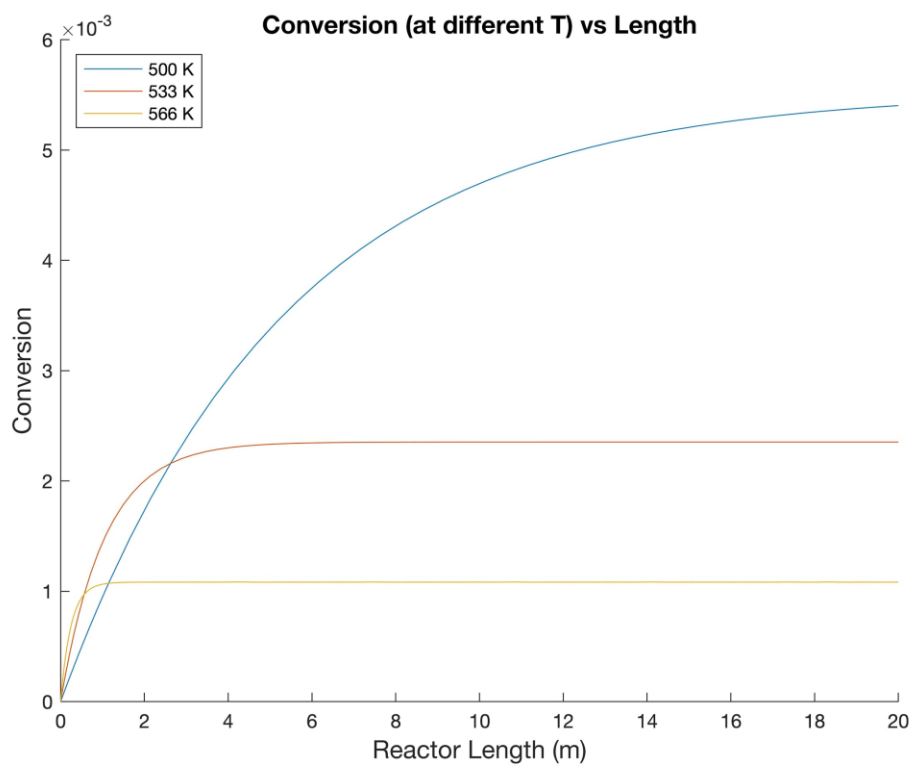


Figure D (task 2)

