Scilab Textbook Companion for Introduction To Chemical Engineering Thermodynamics by J. M. Smith, H. C. Van Ness And M. M. Abbott[1]

Created by
Akshay Garg
B.Tech (pursuing)
Chemical Engineering
NIT, Trichy
College Teacher
Dr. A. Seshagiri Rao
Cross-Checked by
Santosh Kumar, IIT Bombay

August 10, 2013

[1]Funded by a grant from the National Mission on Education through ICT, http://spoken-tutorial.org/NMEICT-Intro. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website http://scilab.in

Book Description

Title: Introduction To Chemical Engineering Thermodynamics

Author: J. M. Smith, H. C. Van Ness And M. M. Abbott

Publisher: McGraw - Hill Companies Inc., New York

Edition: 6

Year: 2001

ISBN: 0-07-049486-X

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes

List of Figures

Chapter 1

Introduction

Scilab code Exa 1.1 Find the Astronaut Mass and Weight on moon

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 1.1
11 //Caption : Program To find the Astronaut's Mass and
       Weight on moon.
12
13 //Given values
14
15 F=730;//Force(N)
16 g_texas=9.792;//Acceleration of gravity in Houston,
     Texas(m/s^2).
17 g_moon=1.67;//Acceleration of gravity at moon(m/s^2)
```

```
18
19 //Solution
20
21 m=approx(F/g_texas,2);//Mass of Astronaut(Kg)
22 F_moon=approx(m*g_moon,2);//Force on Moon(N)
23 disp('Kg',m,'Mass of Astronaut');
24 disp('N',F_moon,'Force on Moon');
25
26 //End
```

Scilab code Exa 1.2 Find Gauge Pressure and absolute Pressure

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 1.2
11 //Caption : Program To Find Gauge Pressure and
     absolute Pressure
12
13 //Given values
14
15 d=0.01;//Diameter(m)
16 m=6.14;//Mass(Kg)
17 g=9.82;//Acceleration of gravity
18 Pb=748;//Barometric Pressure(Torr)
19
```

```
20 //Solution
21
22 F=approx(m*g,3);//Force(N)
23 A=(%pi/4)*d*d;//Area(m^2)
24 Pg=approx(F/A,-2);//Gauge Pressure(N/m^2)
25 Pa=approx(Pg+(Pb*0.013332*(10^4)),-2);//Absolute
     Pressure(Pa)
26 disp('N',F,'Force ');
27 disp('(X 10^4) N/m^2',Pg/10^4,'Gauge Pressure ');
28 disp('KPa',Pa/1000,'Absolute Pressure');
29
30 //End
```

Scilab code Exa 1.3 Find Pressure in a Manometer

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 1.3
11 //Caption : Program To Find Pressure in a Manometer
12
13 //Given Values
14
15 T=300.15;//Temp=300.15K(27'C)
16 h=60.5*(10^-2);//Height=60.5cm
17 rho=13530;//Density(Kg/m^3)
18 g=9.784;//Acceleration of gravity(m/s^2)
```

```
19
20 //Solution
21
22 P=approx(h*rho*g,0);
23 disp('KPa',P/1000,'Pressure in KPa');
24 disp('bar',P/100000,'Pressure in bar');
25
26 //End
```

Scilab code Exa 1.4 Find the velocity and Energy

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 1.4
11 //Caption : Program to find the velocity and Energy
12
13 //Given values
14
15 M=2500;//Mass=2500Kg
16 h1=10;//height1=10m
17 h2=100;//height2=100m
18 g=9.8;//Acceleration of gravity(m/s^2)
19
20 //Solution
21 //(a)
22 PE1=M*h1*g;//(J)
```

```
23 disp('J',PE1,'(a)Potential energy of the elevator in
       its Initial Position')
24
25 //(b)
26 W=M*g*integrate('1','l',h1,h2);//(J)
27 disp('J',W,'(b)Work Done in Raising the Elevator')
28
29 //(c)
30 PE2=M*g*h2;//(J)
31 disp('J',PE2,'(c)Potential energy of the elevator in
       its Highest Position')
32
33 //(d)
34 KE2=0;
35 PE3=0;
36 KE3=PE2;//(J)    //Conservation Of Mechanical Energy
37 u=approx((2*KE3/M)^(1/2),2);//(m/s)
38 disp('m/s',u,'(d)Velocity of the Elevator')
39 disp('J',KE3,'(d)Kinetic Energy of the Elevator')
40
41 //(e)
42 PE_Spring=KE3;//(J)
43 disp('J',PE_Spring,'(e)Potential energy of
     compressed spring ')
44
45 //(f)
46 TE=PE1+W;
47 disp('J',TE,'(f)Total Energy of the System')
48
49 //End
```

Chapter 2

The First Law And Other Basic Concepts

Scilab code Exa 2.1 Find Energy in a Waterfall

```
1  clear all;
2  clc;
3
4  //Example 2.1
5  //Caption : Program to find Energy in a Waterfall
6
7
8  //Given values
9  H=100;//height=100m
10 M=1;//Mass of water=1Kg
11 g=9.8066;//Acceleration due to gravity(m/s^2)
12
13 //Solution
14
15 //Del(Energy of the system)=0
16 //hence, del(U)+del(KE)+del(PE)=0
17
```

```
18 //(a)
19 PE1=M*H*g;//(J)
20 disp('J',PE1,'(a)Potential energy of Water at the
      Top');
21
22 //(b)
23 del_U=0;
24 KE1=0;
25 PE2=0;
26 KE2=PE1;//(J)
27 disp('J',KE2,'(b)Kinetic energy of Water');
28
29 //(c)
30 del_U=KE2;
31 disp('J',del_U,'(c)Change in Internal energy when 1
      kg Water added');
32
33 //End
```

Scilab code Exa 2.3 Find the energy change in a System

```
1 clear all;
2 clc;
3
4 //Example 2.3
5 //Caption : Program to find the energy change in a
      System
6
7 //Given values
8
9 P_atm=101.3;//Atm Pressure=101.3KPa
10 V1=0.1;//Volume1=0.1m^3
11 V2=0.2;//Volume2=0.2m^3
```

```
12
13 //Solution
14
15 del_V=V2-V1;
16 W_by=P_atm*del_V;
17 W_on=-W_by;
18 Q=0;
19 del_Energy=Q+W_on;//KJ
20 disp('KJ',del_Energy,'Energy Change')
21
22 //End
```

Scilab code Exa 2.4 Find the Heat flow in the Path

```
1 clear all;
2 clc;
3
4 //Example 2.4
5 //Caption : Program to find the Heat flow in the
     Path
6
7 //Given values
8 W_acb=40;//J
9 Q_acb=100;//J
10 W_aeb=20;//J
11 W_bda=30;//J
12
13 //Solution
14
15 del_U_ab=Q_acb-W_acb;
16
17 //(a)
18 Q_aeb=del_U_ab-W_aeb;//J
```

```
19 disp('J',Q_aeb,'(a)Heat Flow in acb')
20
21 //(b)
22 del_U_ba=-del_U_ab;//J
23 Q_bda=del_U_ba-W_bda;
24 disp('J',Q_bda,'(b)Heat Flow in bda')
25
26 //End
```

Scilab code Exa 2.5 Find The degree of freedom for the various systems

```
1 clear all;
2 clc;
3
4 //Example 2.5
5 //Caption : Program To Find The degree of freedom
     for the various systems
6
7 //Formula To be Used  F=2-#+N (Where,#(pi)-no of
     phases,N-number of chemical species)
8
9 //(a)-Liquid Water in equllibrium with its vapour.
10 N=1;
11 pi=2;
12 F=2-pi+N;
13 disp(F,'(a)Degree Of freedom is');
14
15 //(b)-Liquid Water in equllibrium with a mixture of
     vapour and nitrogen.
16 N=2;
17 pi=2;
18 F=2-pi+N;
19 disp(F,'(b)Degree Of freedom is');
```

```
20
21 //(c)-A liquid Soln of alcohol in water in
      equillibrium with its vapour
22 N=2;
23 pi=2;
24 F=2-pi+N;
25 disp(F,'(c)Degree Of freedom is');
26
27 //End
```

Scilab code Exa 2.6 Find the work done by gas

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 2.6
11 //Caption : Program to find the work done by gas
12
13 //Given values
14 P=14;//Pressure=14bar
15 V1=0.03;//Initial volume=0.03m^3
16 V2=0.06;//Final Volume
17 //Process is isothermal
18 //(a)-To find the work done by gas in moving the
      External force
19 //(b)-To find the work done by gas if external force
       is suddenly reduced to half its initial value
```

```
20
21 //Solution
22 //(a)
23 K=P*V1*(10^5);//J
24 W1=approx(-K*integrate('1/V','V',0.03,0.06),0);//J
25 P2=K/V2;//Final Pressure(Pa)
26 P2=P2/(10^5);//bar
27 disp('J',W1,'(a)The work done by gas in moving the
     External Force is')
28
29 //(b)
30 W2=-P2*(10^5)*integrate('1','V',0.03,0.06)
31 n=approx((W2/W1)*100,1);//Efficiency
32 disp('J',W2,'(b)The work done by gas if external
     force is reduced to half is')
33 disp('%',n,'Hence the efficiency is')
34
35 //End
```

Scilab code Exa 2.7 Find the Enegy Changes in the Process

```
1 clear all;
2 clc;
3
4 //Example 2.7
5 //Caption : Program to find the Enegy Changes in the
      Process
6
7 //Given values
8 P=7;//pressure=7bar
9 m=45;//Mass of cube
10 mt=23;//mass of piston,piston rod,pan
11 x=0.5;//Distance moved=0.5m
```

```
12 g=9.8;//Acceleration Due to gravity(m/s^2)
13
14 //Solution
15
16 //Acc to Eqn del_U_sys+del_U_surr+del_PE_surr=0
17 del_PE_surr=(m+mt)*g*x;
18 //ans=del_U_sys+del_U_surr
19 disp('J',-del_PE_surr,'Energy Changes in the Process
     ')
20
21 //End
```

Scilab code Exa 2.8 Find Change in Enthalpy and Internal Energy

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 2.8
11 //Caption : Program to find Change in Enthalpy and
     Internal Energy
12
13 //Given values
14 m=1;//1kg of water
15 T=373.15;//Temp=373.15K(100'C)
16 P=101.325;//Pressure=101.325KPa
17 V2=1.673;//Final Volume[m^3]
18 V1=0.00104;//Initial Volume[m^3]
```

```
Sv_liqiud=0.00104;//Specific Volume of Liqiud
Sv_vapour=1.673;//Specific Volume of Vapour
del_H=2256.9;//Heat Added(KJ)

//Solution
Q=del_H;
del_V=V2-V1;
W=P*del_V;//KJ
del_U=approx(del_H-(P*del_V),1);
disp('KJ',del_H,'Change in Enthalpy');
disp('KJ',del_U,'Change in Internal energy');
//End
```

