PROJECT REPORT

ON

“CHEMICAL EQUILIBRIUM”

Session 2024-2026

DEPARTMENT OF MECHANICAL ENGINEERING

INDIAN INSTITUTE OF TEHCNLOGOY, GUWAHATI

Submitted To:

Tapan K. Mankodi

(Assistant Professor)

Submitted By:

Neeraj Kumar

244103315

Objective

Consider the dissociation of CO2 as a function of temperature and pressure,

$$CO_2 \rightleftarrows CO + \frac{1}{2}O_2$$

Find the composition of the mixture, i.e., the mole fractions of CO2, CO and O2, that results from subjecting originally pure CO2 to various temperatures (T = 1500, 2000, 2500, 3000, 3500 K) and pressure (0.1, 1, 10 and 100 atm).

Approach

Formulation: -

a. $\frac{1}{\sqrt{2}}x^{1.5}p^{0.5} - (1 - 1.5x)\exp\left(-\frac{G}{RT}\right) = 0$

b. $x + y + z = 1$

c. $\frac{1}{2} = \frac{x+y}{x+2y+2z}$

Where, $x =$ mole fraction of CO

$y =$ mole fraction of CO2

$z =$ mole fraction of O2

after solving these equations in term of x

$$y = 1 - \frac{3}{2}x$$

$$z = \frac{x}{2}$$

solving equations a, b and c are very difficult. So, these are easily solved by applying Newton-Raphson iteration method.

for Newton-Raphson method,

$$f = \frac{1}{\sqrt{2}} x^{1.5} p^{0.5} - (1 - 1.5x) \exp\left(-\frac{G}{RT}\right)$$

$$f\ derivative = \frac{1.5}{\sqrt{2}} (px)^{0.5} + 1.5 \exp\left(-\frac{G}{RT}\right)$$

Newton- Raphson Formula,

$$Xn = X_{n-1} - \frac{f(x_{n-1})}{f\ derivative\ (x_{n-1})}$$

this Newton-Raphson Formula is solved by using computation method in C language

Coding Algorithm

write header files

here stdio is standard input output library

#include<stdio.h>

#include <math.h>

making different functions

for finding f(x)

```
double  f(double x, double T, double P) {
        if( value of T )  {
                }
```

```
                if else{
                        //value of G;
                }
              //write code for f formula ;
             }
 for finding  f(x)  derivative
      double  f_d(double x, double T, double P) {
              if( value of T  )  {
                            }
                if else {
                 //value of G;
                }
              //write code for f derivative formula;
             }
for  N-R method
     void N-R method(double X_0, double error, double T, double P,
     FILE *file) {
     //write NR  method formula;
            }
```

using while loop for convergence

```
while (error>1e-10){
        //code;
        }
```

declaring main function

calling NR function

declaring guess value of X_1 and error

taking array for temperature and pressure

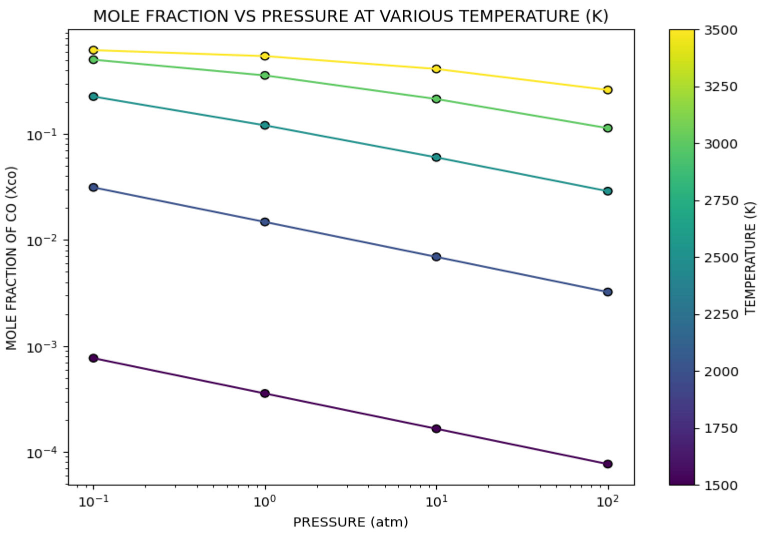
```
//double temperatures [] = {1500, 2000, 2500, 3000, 3500};
// double pressures [] = {0.1, 1, 10, 100};
```