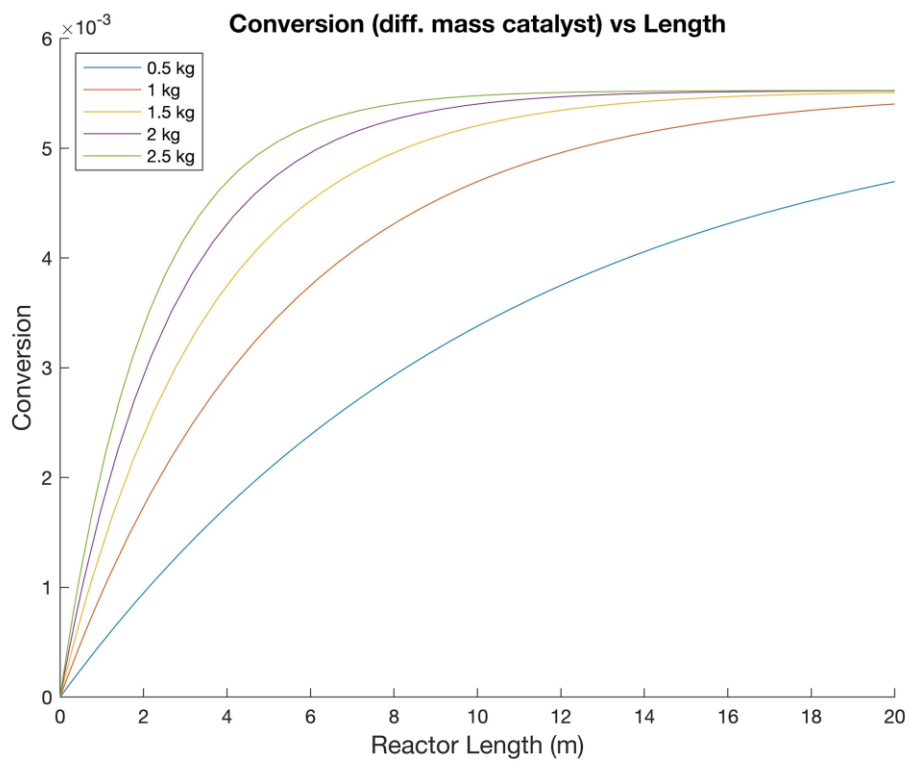


Figure E (task 3)

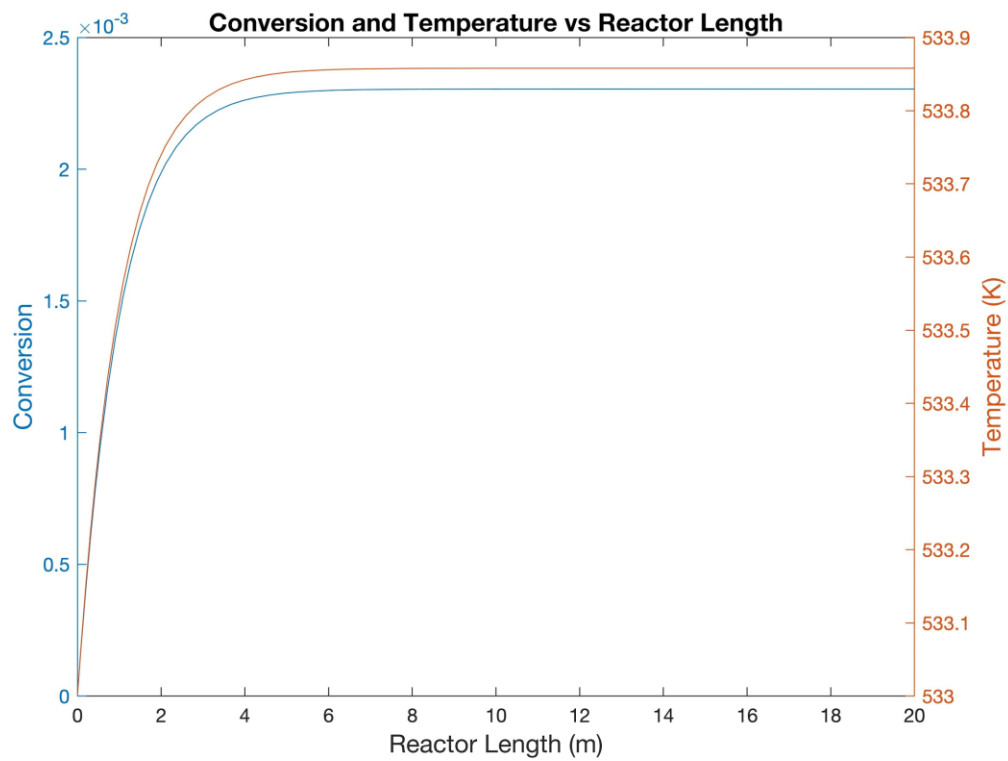


Figure F (task 4)