Scilab code Exa 2.9 Find Work Heat del U and del H

```
clear all;
clc;

//Example 2.9
//Caption : Program To Find Work,Heat,del U and del
   H

//Given values
//Initial
P1=1;//Pressure=1bar
T1=298.15;//Temp=298.15K(25'C)
V1=0.02479;//Molar Volume=0.02479m^3/mol
//Final
P2=5;//Pressure=5bar
Cv=20.78;//J/mol/K
Cp=29.10;//J/mol/K

//to Find del_U,del_H by two processes
```

```
18 V2=V1*(P1/P2);//m^3(1 mol)
19 disp('m^3',V2,'Final Volume')
20
21 //Solution
22
23 //(a)-Cooling at const pressure followed by heating
     at const Volume
24 T2=T1*(V2/V1);//K
25 disp('K',T2,'Final Temperature')
26 del_H=round(Cp*(T2-T1));//J
27 Q1=del_H;//J
28 del_U1=round(del_H-(P1*(10^5)*(V2-V1)));//J
29 //Second Step
30 del_U2=round(Cv*(T1-T2));//J
31 Q2=del_U2;
32 Q=Q1+Q2;
33 del_U=0;
34 W=del_U-Q;//J
35 del_H=0;//const Temperature
36
37 disp('(a) Cooling at const Pressure Followed by
     Heating at const Volume')
38 disp('J',Q,'Heat Required')
39 disp('J',W,'Work Required')
40 disp('J',del_H,'Change in enthalpy')
41 disp('J',del_U,'Change in Energy')
42 //(b)-heating at Const Volume Followed by cooling at
     const Pressure
43 T2=T1*(P2/P1);//K
44 del_U1=round(Cv*(T2-T1));//J
45 Q1=del_U1;
46 del_H=round(Cp*(T1-T2));//J
47 Q2=del_H;
48 del_U2=round(del_H-(P2*(10^5)*(V2-V1)));//J
49 Q=Q1+Q2;
50 del_U=0;
51 W=del_U-Q;//J
52 del_H=0;//const Temperature
```

```
53 disp('(b) Heating at const Volume Followed by
      Cooling at const Pressure')
54 disp('J',Q,'Heat Required')
55 disp('J',W,'Work Required')
56 disp('J',del_H,'Change in enthalpy')
57 disp('J',del_U,'Change in Energy')
58
59
60 //Note
61 disp('Note : The Answer varies From That in the book
      because in Book 4956.44 has been rounded to 4958
      which is absurd')
62 //End
```

Scilab code Exa 2.10 Find change in Internal Energy and Enthalpy

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 2.10
11 //Caption : Program to Find change in Internal
      Energy and Enthalpy
12
13 //Given values
14
15 //Initial values
16 T1=277;//Temp=277K
```

```
17 P1=10;//Pressure=10bar
18 V1=2.28;//molar Volume=2.28m^3/Kmol
19
20 //Final value
21 T2=333;//Temp=333K
22 P2=1;//Pressure=1atm
23
24 Cv=21;//KJ/Kmol/K
25 Cp=29.3;//KJ/Kmol/K
26
27 //Solution
28 //(a)-Cooled at const Vol to the final pressure
29 //(b)-Heated at const Pressure to final temperature
30 T_=T1*(1/10);//Intermediate temperature
31 del_Ta=T_-T1;
32 del_Tb=T2-T_;
33 del_Ua=Cv*del_Ta;//KJ/Kmol
34 del_Ha=del_Ua+(V1*(P2-P1)*(10^5)/(10^3));//KJ/Kmol
35 V2=(V1*P1*T2)/(P2*T1);//m^3/kmol
36 del_Hb=Cp*del_Tb;
37 del_Ub=del_Hb-(P2*(V2-V1)*(10^5)/(10^3));//KJ/Kmol
38
39 del_U=approx(del_Ua+del_Ub,0);
40 del_H=approx(del_Ha+del_Hb,0);
41 disp('KJ/Kmol',del_U,'Change In Internal Energy')
42 disp('KJ/Kmol',del_H,'Change In Enthalpy')
43
44 //End
```

Scilab code Exa 2.13 Find the time for a certain Temperature Drop

```
1 clear all;
2 clc;
```

```
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 2.13
11 //Caption : Program to Find the time for a certain
     Temperature Drop
12
13 //Given values
14 M=190;//Mass=190Kg
15 T0=333.15;//Temperature=333.15K(60'C)
16 m=0.2;//Steady rate of mass(Kg/s)
17 T=308.15;//Temperature=308.15K(35'C)
18 T1=283.15;//Temperature=283.15K(10'C)
19
20 //Solution
21 //Using the Eqn (2.29)
22 t=approx(-(M/m)*log((T-T1)/(T0-T1)),1);//s
23 disp('s',t,'Time Taken for temperature of water to
     drop from 333.15K to 308.15K')
24 t=round(t/60);//min
25 disp('min',t,'Time Taken for temperature of water to
      drop from 333.15K to 308.15K')
26
27 //End
```

Scilab code Exa 2.14 Find the Enthalpy of Steam

```
1 clear all;
2 clc;
```

```
3
4 //Example 2.14
5 //Caption : Program to find the Enthalpy of Steam
6
7 //Given values
8
9 rQ=4.15;//[g/s]  flow rate
10 rQ2=12740;//Rate of Heat addition from resistance
     heater
11
12 //Solution
13 //del_z and del_u*2 are negligible  if Ws and H1=0..
     then H2=Q
14 H2=round(rQ2/rQ);//[J/g]
15 disp('J/g',H2,'Enthalpy of Steam')
16
17 //End
```

Scilab code Exa 2.15 Find the Heat to be Removed during Compression

```
1 clear all;
2 clc;
3
4 //Example 2.15
5 //Caption : Program To find the Heat to be Removed
     during Compression
6
7 //Given values
8
9 V=600;//[m/s]
10 W_compression=240;//[KJ/Kg]
11
12 //Solution
```

```
13 //Using Eqn(2.32a)
14 Q=(1/2*(V*V)/1000)-W_compression;
15
16 disp('KJ/kg',-Q,'Thus Heat Removed from each KG of
      air compressed is')
17
18 //End
```

Scilab code Exa 2.16 Find the Heat to be Removed during Compression

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 2.15
11 //Caption : Program to Find the Temperature in the
      second Tank
12
13 //Given values
14
15 R=3.15*10^-3;//[m^3/s]  Rate of pumping
16 rH=-700;//[KW]  Rate of Heat lost
17 h=15;//[m]  Height
18 rW=1.5;//[KW]
19 rho=958;//[Kg/m^3]  at 366.65K
20 g=9.805;
21 gc=1000;
22 del_z=h;
```

```
23
24 //Solution
25
26 rm=approx(R*rho,3);//[Kg/s]  Mass flow rate
27 Q=approx(rH/rm,1);//[KJ/Kg]
28 W=approx(rW/rm,3);//[KJ/Kg]  Shaft Work
29 K=approx(g/gc*del_z,3);
30
31 //using Eqn(2.32b)
32 del_H=Q+W-K;
33
34 //From Steam tables for water at 366.65K
35 H1=391.6;//[KJ/Kg]
36 H2=del_H+H1;
37 disp('KJ/Kg',H2,'Enthalpy')
38 //From Steam Tables temp at this enthalpy is
39 T=311.35;//[K]
40 disp('K',T,'Temperature in the Second tank')
41
42 //End
```

Chapter 3

Volumetric Properties Of Pure Fluids

Scilab code Exa 3.1 Find Volume Change and Pressure generated

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 3.1
11 //Caption : Program to Find Volume Change and
       Pressure generated for Acetone
12
13 //Given Values for Acetone
14 P1=1;//Pressure=1Bar
15 T1=20;//Temp=293.15K(20'C)
16 Beta=1.487*10^(-3);//vol expansivity(K^-1)
```

```
k=62*10^(-6);//isothermal compressibility(bar^-1)
V1=1.287*10^(-3);//Volume(m^3 kg^-1)

//Solution
//(a)
//Find (dP/dT)v??
//Using eq.(3.4),V constant hence dV=0
ans_a=round(Beta/k);
disp('K^-1',ans_a,'(a)The value of (dp/dT)v is ')

//(b)
//Find Pressure when acetone heated at const. Vol
   from T1(1bar) to T2.
T2_b=30;//Temp2=303.15K(30'C)
del_P=ans_a*(T2_b-T1);
ans_b=P1+del_P;
disp('bar',ans_b,'(b)The pressure is ')

//(c)
//Find vol. change when acetone changed from T1(P1)
   to T2(P2)
T2_c=0;//Temp2=273.15K(0'C)
P2=10;//pressure=10bar
//solve using Eq. (3.5)
ln_value=(Beta*(T2_c-T1))-(k*(P2-P1));//ln(V2/V1)
ratio=exp(ln_value);//taking antilog,V2/V1
V2=ratio*V1;
del_V=approx(V2-V1,6)
disp('(X 10^-3) m^3 kg^-1',del_V*1000,'(c)The change
    in Volume is ')

//End
```

Scilab code Exa 3.2 Find Work Heat del H del H

```
clear all;
clc;

//Example 3.2
//Caption : Program to Find Work,Heat,del H,del H

//Given Values for the Gas

//Figure
P=[1 5];
V=[25 25];
plot2d(V,P,style=1)

V=5:0.5:25;
P=25*V^-1;
plot2d(V,P,style=2)
P=P^1.4;
plot2d(V,P,style=3)
P=[5 9.52];
V=[5 5];
plot2d(V,P,style=3,rect=[0,0,30,10])
xtitle("Diagram for Ex.3.2","V x 10^3(m^3)","P(bar)"
   )
legend("(a)","(b)","(c)")
P=[5 5];
V=[5 25];
plot2d(V,P,style=1)

clear all;
//Initial Stage
P1=1;//Pressure=1bar
T1=298.15;//Temp1=298.15K(25'C)