writing a code for output file

Result/Output

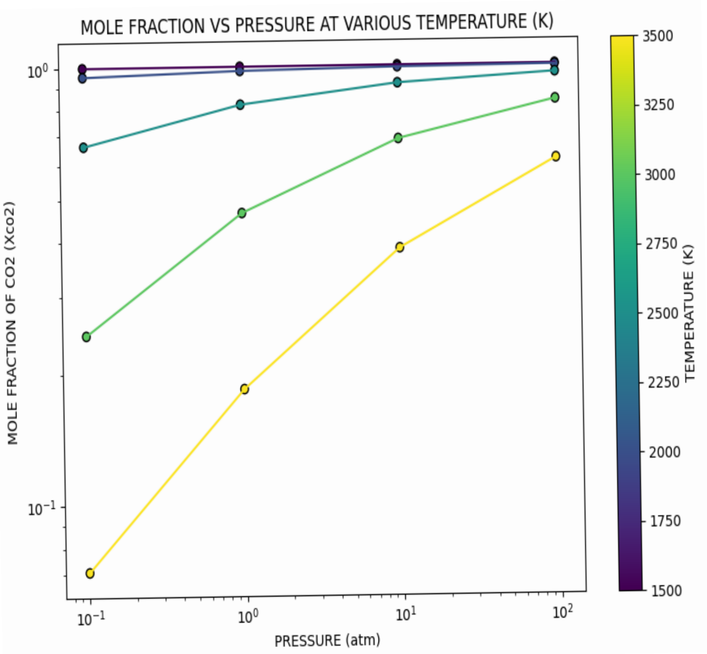
for CO

1500.0	0.1	0.0007664963
1500.0	1.0	0.0003559223
1500.0	10.0	0.0001652360
1500.0	100.0	0.0000767026
2000.0	0.1	0.0313576693
2000.0	1.0	0.0148066452
2000.0	10.0	0.0069279026
2000.0	100.0	0.0032276599
2500.0	0.1	0.2259402493
2500.0	1.0	0.1209305462
2500.0	10.0	0.0602202435
2500.0	100.0	0.0289058732
3000.0	0.1	0.5037077013
3000.0	1.0	0.3579550063
3000.0	10.0	0.2143367446
3000.0	100.0	0.1137380543
3500.0	0.1	0.6196996177
3500.0	1.0	0.5443809338
3500.0	10.0	0.4120319022
3500.0	100.0	0.2609122835



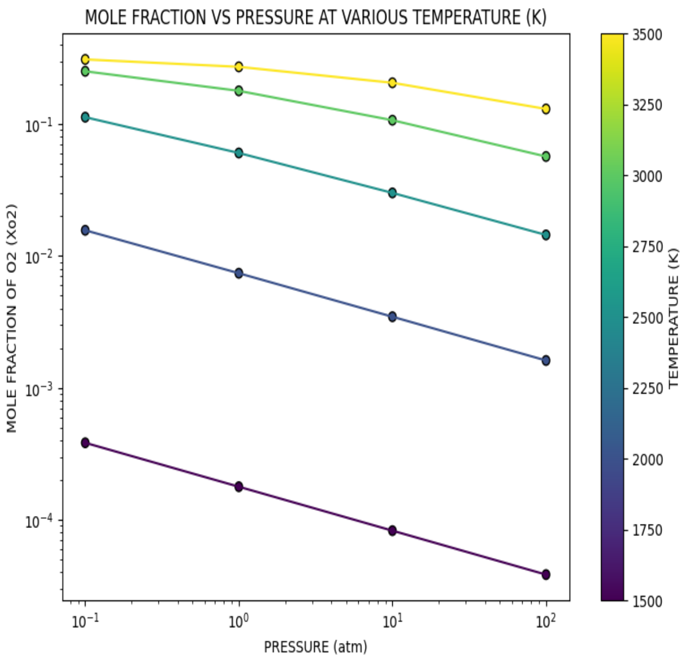
for CO2

1500.0	0.1	0.9988502556
1500.0	1.0	0.9994661166
1500.0	10.0	0.9997521460
1500.0	100.0	0.9998849462
2000.0	0.1	0.9529634960
2000.0	1.0	0.9777900322
2000.0	10.0	0.9896081461
2000.0	100.0	0.9951585101
2500.0	0.1	0.6610896260
2500.0	1.0	0.8186041807
2500.0	10.0	0.9096696347
2500.0	100.0	0.9566411902
3000.0	0.1	0.2444384481
3000.0	1.0	0.4630674905
3000.0	10.0	0.6784948831
3000.0	100.0	0.8293929186
3500.0	0.1	0.0704505735
3500.0	1.0	0.1834285993
3500.0	10.0	0.3819521467
3500.0	100.0	0.6086315748



for O2

1500.0	0.1	0.0003832481
1500.0	1.0	0.0001779611
1500.0	10.0	0.0000826180
1500.0	100.0	0.0000383513
2000.0	0.1	0.0156788347
2000.0	1.0	0.0074033226
2000.0	10.0	0.0034639513
2000.0	100.0	0.0016138300
2500.0	0.1	0.1129701247
2500.0	1.0	0.0604652731
2500.0	10.0	0.0301101218
2500.0	100.0	0.0144529366
3000.0	0.1	0.2518538506
3000.0	1.0	0.1789775032
3000.0	10.0	0.1071683723
3000.0	100.0	0.0568690271
3500.0	0.1	0.3098498088
3500.0	1.0	0.2721904669
3500.0	10.0	0.2060159511
3500.0	100.0	0.1304561417



THANK YOU