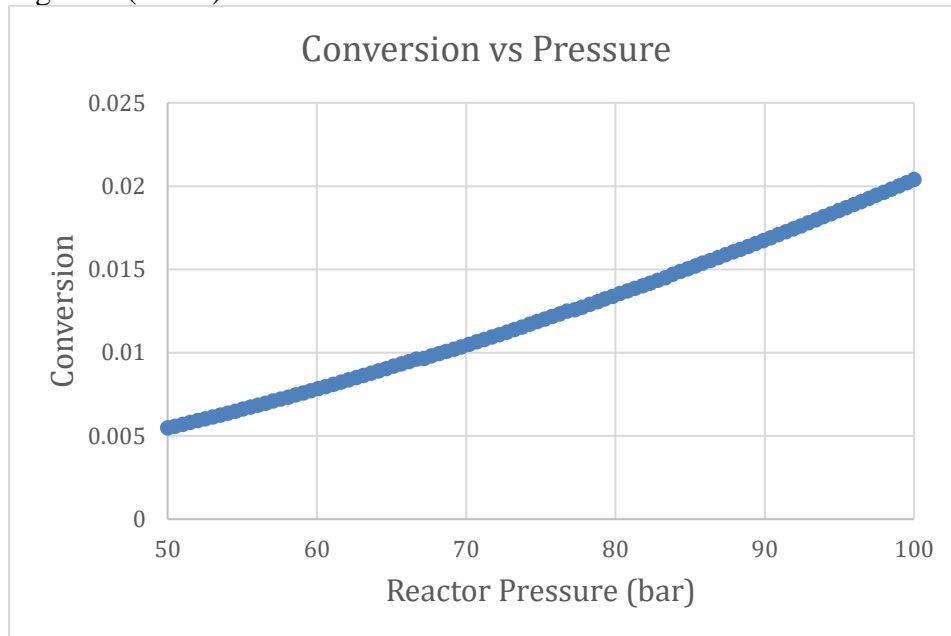
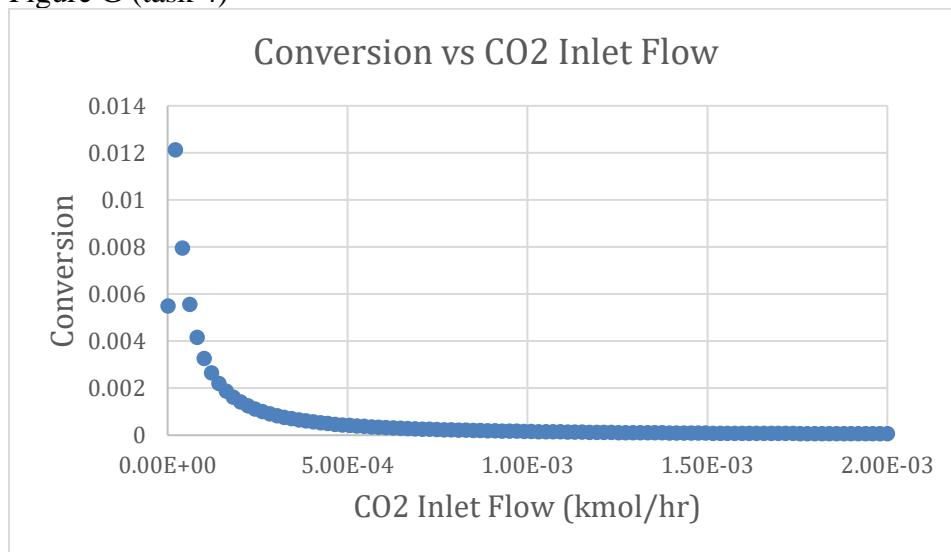


Figure G (task 4)