//Final Stage
P2=5;//Pressure=1bar
//Temp same as Temp1(Isothermal)
```

```
36
37 R=8.314;//J/Mol/K
38 Cv=(5/2)*R;//J/Mol/K
39 Cp=(7/2)*R;//J/Mol/K
40
41 //(a)
42 //Const Vol follwd by const Pressure
43 T2=T1*(P2/P1);
44 //By Eq 2.23
45 del_T=T2-T1;
46 Q1=Cv*(T2-T1);//Heat at const Vol
47 Q2=Cp*(T1-T2);//Heat at const pressure
48
49 Q_a=round(Q1+Q2);
50 W_a=-Q_a;//W=del_U-Q, here del_U=0
51 disp('       (a) Heating at constant volume Followed
      by cooling at constant Pressure')
52 disp('J',W_a,'work done by heating at const vol
     followed by const Pressure ')
53
54 disp('J',Q_a,'Heat Transferred Q ')
55
56 disp('change in Internal Energy and enthalpy = 0')
57
58 //(b)
59 //Isothermal Compression
60 //By Eq. (3.26)
61 Q_b=round(R*T1*log(P1/P2));
62 W_b=-Q_b;
63 disp('       (b) Isothermal compression')
64
65 disp('J',W_b,'work done by Isothermal compression ')
66 disp('J',Q_b,'Heat Transferred Q')
67 disp('change in Internal Energy and enthalpy = 0')
68
69 //(c)
70 //Adiabatic compression
71 gama=Cp/Cv;
```

```
V1=(R*T1)/(P1^(10^5));
V2=V1*(P1/P2);
T2_c=T1*((V1/V2)^(gama-1));//Kelvin(K)
P2_c=P1*((V1/V2)^gama);//bar
//Using Eq. (3.31)
W_c=round(Cv*(T2_c-T1));//W=Cv*del_T(Joules)
Q_c=-W_c;

disp('       (c) Adiabatic compression followed by
   cooling at constant Volume')

disp('J',W_c,'work done by Adiabatic compression
   Followed by Cooling at const Vol ')

disp('J',Q_c,'Heat Transferred Q')
disp('change in Internal Energy and enthalpy = 0')

//End
```

Scilab code Exa 3.3 Find W Q del U and del H for the Figure

```
clear all;
clc;

//Example 3.3
//Caption : Program to Find W,Q, del U and del H for
   the Figure

//Figure
V=2083:0.5:2853;
P=2853*V^-1;
```

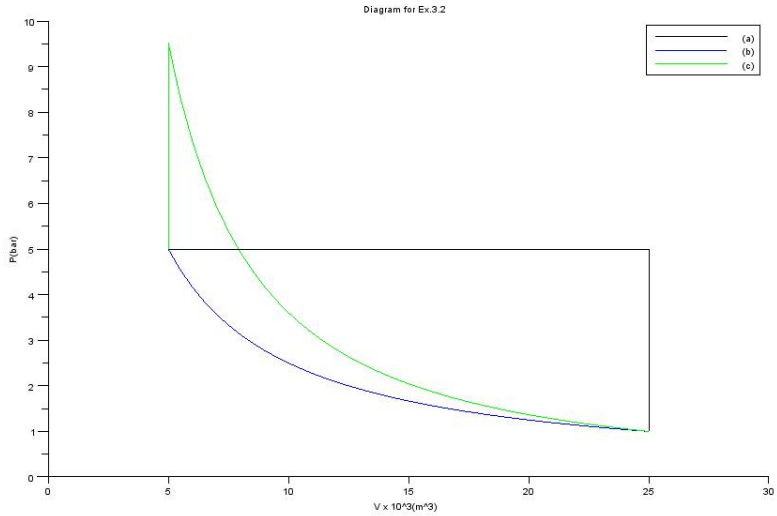

Figure 3.1: Find Work Heat del H del H

```
10 P=P^1.67;
11 plot2d(V,P,style=1)
12 P=[1.698 1.698];
13 V=[1690 2083];
14 plot2d(V,P,style=2)
15 V=1690:0.5:2853;
16 P=2853*V^-1;
17 plot2d(V,P,style=3,rect=[1500,0.8,3000,2])
18 xtitle("Diagram for Ex.3.3","V","P")
19 legend("(a)","(b)","(c)")
20 clear all;
21
22
23 //Given Values for the Ideal Gas
24 R=8.314;//J/Mol/K
25 Cv=(3/2)*R;//J/Mol/K
26 Cp=(5/2)*R;//J/Mol/K
27 gama=Cp/Cv;
28
```

```
29 //Solution
30
31 //(a)
32 //Adiabatic Compression
33 P1=1;//Pressure=1bar
34 T1=343.15;//Temp1=343.15K(70'C)
35 T2=423.15;//Temp2=423.15K(150'C)
36 Q_a=0;//Adiabatic Compression
37 del_U_a=round(Cv*(T2-T1));
38 W_a=del_U_a;
39 del_H_a=round(Cp*(T2-T1));
40 //Using Eq. (3.29b)
41 P2=P1*((T2/T1)^(gama/(gama-1)));//bar
42
43 //(b)
44 //cooled form 150'C to 70'C at Const pressure
45 //Using Eq.(3.27)
46 Q_b=round(Cp*(T1-T2));
47 del_H_b=Q_b;
48 //for Ideal Gas
49 del_U_b=round(Cv*(T1-T2));
50 //by First law
51 W_b=del_U_b-Q_b;//Joules
52
53 //(c)
54 //Expanded Isothermally to original state
55 del_U_c=0;//isothermal
56 del_H_c=0;//isothermal
57 Q_c=round(R*T1*log(P2/P1));
58 W_c=-Q_c;
59
60 //Entire process
61 Qt = Q_a+Q_b+Q_c;
62 Wt = W_a+W_b+W_c;
63 del_Ut=del_U_a+del_U_b+del_U_c;
64 del_Ht=del_H_a+del_H_b+del_H_c;
65
66
```

```
67 //PartII(Irreversible)
68 eta=.80;//Efficiency=80%
69
70 //(a)
71 Wm_a=round(W_a/eta);
72 Qm_a=del_U_a-Wm_a;//del_U remains same (by First Law
   )
73
74 //(b)
75 Wm_b=round(W_b/eta);
76 Qm_b=del_U_b-Wm_b;//del_U remains same (by First Law
   )
77
78 //(c)
79 Wm_c=round(W_c*eta);
80 Qm_c=del_U_c-Wm_c;//del_U remains same (by First Law
   )
81
82 //Entire Process
83 Qmt = Qm_a+Qm_b+Qm_c;
84 Wmt = Wm_a+Wm_b+Wm_c;
85
86
87 del_U_rev=[del_U_a,del_U_b,del_U_c];
88 del_H_rev=[del_H_a,del_H_b,del_H_c];
89 Qrev=[Q_a,Q_b,Q_c];
90 Wrev=[W_a,W_b,W_c];
91 Sumr=[del_Ut,del_Ht,Qt,Wt];
92
93 del_U_irev=del_U_rev;
94 del_H_irev=del_H_rev;
95 Qirev=[Qm_a,Qm_b,Qm_c];
96 Wirev=[Wm_a,Wm_b,Wm_c];
97 Sumi=[del_Ut,del_Ht,Qmt,Wmt];
98
99
100 disp('      (a)Adiabatic Compression')
101 disp('      (b)Cooled form 150'C to 70'C at Const
```

```
    pressure')
disp('     (c)Expanded Isothermally to original
    state')

disp('   Mechanically reversible');

Ans_rev=[del_U_rev',del_H_rev',Qrev',Wrev'];

disp(Sumr,'Sum',Ans_rev,'   del U   del H     Q
          W')

disp('         Irreversible');

Ans_irev=[del_U_irev',del_H_irev',Qirev',Wirev'];

disp(Sumi,'Sum',Ans_irev,'   del U   del H     Q
          W')

//End
```

Scilab code Exa 3.4 Find Q W del U and del H in a PV Diagram

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