Code

Task 1

```
%Task 1 a
%Creates a temperature vector with 1000 entries from 250 K to 600 K
T = linspace(250,600,1000);

%intdCp is an anonymous function that takes in an input T (temperature) and
%calculates intdCp where this is an integrated expression of the delta Cp
%based on the rxn (Cpmeoh + Cph2o - Cpco2 -3*Cph2) from 298 K to T
intdCp = @(T) (93791893782023919*T.^2)/2305843009213693952 - 719600./T -...
(9193*T)/125 - (4943451940375993*T.^3)/590295810358705651712 +...
(3265550423943037*T.^4)/9671406556917033397649408 + 20937.6652723069;
```

```

%Using the equation deltaH= deltaH0+integration of delta Cp from 298 K to T
dH = -49500 + intdCp(T);

%Plot DeltaH against Temperature
plot(T,dH)
title('delta H rxn vs T','fontsize',14)
xlabel('Temperature (K)','fontsize',14)
ylabel('delta H rxn (J/mol)','fontsize',14)

%Task 1 b
%another anonymous function, intdCp_T is the integrated expression of delta
%Cp divided by T evaluated from 298 K to T
intdCp_T = @(T) (93791893782023919*T)/1152921504606846976 - (9193*log(T))/125
-...
359800./T.^2 - (14830355821127979*T.^2)/1180591620717411303424 + ...
(3265550423943037*T.^3)/7253554917687775048237056 + 399.899539846063;

%Substitute T to get the value of deltaCp/T
dS = intdCp_T(T);

%calculate deltaG/RT with delta Cp and deltaCp/T calculated above
dGrt = (-49500/8.314./T) + (3827+49500)/(8.314*298) + ...
(1/8.314./T).*intdCp(T) - (1/8.314)*dS;

%calculate K equilibrium using deltaG/RT
K = exp(-dGrt);

%Task 1 c
%Total Pressure
P0 = 5; %MPa
%Calculate the partial pressure of components
Pa0 = 0.2*P0; %H2
Pb0 = 0.07*P0; %CO2
Pc0 = 0.14*P0; %H2O
Xeq = [];

%this for loop uses fzero to solve the non linear equation (Kfun)
%at each K equilibrium with an initial guess of 0.3, The whole big term with
%conversion in it is supposed to be an expression for K, and z is the already
%calculated K values at each temperature from the previous section. Thus
%Kfun should be 0 when the correct x value is found.
for m = 1:length(K)
    z = K(m);
    Kfun = @(x) 0.01*((Pa0/3*x)/(1+(-2/3*0.2)*x))*((Pc0 + Pa0/3*x)/(1+(-
2/3*0.2)*x))/...
((Pb0 - Pa0/3*x)/(1+(-2/3*0.2)*x))*(Pa0*(1-x)/(1+(-2/3*0.2)*x))^3 -
z;
    roots = fzero(Kfun,0.3);
    Xeq = [Xeq roots];
end

%Plot Keq and Xeq against Temperature
figure
yyaxis left

```



```

plot(T,K)
ylabel('Equilibrium Constant','fontsize',14)

yyaxis right
plot(T,Xeq)
title('Keq and Xeq vs T','fontsize',14)
xlabel('Temperature (K)','fontsize',14)
ylabel('Equilibrium Conversion','fontsize',14)

```

Task 2

```

%this allows T & cat to be defined globally rather than locally within the
%scopes of certain functions (ie, the value of T in one function will be
%the same value of T in another function)
global T cat
tempvec = [500 533 566]; %temperature vector in K

%this for loop changes the temperature of the reactor and plots the
%conversions from each T
actualX = [];
for m = 1:3
    T = tempvec(m);
    cat = 1; %mass of catalyst kg
    [L, X] = ode45(@deriv,[0 20],0);

    hold on
    plot(L,X)
    title('Conversion (at different T) vs Length','fontsize',13)
    xlabel('Reactor Length (m)','fontsize',13)
    ylabel('Conversion','fontsize',13)

    actualX = [actualX X(end)]; %conversions at the end of reactor for 500,
533, and 566 K
end
legend('500 K','533 K','566 K','location','northwest')
hold off

%this for loop changes the mass of catalyst in the reactor and plots
%conversion based on those masses
%we are using T as 500 K since this gives us the highest conversion
T = 500;
masscat = [0.5 1 1.5 2 2.5]; %kg of catalyst
actualXcat = [];
figure
for n = 1:5
    cat = masscat(n); %mass of catalyst in PBR
    [L, X] = ode45(@deriv,[0 20],0);

    hold on
    plot(L,X)
    title('Conversion (diff. mass catalyst) vs Length','fontsize',13)
    xlabel('Reactor Length (m)','fontsize',13)
    ylabel('Conversion','fontsize',13)
    actualXcat = [actualXcat X(end)]; %conversions at end of reactor for
different mass of catalyst