```

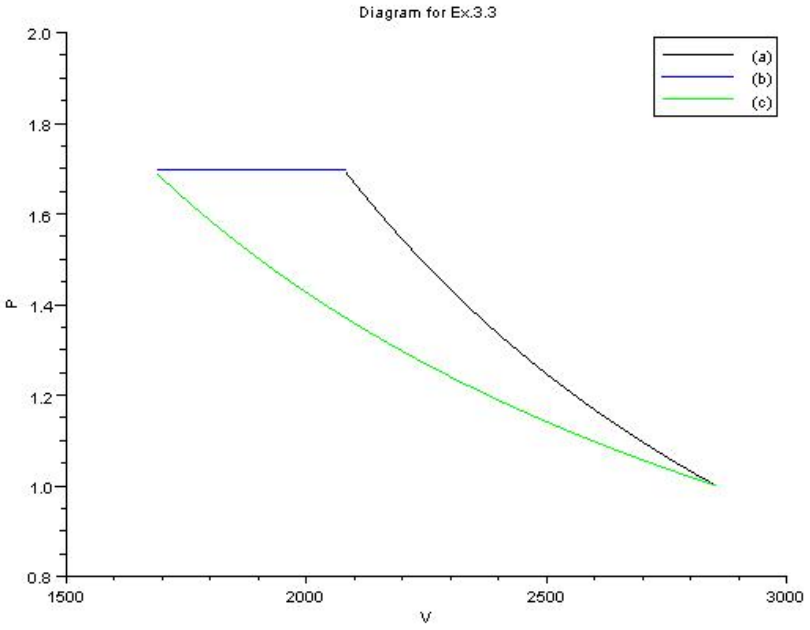

Figure 3.2: Find W Q del U and del H for the Figure

```
10 //Example 3.4
11 //Caption : Program To find Q,W,del_U and del_H in a
       PV Diagram
12
13 //Figure
14 P=[1.35 1.35];
15 V=[0.24 0.264];
16 plot2d(V,P,style=2)
17 V=0.12:0.001:0.24;
18 P=0.324*V^-1;
19 plot2d(V,P,style=5)
20 P=[2.7 2.97];
21 V=[0.12 0.12];
22 plot2d(V,P,style=9,rect=[0.1,0.5,0.3,3])
23 xtitle("Diagram for Ex.3.4","V(m^3)","P(bar)")
24 legend("(a)","(b)","(c)")
25 P=[2.97 2.97];
26 V=[0 0.12];
27 plot2d(V,P,style=4)
28 P=[2.7 2.7];
29 V=[0 0.12];
30 plot2d(V,P,style=4)
31 P=[1.35 1.35];
32 V=[0 0.24];
33 plot2d(V,P,style=4)
34 P=[0 2.7];
35 V=[0.12 0.12];
36 plot2d(V,P,style=4)
37 P=[0 1.35];
38 V=[0.24 0.24];
39 plot2d(V,P,style=4)
40 P=[0 1.35];
41 V=[0.264 0.264];
42 plot2d(V,P,style=4)
43
44 clear all;
45
46 //Given Values for Nitrogen Gas
```

```
m=0.4;//Kg
M=28;//Molecular Mass Of Nitrogen
T1=300.15;//Temp=300.15K(27'C)
Pn=0.35;//Pressure of nitrogen = 0.35bar
Pa=1;//Atm Pressure = 1bar
R=8.314;//J/Mol/K
Cv=(5/2)*R;//J/Mol/K
Cp=(7/2)*R;//J/Mol/K
gama=Cp/Cv;

n=(m/M)*1000;//moles

//Solution

//(a)
//Immersed In ice/water bath
T2=273.15;//Temp=273.15K(0'C)
W_a=-round(n*R*(T2-T1));//Joules
del_H_a=approx(Cp*(T2-T1),0);
Q_a=round(n*del_H_a);
del_U_a=approx((Q_a+W_a)/n,0);
disp('(a)Immersed In ice/water bath')
disp('J',W_a,'work done ')
disp('J',Q_a,'Heat Transferred Q = ')
disp('J',del_U_a,'change in Internal Energy ')
disp('J',del_H_a,'change in enthalpy ')


//(b)
//Isothermal Compression
del_U_b=0;//Isothermal
del_H_b=0;//Isothermal
W_b=-round(n*R*T2*log(1/2));//W=nRTln(V3/V2),here V3
   /V2=0.5(Given)
Q_b=-W_b;
disp('(b)Isothermal Compression')
disp('J',W_b,'work done by Isothermal Compression ')
disp('J',Q_b,'Heat Transferred Q = ')
```

```
84 disp('J',del_U_b,'change in Internal Energy ')
85 disp('J',del_H_b,'change in enthalpy ')
86
87
88 //(c)
89 //constant Vol Process
90 W_c=0;//const Vol
91 del_H_c=approx((Cp*(T1-T2))/n,0);
92 del_U_c=approx(Cv*(T1-T2),0);
93 Q_c=round(n*del_U_c);
94 disp('(c)Constant Vol Process')
95 disp('J',W_c,'work done by Const Vol Process ')
96 disp('J',Q_c,'Heat Transferred Q = ')
97 disp('J',del_U_c,'change in Internal Energy ')
98 disp('J',del_H_c,'change in enthalpy ')
99
100
101 //End
```

Scilab code Exa 3.6 Find Change in KE and Temperature

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10
```

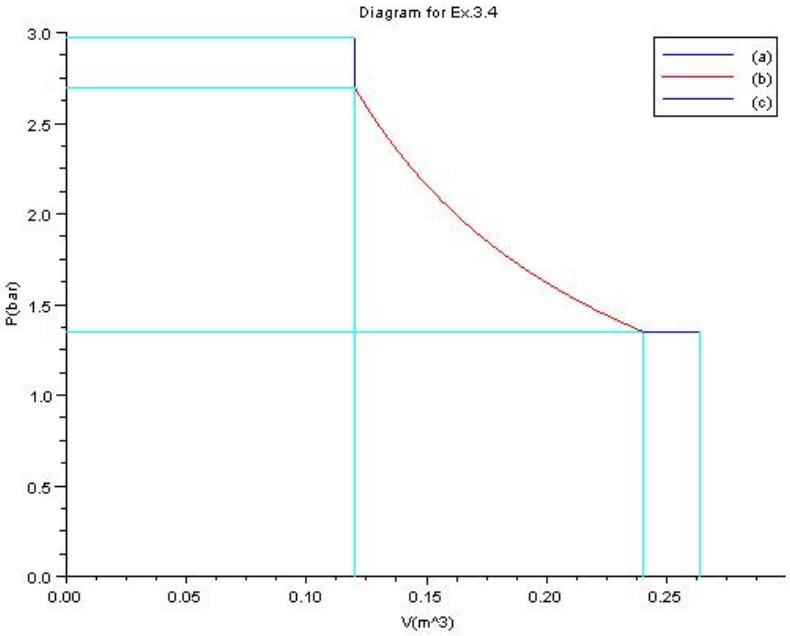

Figure 3.3: Find Q W del U and del H in a PV Diagram

```
11 //Example 3.6
12 //Caption : Program To find (a)Kinetic energy Change
       (b)change in temperature
13
14 //Given values
15 n=1;//Molar Rate(mol/s)
16 D=5;//inner Diameter(cm)
17 R=83.14;
18 Cp=(7/2)*R;
19 M=29*10^-3;//Molar mass(g/mol)
20 T=293.15;//temperature=293.15K(20'C)
21 P1=6;//Upstream Pressure
22 P2=3;//Downstream Pressure
23
24 //Solution
25 //from Eq.(2.24b)
26 A=(%pi/4)*((D*10^-2)^2);//Area(m^2)
27 //upstream molar Volume
28 V1=(R*T/P1)*10^-6;//m^3/mol
29 u1=n*V1/A;//velocity(m/s)
30 V2=2*V1;
31 u2=2*u1;
32 del_KE=approx(n*M*((u2^2)-(u1^2))/2,3);//J/s(W)
33 del_T=approx(-del_KE/(Cp*0.1),4);//K
34 disp('W or J/s',del_KE,'Change in KE ')
35 disp('K',del_T,'Change in Temperature ')
36
37 //End
```

Scilab code Exa 3.7 Find V and Z for isopropyl vapor

```
1 clear all;
2 clc;
```

```
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 3.7
11 //Caption : Program to Find V and Z for isopropyl
       vapor
12
13 //Given Values
14 T=473.15;//Temp=473.15k(200'C)
15 P=10;//Pressure=10bar
16 B=-0.388;//Viral Coefficient(m^3/Kmol)
17 C=-26*(10^(-3));//Viral Coefficient(m^6/(kmol)^2)
18 //Calculate V and Z for isopropyl vapor
19
20 //Solution
21
22 R=83.14*(10^(-3));//m^3bar/Kmol/K
23
24 //(a)
25 //Ideal Gas equation
26 V_a=approx((R*T)/P,3);
27 Z_a=1;//Ideal Gas
28 disp('(a) By Ideal gas Equation')
29 disp('m^3/kmol',V_a,'V = ')
30 disp(Z_a,'Z = ')
31
32 //(b)
33 //Using Eqution 3.37 -> Z=PV/RT=1+BP/RT
34 V_b=approx((R*T/P)+B,3);
35 Z_b=approx(P*V_b/(R*T),4);
36 disp('(b) Using Eqution 3.37 -> Z=PV/RT=1+BP/RT')
37 disp('m^3/kmol',V_b,'V = ')
38 disp(Z_b,'Z = ')
39
```

```
40 //(c)
41 //Using Equation 3.39 -> Z=PV/RT=1+(B/V)+(C/(V^2))
42 //Iterations
43 a=V_a;//Initial
44 i=-1;
45 while(i==-1)
46    b=((R*T/P)*(1+(B/a)+(C/(a^2))));
47    c=abs(b-a)
48    if(c<=0.0001)
49      i=1;
50      break;
51    end
52    a=b;
53 end
54
55 V_c=approx(b,3);
56 Z_c=approx(P*V_c/(R*T),4);
57 //Ans
58 disp('(c) Using Equation 3.39 -> Z=PV/RT=1+(B/V)+(C
      /(V^2))')
59 disp('m^3/kmol',V_c,'V = ')
60 disp(Z_c,'Z = ')
61
62 //End
```

Scilab code Exa 3.8 Find Molar Volume of nButane

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
```

```
7     funcprot(0)
8   endfunction
9
10  //Example 3.8
11  //Caption : Program to Find Molar Volume of n-Butane
12
13  //Given Values
14  T=350;//Temp=350K(76.85'C)
15  P=9.4573;//Pressure=9.4573bar
16  R=83.14;
17
18  Tc=425.1;//App B
19  Pc=37.96;//App B
20  Tr=T/Tc;
21  Pr=P/Pc;
22  //Parameters for RK
23  si=0.42748;
24  omega=0.08664;
25  epsilon=0;
26  sigma=1;
27  a=Tr^-0.5;
28
29  //Solution
30
31  //Using Eq(3.51)
32  q=si*a/(omega*Tr);
33  Beta=omega*Pr/Tr;
34
35  disp('The Following Results given By Redlich/Kwong
       Equation')
36  //(a)
37  Z=1;//initial
38  a=Z;
39  for i=0:10
40    b=1+Beta-((q*Beta)*(a-Beta)/(a*(a+Beta)));
41     if((b-a)==0.0001)
42       break;
43     end
```

```
44      a=b;
45      i=i+1;
46  end
47  Z=approx(b,4)
48  V=round(Z*R*T/P);
49  disp('cm^3/mol',V,'Molar Volume of saturated Vapor
        is ');
50
51  //(b)
52  Z=Beta;//initial
53  a=Z;
54  for i=0:20
55    b=Beta+(a*(a+Beta)*(1+Beta-a)/(q*Beta));
56      if((b-a)==0.0001)
57        break;
58      end
59      a=b;
60      i=i+1;
61  end
62  Z=approx(b,5)
63  V=approx(Z*R*T/P,1);
64  disp('cm^3/mol',V,'Molar Volume of Saturated Liquid
        is ');
65
66  //Given Values
67  T=350;//Temp=350K(76.85'C)
68  P=9.4573;//Pressure=9.4573bar
69  R=83.14;
70
71  Tc=425.1;//App B
72  Pc=37.96;//App B
73  Tr=T/Tc;
74  Pr=P/Pc;
75  //Parameters for eqns[vdW,RK,SRK,PR]
76  si=[27/64,0.42748,0.42748,0.45724];
77  omega=[1/8,0.08664,0.08664,0.07779];
78  epsilon=[0,0,0,(1-sqrt(2))];
79  sigma=[0,1,1,(1+sqrt(2))];
```

```
80 w=0.2;//App B
81 aSRK=(1+((0.480+(1.574*w)-(0.1768*w^2))*(1-Tr^0.5)))
      ^2;
82 aPR=(1+((0.37464+(1.54226*w)-(0.26992*w^2))*(1-Tr
      ^0.5)))^2;
83 a=[1,Tr^-0.5,aSRK,aPR];
84
85 //Solution
86
87 disp('      By All Equations')
88 //Using Eq(3.51)
89 q=si.*a./(omega.*Tr);
90 Beta=omega.*Pr./Tr;
91
92 //disp('The Following Results given By Redlich/Kwong
      Equation')
93 //(a)
94 for j=1:4
95
96     Z=1;//initial
97     A=Z;
98     for i=0:10
99       b=1+Beta(j)-((q(j)*Beta(j))*(A-Beta(j))/((A+(
           epsilon(j)*Beta(j)))*(A+(sigma(j)*Beta(j)))
           ));
100        if((b-A)==0.0001)
101          break;
102        end
103        A=b;
104        i=i+1;
105    end
106    z(j)=approx(b,4);
107 end
108 V=round(z.*R*T/P);
109 disp('Molar Volume(Vv) of Saturated Vapor');
110 disp(V','    vdW      RK      SRK      PR')
111
112 //(b)
```

```
113 for j=1:4
114     Z=Beta(j);//initial
115     A=Z;
116     for i=0:20
117       b=Beta(j)+((A+(epsilon(j)*Beta(j)))*(A+(sigma(
            j)*Beta(j)))*(1+Beta(j)-A)/(q(j)*Beta(j)));
118        if((b-A)==0.0001)
119          break;
120        end
121        A=b;
122        i=i+1;
123     end
124     z(j)=approx(b,5);
125 end
126 V=approx(z*R*T/P,1);
127 disp('Molar Volume(Vl) of Saturated Liquid');
128 disp(V','    vdW       RK       SRK       PR')
129
130 disp('Note : Exp Value is Vv = 2482 cm^3/mol and Vl
    = 115 cm^3/mol')
131
132 //End
```

Scilab code Exa 3.9 Find Molar Volume of n Butane by Various Eqn

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
```

```
9
10 //Example 3.9
11 //Caption : Program to Find Molar Volume of n-Butane
       by Various Eqn
12 //Given Values
13 T=510;//Temp=510K
14 P=25;//Pressure=25bar
15 R=0.08314;
16
17 //(a)
18 //By the Ideal-gas Equation
19
20 V=approx(R*T/P,4);//m^3/kmol
21 disp('(a)By the Ideal-gas Equation')
22 disp('m^3/kmol',V,'The Molar Volume is ')
23
24 //(b)
25 //The Generalized compressibility-factor Correlation
26 Tc=425.1;//From App.B
27 Pc=37.96;//From App.B
28 Tr=approx(T/Tc,1);
29 Pr=approx(P/Pc,3)
30 //Interpolation in Tables E.1 and E.2 then provides
31 Z0=0.865;
32 Z1=0.038;
33 w=0.200;
34 Z=Z0+(w*Z1);
35 V=approx(Z*R*T/P,2);//m^3/kmol
36 disp('(b)The Generalized compressibility-factor
       Correlation')
37 disp('m^3/kmol',V,'The Molar Volume is ')
38
39 //(c)
40 //The Generalized Virial-coefficient Correlation
41 B0=0.083-(0.422/(Tr^1.6));//Eqn (3.61)
42 B1=0.139-(0.172/(Tr^4.2));//Eqn (3.62)
43 K=approx(B0+(w*B1),3)//K=BPc/RTc By Eqn (3.59)
44 //By Eqn(3.58)
```

```
45 Z=approx(1+(K*Pr/Tr),3)
46 V=approx(Z*R*T/P,4);//m^3/kmol
47 disp('(c)The Generalized Virial-coefficient
      Correlation')
48 disp('m^3/kmol',V,'The Molar Volume is ')
49
50 //End
```

Scilab code Exa 3.10 Find Pressure generated for methane

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 3.10
11 //Caption : Program To Find Pressure generated for
      methane
12
13 //Given Values
14 T=323.15;//Temp=323.15K(50'C)
15 V=0.125;//Volume=0.125m^3
16 R=0.08314;
17
18 //(a)
19 //By Ideal-gas equation,
20 P=approx(R*T/V,1);//in bar
21 disp('(a)By Ideal-gas equation')
22 disp('bar',P,'Pressure is ')
```

```
23
24 //(b)
25 //for Redlich/Kwong equation
26 Tc=190.6;//App B
27 Tr=T/Tc;
28 si=0.42748;
29 omega=0.08664;
30 Pc=45.99;//App B
31 a=approx(si*((Tr^(-0.5))*(R^2)*(Tc^2))/Pc,3)//Eqn
      (3.42)Units of a(T) bar m^6
32 b=approx(omega*R*Tc/Pc,5)//Eqn (3.43)Units of b m^3
33 //Using eqn (3.41)
34 //P=RT/(V-b)-a(T)/(V+Eb)(V+~b),E->epsilon,~->sigma
35 epsilon=0;
36 sigma=1;
37 P=approx(((R*T/(V-b))-(a/((V+(epsilon*b))*(V+(sigma*
      b))))),2);
38 disp('(b)for Redlich/Kwong equation')
39 disp('bar',P,'Pressure is ')
40
41 //(c)
42 //A generalized Correlation
43 Z0=0.887;//from Table E.3 and E.4
44 Z1=0.258;//from Table E.3 and E.4
45 w=0.012;
46 Z=Z0+(w*Z1);
47 P=approx(Z*R*T/V,1);//bar
48 disp('(c)A generalized Correlation')
49 disp('bar',P,'Pressure is ')
50
51 //End
```

Scilab code Exa 3.11 Find Pressure generated for ammonia

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 3.11
//Caption : Program to Find Pressure generated for
   ammonia

//Given Values
T=338.15;//Temp=338.15K(65'C)
Vt=0.03;//Volume=0.03m^3
R=0.08314;
m=0.5;//mass in Kg
M=17.02;//Molecular Mass
V=Vt/(m/M);// n=m/M(moles)

//(a)
//By Ideal-gas equation,
P=approx(R*T/V,2);//in bar
disp('(a)By Ideal-gas equation')
disp('bar',P,'Pressure is ')

//(b)
//A generalized correlation
Tc=405.7;//App B
Tr=T/Tc;
Pc=112.8;//App B
B0=0.083-(0.422/(Tr^1.6));//Eqn (3.61)
B1=0.139-(0.172/(Tr^4.2));//Eqn (3.62)
//Substituting in eq(3.59)
w=0.253;
K=B0+(w*B1);//K=BPc/RTc
B=K*R*Tc/Pc;//m^3 kmol^-1
```

```
38 //solving eq.(3.37)
39 P=approx(R*T/(V-B),2);
40 disp('(b)A generalized Correlation')
41 disp('bar',P,'Pressure is ')
42
43 //End
```

Scilab code Exa 3.12 Find density for ammonia

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 3.12
11 //Caption : Program to Find density for ammonia
12
13 //Given Values
14 T=310;//Temp=310K(36.85'C)
15 M=17.02;
16
17 //Solution
18
19 //(a)
20 //saturated liquid
21 Tc=405.7;//App B
22 Vc=0.07247;//App B
23 Zc=0.242;//App B
24 Vsat=approx(Vc*(Zc^((1-Tr)^0.2857)),5);//m^3kmol^-1
```

```
25 rho=approx(M/Vsat,2);
26 disp('(a)Saturated liquid')
27 disp('m^3/kmol',Vsat,'Volume is ')
28 disp('kmol/m^3',rho,'Density is ')
29
30 //(b)
31 //Liquid at 100bar
32 P=100;//Pressure=100bar
33 Pc=112.8;//App B
34 Pr=P/Pc;
35 rho_r=2.38;//From Graph
36 V=Vc/rho_r;
37 //but this Gives large error
38 rho_r1=2.34;
39 V_new=approx(V*rho_r1/rho_r,5);
40 //In exceptance with Experimental Value
41
42 rho=approx(M/V_new,2);
43 disp('(b)For Liquid at 100bar')
44 disp('m^3/kmol',V_new,'Volume is ')
45 disp('kmol/m^3',rho,'Density is ')
46
47 //End
```

Chapter 4

Heat Effects

Scilab code Exa 4.2 Find Heat Required to Heat Methane gas

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=ICPH(T0,T,A,B,C,D)
11   t=T/T0;
12   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))*(T-T0)
13   funcprot(0);
14 endfunction
15
16
17 //Example 4.2
18 //Caption : Program to find Heat Required to Heat
```

```
    Methane gas
19
20 //Given values for methane
21 R=8.314;
22 T0=533.15;
23 T=873.15;
24 A=1.702;
25 B=9.081*(10^-3);
26 C=-2.164*(10^-6);
27 D=0;
28
29
30 //Solution
31 Q=approx(R*ICPH(T0,T,A,B,C,D),0);
32 disp('J',Q,'Heat Required')
33
34 //End
```

Scilab code Exa 4.3 Find the Final Temperature with Heat Given

```
1 clear all;
2 clc;
3
4 //Example 4.3
5 //Caption : Program To Find the Final Temperature
    with Heat Given
6
7 function[Q]=MCPH(T0,T,A,B,C,D)
8   t=T/T0;
9   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
      /(t*T0*T0)))
10  funcprot(0);
11 endfunction
```

```
12
13
14 //Given values for Ammonia
15 R=8.314;
16 T0=533.15;
17 A=3.578;
18 B=3.020*(10^-3);
19 C=0;
20 D=-0.186*(10^5);
21 Q=422*(10^3);
22 n=11.3;
23 del_H=Q/n;
24
25 //Solution
26 i=-1;
27 a=round(T0);//Initial
28 while (i==-1)
29   b=R*MCPH(T0,a,A,B,C,D);
30   c=b*(a-T0);
31   flag=del_H-c;
32   if(flag<=100) then
33     T=a-1;
34     i=1;
35   else
36     a=a+1;
37     i=-1;
38   end
39 end
40
41
42 disp('K',T,'Temperature Required(Approx)')
43
44 //End
```

Scilab code Exa 4.4 Find the Latent Heat

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 4.4
//Caption : Program to find the Latent Heat

del_H1=2257;//latent Heat of Vapourisation of water
   at 373.15K(100'C)[KJ/Kg]
Tr1=373.15/647.1;
Tr2=573.15/647.1;

del_H2=approx(del_H1*((1-Tr2)/(1-Tr1))^0.38,0);//KJ/
   Kg
disp('KJ/Kg',del_H2,'latent Heat at 573.15K')

disp('Note: The Value as given in steam tables at
   573.15K is 1406 KJ/Kg')

//End
```

Scilab code Exa 4.5 Find the Standard Heat at 298K

```
clc;
clear all;

```

```
4 //Example 4.5
5 //Caption : Program to Find the Standard Heat at
     298.15K
6
7 //4HCL + O2 --> 2H2O + 2Cl2
8 del_H_HCL=-92.307;//KJ Heat Of Formation
9 del_H_H2O=-241.818;//KJ
10
11 //4HCL --> 2H2 + 2Cl2
12 del_H_298_HCL=4*(-1)*del_H_HCL;
13 //2H2 + O2 --> 2H2O
14 del_H_298_H2O=2*del_H_H2O;
15 //Final
16 del_H_298=del_H_298_HCL+del_H_298_H2O;
17
18 disp('KJ',del_H_298,'Standard Heat')
19
20 //End
```