```

```

end
legend('0.5 kg','1 kg','1.5 kg','2 kg','2.5 kg','location','northwest')

%differential expression function
function dxdL = deriv(L,X)

global T cat % T & cat outside the scope of this function is the same as T &
cat in this function

n0 = 0.382*(1/1.01325)/0.08206/273.15; %mol/hr, total initial molar flow
Fa0 = .2*n0; %H2 initial molar flow in mol/hr
pa0 = 0.2*5; %H2 initial partial P in MPa
pb0 = 0.07*5; %CO2 initial partial P
pc0 = 0.14*5; %H2O initial partial P

%dGrt is used to evaluate K at different T
%this is the very long expression for delta G/(RT) evaluated from 298 K to
some temperature T
dGrt = (3827+49500)/(8.314*298) + (-49500/8.314./T) + (1./(8.314*T)).*...
(93791893782023919*T.^2)/2305843009213693952 - 719600./T - (9193*T)/125
-...
(4943451940375993*T.^3)/590295810358705651712 +
(3265550423943037*T.^4)/9671406556917033397649408 +...
20937.665) - (1/8.314)*(93791893782023919*T)/1152921504606846976 -...
(9193*log(T))/125 - 359800./T.^2 -
(14830355821127979*T.^2)/1180591620717411303424 +...
(3265550423943037*T.^3)/7253554917687775048237056 + 399.899539846063);

%this is from the expression delta G = -RTlnK
K = exp(-dGrt);

k = 8.89*exp(-6163./T); %vector of k values in mol/(kg-cat * h * MPa^2)
e = -2/3*0.2; %epsilon (1 + eX) based on using mol frac of H2

%these pressures come from ideal gas law where Pa*v = Fa*R*T0, Pa0*v0 =
Fa0*R*T0,
% Fa = Fa0(1-X), and v = v0(1+eX)
Pmeoh = (pa0/3*X)/(1+e*X);
Ph2o = (pc0 + pa0/3*X)/(1+e*X);
Pco2 = (pb0 - pa0/3*X)/(1+e*X);
Ph2 = (pa0*(1-X))/(1+e*X);

a = (0.01*Pmeoh.*Ph2o)/(K.*Pco2.*(Ph2.^3));

%rate of consumption of H2
rA = -(3*cat/(pi*2.5^2*20)*k).*(Pco2.^0.5).*(Ph2.^1.5).*(1-a);

dxdL = (pi*2.5^2).*(-rA/Fa0);
end