Scilab code Exa 4.6 Find the Standard Heat of Methanol Synthesis

```
1 clear all;
2 clc;
3
4 //Example 4.6
5 //Caption : Program to Find the Standard Heat of
     Methanol Synthesis
6
7 //To find Approx Value
8 function[A]=approx(V,n)
9   A=round(V*10^n)/10^n;//V-Value  n-To what place
10  funcprot(0)
11 endfunction
```

```
12
13 function[Q]=IDCPH(T0,T,dA,dB,dC,dD)
14   t=T/T0;
15   Q=(dA+((dB/2)*T0*(t+1))+((dC/3)*T0*T0*((t^2)+t+1))
       +(dD/(t*T0*T0)))*(T-T0)
16   funcprot(0);
17 endfunction
18
19
20 //Methanol Synthesis @ 1073.15K(800'C)
21 //CO + 2H2 --> CH3OH
22 del_H_CO=-110.525//@298K from Table C.4
23 del_H_CH3OH_g=-200.660;//@298K from Table C.4
24 del_H_298=((1)*del_H_CH3OH_g)-((1)*del_H_CO);//KJ/
     mol
25 T0=298.15;
26 T=1073.15;
27 R=8.314;
28 //Moles (CH3OH,CO,H2)
29 n=[1;-1;-2];
30 //A..from Table C.1
31 A=[2.211;3.376;3.249];
32 //B..from Table C.1
33 B=(10^-3)*[12.216;0.557;0.422];
34 //C..from Table C.1
35 C=(10^-6)*[-3.450;0;0];
36 //D..From table C.1
37 D=(10^5)*[0;-0.031;0.083];
38
39 del_A=0;
40 del_B=0;
41 del_C=0;
42 del_D=0;
43 for(i=1:3)
44   del_A=del_A+n(i,1)*A(i,1);
45   del_B=del_B+n(i,1)*B(i,1);
46   del_C=del_C+n(i,1)*C(i,1);
47   del_D=del_D+n(i,1)*D(i,1);
```

```
48 end
49
50 I=IDCPH(T0,T,del_A,del_B,del_C,del_D);
51 del_H=approx(del_H_298+(R*I/10^3),3);
52
53 disp('KJ',del_H,'Standard Heat Of Enthalpy');
54
55 //End
```

Scilab code Exa 4.7 Find Max Temperature reached in Combustion of CH4

```
1 clear all;
2 clc;
3
4 //Example 4.7
5 //Caption : Program To Find Max Temperature reached
     in Combustion of Methane
6
7 function[Q]=MCPH(T0,T,A,B,C,D)
8   t=T/T0;
9   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))
10  funcprot(0);
11 endfunction
12
13
14 //Combustion Of methane
15 //CH4 + 2O2 --> CO2 + 2H2O
16 R=8.314;
17 del_H_CO2=-393509;//from table C.4
18 del_H_O2=-241818;//from table C.4
19 del_H_CH4=-74520;//from table C.4
```

```
20 del_H_298=del_H_CO2+(2*del_H_O2)-del_H_CH4;
21 del_Hp=-del_H_298;
22 //moles of reactants
23 n_CH4=1;
24 n_O2=2+(0.2*2);//20% Excess
25 n_N2=n_O2*(79/21);
26 //Moles Of Products..(CO2,H2O,O2,N2)
27 np=[1;2;0.4;9.03];
28 //A..from Table C.1
29 A=[5.457;3.470;3.639;3.280];
30 //B..from Table C.1
31 B=(10^-3)*[1.045;1.450;0.506;0.593];
32 //C..from Table C.1
33 C=(10^-6)*[0;0;0;0];
34 //D..From table C.1
35 D=(10^5)*[-1.157;0.121;-0.227;0.040];
36
37 E_A=0;
38 E_B=0;
39 E_C=0;
40 E_D=0;
41 for(i=1:4)
42   E_A=E_A+np(i,1)*A(i,1);
43   E_B=E_B+np(i,1)*B(i,1);
44   E_C=E_C+np(i,1)*C(i,1);
45   E_D=E_D+np(i,1)*D(i,1);
46 end
47
48 T0=298.15;
49 a=round(T0);//Initial
50 i=-1
51 while (i==-1)
52   b=R*MCPH(T0,a,E_A,E_B,E_C,E_D);
53   c=b*(a-T0);
54   flag=del_Hp-c;
55   if(flag<=100) then
56     T=a-1;
57     i=1;
```

```
58    else
59      a=a+1;
60      i=-1;
61    end
62  end
63  disp('K',T,'Temperature Required(Approx)')
64  //End
```

Scilab code Exa 4.8 Find the Heat Requirement for the Reactor

```
1  clc;
2  clear all;
3
4  //Example 4.8
5  //Caption : Program to Find the Heat Requirement for
       the Reactor
6
7  //To find Approx Value
8  function[A]=approx(V,n)
9    A=round(V*10^n)/10^n;//V-Value  n-To what place
10   funcprot(0)
11 endfunction
12
13 function[Q]=MCPH(T0,T,A,B,C,D)
14   t=T/T0;
15   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))
16   funcprot(0);
17 endfunction
18
19
20 //CH4 + H2O --> CO + 3H2 (A)
21 //CH4 + 2H2O --> CO2 + 4H2  (B)
```

```
22 del_H_A=205813;//J
23 del_H_B=164647;//J
24 //0.87 mol of CH4 for (A) (1-0.87)mol of CH4 for (B)
25 del_H_298=(0.87*del_H_A)+(0.13*del_H_B);
26 R=8.314;
27 T0=298.15;
28 T_A=600;//Cooled
29 T_B=1300;//Heated
30 //Moles of reactants (CH4,H2O)
31 nr=[1;2];
32 //Moles of Products (CO,H2,CO2,H2O)
33 np=[0.87;3.13;0.13;0.87];
34 //For Reactants
35 //for CH4
36 I1=MCPH(T0,T_A,1.702,9.081*(10^-3),-2.164*(10^-6),0)
     ;
37 //For H2O
38 I2=MCPH(T0,T_A,3.470,1.450*(10^-3),0,0.121*(10^5));
39 del_Hr=R*((nr(1,1)*I1)+(nr(2,1)*I2))*(T0-T_A);//J
40 //For Products
41 //for CO
42 I1=MCPH(T0,T_B,3.376,0.557*(10^-3),0,-0.031*(10^5));
43 //For H2
44 I2=MCPH(T0,T_B,3.249,0.422*(10^-3),0,0.083*(10^5));
45 //for CO2
46 I3=MCPH(T0,T_B,5.457,1.045*(10^-3),0,-1.157*(10^5));
47 //For H2O
48 I4=MCPH(T0,T_B,3.470,1.450*(10^-3),0,0.121*(10^5));
49 del_Hp=R*((np(1,1)*I1)+(np(2,1)*I2)+(np(3,1)*I3)+(np
     (4,1)*I4))*(T_B-T0);//J
50 //del_H
51 del_H=del_H_298+del_Hr+del_Hp;
52 Q=approx(del_H,-1);
53 disp('J',Q,'Heat Required');
54
55 //End
```

Chapter 5

The Second Law Of Thermodynamics

Scilab code Exa 5.1 Find the Heat discarded to the River

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 5.1
11 //Caption : Program To Find the Heat discarded to
       the River
12
13 //Given Values
14 Tc=295;//K
15 Th=585;//K
16 W=800000;//KW
```

```
17 n_max=1-(Tc/Th);
18 n=approx(0.7*n_max,3)
19 Qc=approx(((1-n)/n)*W,-2);
20 disp('KW',Qc,'Heat required');
21
22 //End
```

Scilab code Exa 5.3 Find the Final Temperature

```
1  clear all;
2  clc;
3
4  //Example 5.3
5  //Caption : Program to Find the Final Temperature in
       Reversible Adiabatic Expansion
6
7  function[Q]=MCPS(T0,T,A,B,C,D)
8    t=T/T0;
9    Q=(A)+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)))*(t+1)
       /2))*((t-1)/log(t)))
10   funcprot(0);
11 endfunction
12
13
14 //Given values
15 P2=1;//bar
16 P1=5;//bar
17 T0=550;//K
18 A=1.702;
19 B=9.081*(10^-3);
20 C=-2.164*(10^-6);
21 D=0;
22
```

```
23 //Equation to be used
24 //(<Cp>s/R)ln(T2/T1)=ln(P2/P1) since del_S=0
25 //let I=(<Cp>s/R)
26
27 //T2=exp(log(1/5)/I);
28 a=T0-1;//Initial
29 i=-1;
30 while (i==-1)
31   b=MCPS(T0,a,A,B,C,D);
32   c=(log(1/5))/(log(a/T0));
33   flag=c-b;
34   if(flag<=0.0001) then
35     T=a;
36     i=1;
37   else
38     a=a-.01;
39     i=-1;
40   end
41 end
42
43 disp('K',T,'Final Temperature')
44
45 //End
```

Scilab code Exa 5.4 Find the change in entropy in a steel Casing

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
```

```
8  endfunction
9
10 //Example 5.4
11 //Caption : Program To Find the change in entropy in
       a steel Casing
12
13 //Given Values
14
15 //For Casting
16 Cp_Casting=0.5;//[KJ/Kg/K]
17 T1=723.15;//[K]
18 T0=298.15;//[K]
19 M_Casting=40;//[Kg]
20 //For Oil
21 Cp_Oil=2.5;//[KJ/Kg/K]
22 M_Oil=150;//[Kg]
23
24 //Formula  M1Cp1dT'=M2Cp2dT
25 //40*0.5*(T-723.15)=150*2.5*(298.15-T)
26 //Whence
27 T=((T1*M_Casting*Cp_Casting)+(T0*M_Oil*Cp_Oil))/((
       M_Casting*Cp_Casting)+(M_Oil*Cp_Oil));
28
29 //(a)-change in entropy For casting
30 del_S_Casting=approx(M_Casting*Cp_Casting*integrate(
       '1/T','T',T1,T),2);
31
32 disp('KJ/K',del_S_Casting,'(a)Change In Entropy of
       Casting')
33
34 //(b)-change in entropy For Oil
35 del_S_Oil=approx(M_Oil*Cp_Oil*integrate('1/T','T',T0
       ,T),2);
36
37 disp('KJ/K',del_S_Oil,'(b)Change In Entropy of Oil')
38
39 //(c)-Total
40 del_S_total=del_S_Casting+del_S_Oil;
```

```
41 disp('KJ/K',del_S_total,'(c)Total entropy change')
42
43 //End
```

Scilab code Exa 5.5 Find the Rate of Heat Transfer and Entropy

```
1  clear all;
2  clc;
3
4  //Example 5.5
5  //Caption : Program to Find the Rate of Heat
       Transfer and Entropy
6
7  //Given Values
8
9  //Gas A
10 rn_A=1;//rate[mol/s]
11 T_A=600;//[K]
12
13 //Gas B
14 rn_B=2;//rate[mol/s]
15 T_B=450;//[K]
16
17 //product
18 rn=rn_A+rn_B;//[mol/s]
19 T=400;//[K]
20 R=8.314;
21 Cp=(7/2)*R;
22 T_s=300;//Temperature[K]
23
24 //By equation (2.30) rQ=rn*H-rn_A*H_A-rn_B*H_B=rn_A(
       H-H_A)+rn_B*(H-H_B)  Rate of heat transfer
25 rQ=(rn_A*Cp*(T-T_A))+(rn_B*Cp*(T-T_B));//[J/s] or  [
```

```
    W]
26 //By eqn (5.22) rSg=rn*S-(rn_A*S_A)-(rn_B*S_B)-(rQ/
    T_s) rate of entropy generation for the process
27 rSg=approx((rn_A*Cp*log(T/T_A))+(rn_B*Cp*log(T/T_B))
    -(rQ/T_s),3);//[J/K/s]
28
29 disp('J/s or W',rQ,'Rate of heat transfer')
30 disp('J/K/s',rSg,'Rate of entropy generation')
31
32 //End
```

Scilab code Exa 5.6 Find the Feasibility of a Process

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 5.6
11 //Caption : Program to Find the Feasibility of a
    Process
12
13 //Given Values
14 //Saturated Steam
15 //At T=373.15K
16 H1=2676;//[KJ/Kg] from Steam table(App F)
17 S1=7.3554;//[KJ/Kg/K] from steam table(App F)
18 //At T=273.15K  Liquid water
19 H2=0;
```

```
S2=0;

T_sigma=273.15;//[K]
T_r=473.15;//[K]
Q_r=-2000;//[KJ]

del_H=H2-H1;
Q=del_H;
Q_sigma=Q-Q_r;

del_S=S2-S1;
//For Heat Reservoir at 473.15K
del_St_T_r=(-Q_r/T_r);//[KJ/K]
//For Heat reservoir provided by cooling water at
   273.15K
del_St_T_sigma=-Q_sigma/T_sigma;
del_S_total=del_S+del_St_T_r+del_St_T_sigma;
disp('Since del_S_total<0 Process not feasible')

//Actual
Q_r=approx((T_r/(T_r-T_sigma))*(del_H-(T_sigma*del_S
   )),1);
disp('KJ/Kg',Q_r,'Actual Heat transfer')

//End
```

Scilab code Exa 5.7 Find the Maximum Work obtained

```
clear all;
clc;


//Example 5.7
```

```
6  //Caption : Program to Find the Maximum Work
       obtained in a Steady state Flow
7
8
9  //To find Approx Value
10 function[A]=approx(V,n)
11   A=round(V*10^n)/10^n;//V-Value  n-To what place
12   funcprot(0)
13 endfunction
14
15 function[Q]=ICPH(T0,T,A,B,C,D)
16   t=T/T0;
17   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))*(T-T0)
18   funcprot(0);
19 endfunction
20
21 function[Q]=ICPS(T0,T,A,B,C,D)
22   t=T/T0;
23   Q=((A)*log(t))+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)
       ))*(t+1)/2))*(t-1))
24   funcprot(0);
25 endfunction
26
27
28 //Given Values
29 P1=50;//bar
30 P2=1.013;//bar
31 T1=800;//[K]
32 T2=300;//[K]
33 R=8.314;
34
35 //del_H=intergral(CpdT) in the limits T1 and T2
36 A=3.280;
37 B=0.593*(10^-3);
38 C=0;
39 D=0.040*(10^5);
40 del_H=R*ICPH(T1,T2,A,B,C,D);//[J/mol]
```

```
41
42 //del_S=integral[Cp(dT/T)] -Rln(P2/P1)  btw the
     limits T1,T2
43 del_S=(R*ICPS(T1,T2,A,B,C,D))-(R*log(P2/P1));//[J/
     mol/K]
44 W_ideal=approx(del_H-(T2*del_S),0);//[J/mol]
45 disp('J/mol',W_ideal,'Maximum Work')
46
47 //End
```

Scilab code Exa 5.8 Find the Maximum Possible Work for Ideal Condition

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 5.8
11 //Caption : Program To Find the Maximum Possible
     Work for Ideal Condition
12
13 //Given Values
14 //Saturated Steam
15 //At T=373.15K
16 H1=2676;//[KJ/Kg] from Steam table(App F)
17 S1=7.3554;//[KJ/Kg/K] from steam table(App F)
18 //At T=273.15K  Liquid water
19 H2=0;
```

```
20 S2=0;
21
22 T_sigma=273.15;//[K]
23 T_r=473.15;//[K]
24
25 del_H=H2-H1;
26 del_S=S2-S1;
27 W_ideal=del_H-(T_sigma*del_S);//[KJ/Kg]
28 Q=approx(abs(W_ideal*(T_r/(T_sigma-T_r))),1);//[KJ]
29 disp('KJ',Q,'Maximum Possible Work')
30
31 //End
```

Scilab code Exa 5.9 Find the Lost Work in Heat Exchangers

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 5.9
11 //Caption : Program to find the Lost Work in Heat
     Exchangers
12
13 //Given Values
14
15 T_H1=400;//[K]
16 T_H2=350;//[K]
17 T_C1=300;//[K]
```

```
T_sigma=300;//[K]
rn_H=1;//[mol/s]
R=8.314;
Cp=(7/2)*R;

T_C2_a=T_H2-10;
T_C2_b=T_H1-10;

//Figure
mtlb_axis('auto');
subplot(1,2,1);
X=[0,1];
Y=[T_C1,T_C2_a];
plot2d(X,Y);
Y=[T_H1,T_H2];
plot2d(X,Y,style=3,rect=[0,290,1,410]);
legend('Tc','Th')
X=[1,1];
Y=[290,410];
plot2d(X,Y);
X=[0,0.25];
Y=[T_C1,T_C1];
plot(X,Y,'--');
Y=[T_H1,T_H1];
plot(X,Y,'--');
X=[0.75,1];
Y=[T_C2_a,T_C2_a];
plot(X,Y,'--');
Y=[T_H2,T_H2];
plot(X,Y,'--');
xtitle("(a)Case 1,Cocurrent","Qc","T");

subplot(1,2,2);
X=[0,1];
Y=[T_C1,T_C2_b];
plot2d(X,Y);
Y=[T_H2,T_H1];
plot2d(X,Y,style=3,rect=[0,290,1,410]);
```

```
56 legend('Tc','Th')
57 X=[1,1];
58 Y=[290,410];
59 plot2d(X,Y);
60 X=[0,0.25];
61 Y=[T_C1,T_C1];
62 plot(X,Y,'--');
63 Y=[T_H2,T_H2];
64 plot(X,Y,'--');
65 X=[0.75,1];
66 Y=[T_C2_b,T_C2_b];
67 plot(X,Y,'--');
68 Y=[T_H1,T_H1];
69 plot(X,Y,'--');
70 xtitle("(b)Case 2,Countercurrent","Qc","T");
71
72 //Solution
73 //Equation to be used
74 //(rn_H*Cp(T_H2-T_H1))+(rn_C*Cp(T_C2-T_C1))=0  Eq(A)
75 //del_rS=rn_H*Cp*(ln(T_H2/T_H1)+kln(T_C2/T_C1))  k=
      rn_C/rn_H  r-->Rate  Eqn(B)
76 //rW_lost=T_sigma*del_rS  Eqn(C)
77
78 //(a)-Cocurrent
79 //by Eqn(A)
80 T_C2_a=T_H2-10;
81 k=(T_H1-T_H2)/(T_C2_a-T_C1);//k=rn_C/rn_H
82 //By Eqn(B)
83 del_rS=approx(rn_H*Cp*(log(T_H2/T_H1)+(k*log(T_C2_a/
      T_C1))),3);//[J/K/s]
84 //By Eqn(C)
85 rW_lost=approx(T_sigma*del_rS,1);//[J/s]or[W]
86 disp('(a)-Cocurrent')
87 disp('J/K/s',del_rS,'Rate Of change of entropy')
88 disp('J/s or W',rW_lost,'Lost Work')
89
90 //(b)-Countercurrent
91 T_C2_b=T_H1-10;
```

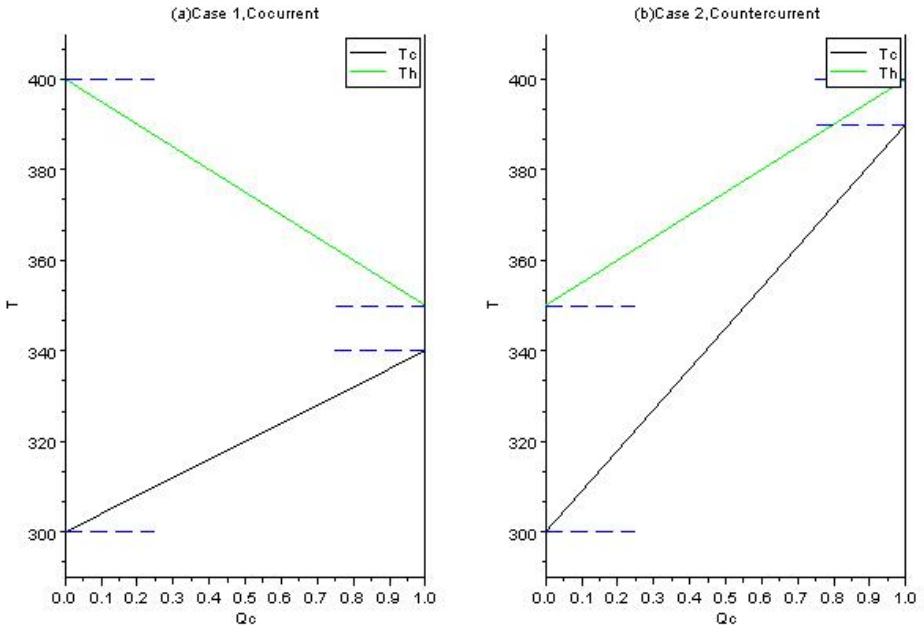

Figure 5.1: Find the Lost Work in Heat Exchangers