```

Task 3

%L is a vector the reactor length from 0 to 20 m

```
%y is a 2 column matrix of the solutions to the system diff. eqns in the code
where the 1st
% column is conversion and the 2nd is temperature
```

```
%All calculations are based on taking H2 as the limiting reagent
[L, y] = ode45(@deriv,[0 20],[0 533]);
```

```
X = y(:,1); %conversion
T = y(:,2); %temperature in K
actualX = X(end); %conversion at end of reactor
actualT = T(end); %temperature at end of reactor
```

```
yyaxis left
plot(L,X)
title('Conversion and Temperature vs Reactor Length','fontsize',13)
xlabel('Reactor Length (m)','fontsize',13)
ylabel('Conversion','fontsize',13)
```

```
yyaxis right
plot(L,T)
ylabel('Temperature (K)','fontsize',13)
```

```
%differential equations function
function dydl = deriv(l,y)
n0 = 0.382*(1/1.01325)/0.08206/273.15; %mol/hr, total initial molar flow
Fa0 = .2*n0; %H2 initial molar flow in mol/hr
Fb0 = .07*n0; %CO2 initial molar flow
Fc0 = .14*n0; %H2O initial molar flow
Fd0 = .02*n0; %O2 initial molar flow
Fe0 = .57*n0; %N2 initial molar flow
```

```
pa0 = 0.2*5; %H2 intial partial P in MPa
pb0 = 0.07*5; %CO2 intial partial P
pc0 = 0.14*5; %H2O intial partial P
```

```
X = y(1);
T = y(2);
```

```
%this is the integrated expression of delta Cp (Cpmeoh + Cph2o - Cpco2
% -3*Cph2) from 298 K to sometemperature T
intdCp = (93791893782023919*T.^2)/2305843009213693952 - 719600./T -...
(9193*T)/125 - (4943451940375993*T.^3)/590295810358705651712 +...
(3265550423943037*T.^4)/9671406556917033397649408 + 20937.6652723069;
```

```
%Cp expressions from the project handout
Cpco2 = 24.997 + 55.187/1000*T - 33.691/1000^2*T.^2 + 7.9484/1000^3*T.^3 -
0.1366*1000^2./T.^2;
Cph2 = 33.066 - 11.363/1000*T + 11.433/1000^2*T.^2 - 2.7729/1000^3*T.^3 -
0.1586*1000^2./T.^2;
Cph2o = 30.092 + 6.8325/1000*T + 6.7934/1000^2*T.^2 - 2.5345/1000^3*T.^3 +
0.0821*1000^2./T.^2;
Cpo2 = 31.332 - 20.235/1000*T + 57.866/1000^2*T.^2 - 36.506/1000^3*T.^3 -
0.0074*1000^2./T.^2;
Cpn2 = 19.506 + 19.887/1000*T - 8.5985/1000^2*T.^2 + 1.3698/1000^3*T.^3 +
0.5276*1000^2./T.^2;
```

```

%delta Cp expression of the reaction (Cpmeoh + Cph2o - Cpco2 -3*Cph2)
deltaCp = (93791893782023919*T)/1152921504606846976 + 719600./T.^2 -...
(14830355821127979*T.^2)/590295810358705651712 +...
(3265550423943037*T.^3)/2417851639229258349412352 - 9193/125;

k = 8.89*exp(-6163./T); %rate constants values in mol/(kg-cat * h * MPa^2)
e = -2/3*0.2; %epsilon (1 + eX) based on using mol frac of H2

%partial pressures at some point along the reactor
%these pressures come from ideal gas law where Pa*v = Fa*R*T, Pa0*v0 =
Fa0*R*T0,
% Fa = Fa0(1-X), and v = v0(1+eX)T/T0
Pmeoh = (pa0/3*X)/(1+e*X);
Ph2o = (pc0 + pa0/3*X)/(1+e*X);
Pco2 = (pb0 - pa0/3*X)/(1+e*X);
Ph2 = (pa0*(1-X))/(1+e*X);

%dGrt is used to evaluate K at different temperatures
%this is the very long expression for delta G/(RT) (eqn from handout 1)
dGrt = (3827+49500)/(8.314*298) + (-49500/8.314./T) + (1./(8.314*T)).*intdCp
-...
(1/8.314)*((93791893782023919*T)/1152921504606846976 - (9193*log(T))/125
-...
359800./T.^2 - (14830355821127979*T.^2)/1180591620717411303424 +...
(3265550423943037*T.^3)/7253554917687775048237056 + 399.899539846063);

%this comes from the expression delta G = -RTlnK
K = exp(-dGrt);

a = (0.01*Pmeoh.*Ph2o)/(K.*Pco2.*(Ph2.^3));
rA = -(3/(pi*2.5^2*20)*k).*(Pco2.^0.5).*(Ph2.^1.5).*(1-a); %rate of H2
consumption in mol/(m^3*hr)

%differential eqn for conversion down the length of the reactor
dxd1 = (pi*2.5^2).*(-rA/Fa0);

%differential eqn for temperature down the length of the reactor
dtd1 = (Fa0*dxd1.*(-49500 + intdCp))./(-Fa0*X.*deltaCp -(Fa0*Cph2 + Fb0*Cpco2
+ Fc0*Cph2o +...
Fd0*Cpo2 + Fe0*Cpn2));

dyd1 = [dxd1; dtd1];
end

```