```
92  k=(T_H1-T_H2)/(T_C2_b-T_C1);//k=rn_C/rn_H
93  //By Eqn(B)
94  del_rS=approx(rn_H*Cp*(log(T_H2/T_H1)+(k*log(T_C2_b/
        T_C1))),3);//[J/K/s]
95  //By Eqn(C)
96  rW_lost=approx(T_sigma*del_rS,1);//[J/s] or [W]
97  disp('(b)-Countercurrent')
98  disp('J/K/s',del_rS,'Rate Of change of entropy')
99  disp('J/s or W',rW_lost,'Lost Work')
100
101 //End
```

Chapter 6

Thermodynamic Properties Of Fluids

Scilab code Exa 6.1 Find the Changes in enthalpy and entropy

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 6.1
11 //Caption : Program To find the Changes in enthalpy
     and entropy
12
13 //Given Values
14
15 //At Temp T1=298.15K
16 T1=298.15;//[K]
```

```
17 P1=1;//[bar]
18 P2=1000;//[bar]
19 Cp_T1=75.305;//[KJ Kmol/K]
20 V1_T1=18.071*10^-3;//[m^3/Kmol]
21 V2_T1=18.012*10^-3;//[m^3/Kmol]
22 beta1_T1=256*10^-6;//[1/K]
23 beta2_T1=366*10^-6;//[1/K]
24
25 //At Temp T2=323.15K
26 T2=323.15;//[K]
27 P1=1;//[bar]
28 P2=1000;//[bar]
29 Cp_T2=75.314;//[KJ Kmol/K]
30 V1_T2=18.234*10^-3;//[m^3/Kmol]
31 V2_T2=18.174*10^-3;//[m^3/Kmol]
32 beta1_T2=458*10^-6;//[1/K]
33 beta2_T2=568*10^-6;//[1/K]
34
35 //Solution
36
37 //Formula to be used
38 //Eqn (6.28) del_H=((Cp)(T2-T1))-((V)(1-(beta)(T2)(
      P2-P1))
39 //Eqn (6.29) del_S=((Cp)ln(T2/T1)-((beta)(V)(P2-P1))
40
41 //For P=1
42 Cp=(Cp_T1+Cp_T2)/2;
43 //For T=323.15K
44 V=(V1_T2+V2_T2)/2;
45 beta_T=(beta1_T2+beta2_T2)/2;
46
47 del_H=approx((Cp*(T2-T1))+(V*(1-(beta_T*T2))*(P2-P1)
      *10^5*10^-3),0);
48 del_S=approx((Cp*(log(T2/T1)))-(beta_T*V*(P2-P1)
      *10^5*10^-3),2);
49
50 disp('KJ/Kmol',del_H,'Change In Enthalpy')
51 disp('KJ/Kmol/K',del_S,'Change In Entropy')
```

```
52
53 //End
```

Scilab code Exa 6.3 find Entropy and Enthalpy of Saturated isobutane

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=ICPS(T0,T,A,B,C,D)
11   t=T/T0;
12   Q=((A)*log(t))+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)
       ))*(t+1)/2))*(t-1))
13   funcprot(0);
14 endfunction
15
16 function[Q]=ICPH(T0,T,A,B,C,D)
17   t=T/T0;
18   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))*(T-T0)
19   funcprot(0);
20 endfunction
21
22 //Example 6.3
23 //Caption : Program to find Entropy and Enthalpy of
     Saturated isobutane Vapor
24
25 //Given Values
```

```
26
27 T0=300;//[K]
28 T=360;//[K]
29 R=8.314;
30 P=15.14;//[bar]
31 A=1.7765;
32 B=33.037*10^-3;
33 C=0;
34 D=0;
35 H0=18115;   //J/mol
36 S0=295.976;//J/mol/K
37
38 //Graph
39 X=[0,0.10,0.50,2,4,6,8,10,12,14,15.41];
40 Y1
      =[1.780,1.700,1.514,1.293,1.290,1.395,1.560,1.777,2.073,2.432,2.7
      //[(dZ/dT)p/P]
41 Y2
      =[2.590,2.470,2.186,1.759,1.591,1.544,1.552,1.592,1.658,1.750,1.8
      //[-(Z-1)/P]
42 subplot(1,2,1);
43 plot2d(Y1,X);
44 xgrid();
45 xtitle("(a)","P(bar)","[(dZ/dT)p/P]X10^4(K^-1 bar
      ^-1)");
46 subplot(1,2,2);
47 plot2d(Y2,X);
48 xgrid();
49 xtitle("(b)","P(bar)","[-(Z-1)/P]X10^2(bar^-1)");
50
51
52 //Area Under the Curve (a)
53 Y1=Y1*10^-4;
54 A1=0;
55 for i=2:11;
56   A1=A1+((X(i-1)-X(i))*Y1(i));
57 end
58 disp('(X 10^-4) K^-1',A1*10000,'Area under the graph
```

```
    (a)')
//Area Under the Curve (b)
Y2=Y2*10^-2;
A2=0;
for i=2:11;
  A2=A2+((X(i-1)-X(i))*Y2(i));
end
disp(approx(A2,4),'Area under the graph(b)')


K=A1*T;//Hr/RT
//From Eqn(6.47)
Hr=R*T*(K);//[J/mol]
//From Eqn(6.48)
Sr=R*(K-(A2));//[J/mol/K]

//From Eqn(6.49) and Eqn(6.50)
H1=R*ICPH(T0,T,A,B,C,D);
S1=R*ICPS(T0,T,A,B,C,D);

H=H0+H1+Hr;
S=approx(S0+S1+Sr-(R*log(P)),3);

disp('J/mol',H,'Enthalpy')
disp('J/mol/K',S,'Entropy')
disp('Note: The Answer is different with that of the
    Book because the Method Used to find the Area
   under the Graph is done by finding the area of
   small       Rectangles')

//End
```

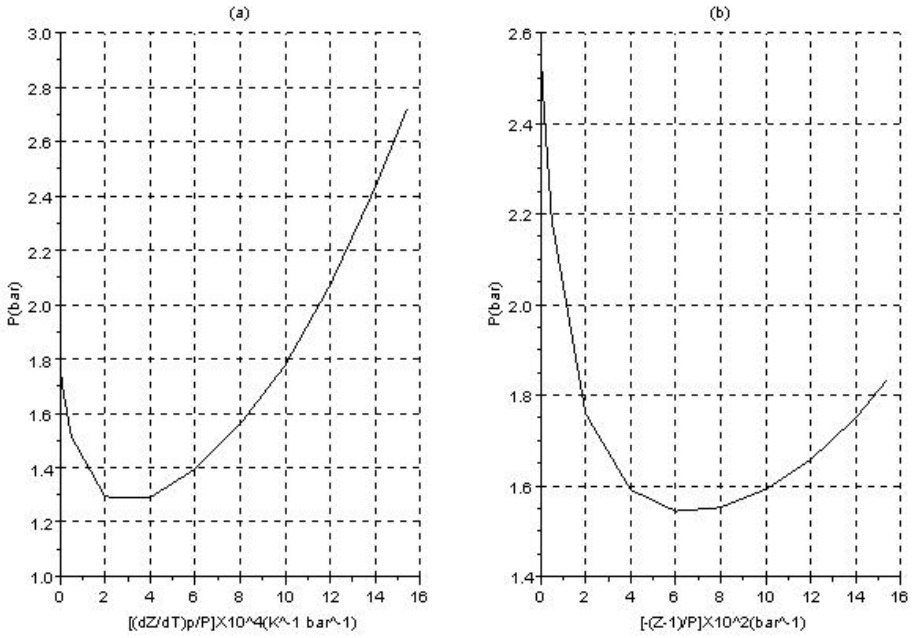

Figure 6.1: find Entropy and Enthalpy of Saturated isobutane

Scilab code Exa 6.4 Find the Residual Enthalpy and Residual Entropy

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 6.4
//Caption : Program to Find the Residual Enthalpy
   and Residual Entropy

//Given Values and values from Table(3.1)
T=500;//[K]
R=8.314;
Tc=425.1;//[K]
P=50;//[bar]
Pc=37.96;//[bar]
omega=0.08664;
si=0.4636;
Tr=T/Tc;
Pr=P/Pc;
alpha_Tr=Tr^(-0.5);//a(Tr)
//Using Eqn(3.50)
Beta=omega*(Pr/Tr);
//Using Eqn(3.51)
q=si*alpha_Tr/(omega*(Tr^1.5));

//using eqn(3.49)
//Z=1+beta-q*beta*((Z-beta)/((Z+(epsilon*beta))*(Z+(
   sigma*beta)))

//calculation of Z
Z=1;//Initial
a=Z;
```

```
for i=0:10
  b=1+Beta-(q*Beta*((a-Beta)/(a*(a+Beta))));
  if((b-a)==0.0001)
     break;
   end
   a=b;
   i=i+1;
end
Z=approx(b,3)

//Using Eqn(6.64) and eqn(6.65)
//(Hr/RT)=Z-1+[(d ln(alpha_Tr)/d ln Tr)-1]qI    I=ln
   ((Z+beta)/Z)  d ln(alpha_Tr)/d ln Tr=-0.5
//Sr/R)=ln(Z-beta)+[d ln(alpha_Tr)/d ln Tr]qI   I=ln
   ((Z+beta)/Z)  d ln(alpha_Tr)/d ln Tr=-0.5
I=log((Z+Beta)/Z);
Hr=approx(R*T*(Z-1+((-0.5-1)*q*I)),0);
Sr=approx(R*(log(Z-Beta)+(-0.5*q*I)),3);

disp('Using Redlich/Kwong Equation')
disp('J/mol',Hr,'Residual Enthalpy')
disp('J/mol/K',Sr,'Residual Entropy')

//End
```

Scilab code Exa 6.6 Find the State of Steam at the Exit Nozzle

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
```

```
7   funcprot(0)
8 endfunction
9
10 //Example 6.6
11 //Caption : Program to Find the State of Steam at
      the Exit Nozzle
12
13 //Given values
14 P1=1000;//[KPa]
15 T=533.15;//[K]
16 P2=200;//[KPa]
17 H1=2965.2;//[KJ/kg]  from Steam tables
18 S1=6.9680;//[KJ/Kg/K] From steam tables
19 S2=S1;
20 S_l=1.5301;//[KJ/Kg/K]  Entropy Of Saturated Liquid
      @ 200KPa
21 S_v=7.1268;//[KJ/Kg/K]  Entropy Of Saturated vapor
      @ 200KPa
22 H_l=504.7;//[KJ/Kg]     Enthalpy Of saturated liquid
       @ 200KPa
23 H_v=2706.7;//[KJ/Kg]    Enthalpy Of saturated vapor
      @ 200KPa
24
25
26 //Solution
27 //find x_v from the eqn S=(1-x_v)S_l+x_c*S_v
28 x_v=approx((S1-S_l)/(S_v-S_l),4);
29
30 //From Eqn(6.73a)
31 H2=((1-x_v)*H_l)+(x_v*H_v);
32 del_H=approx(H2-H1,0);//[KJ/Kg]
33
34 disp('%',x_v*100,'Percent vapor')
35 disp('%',(1-x_v)*100,'Percent Liquid')
36 disp('KJ/Kg',del_H,'Change In Enthalpy')
37
38 //End
```

Scilab code Exa 6.7 Find how much Energy must be Transferred

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 6.7
//Caption : Program to Find how much Energy must be
   Transferred to the Tank

//Given values(from steam tables)
H=293;//[KJ/Kg]  at 343.15K
H_liquid=419.1;//[KJ/Kg]  at 373.15K
H_vapor=2676;//[KJ/Kg]    at 373.15K
V_vapor=1.5;//[m^3]
m1_liquid=500;//[Kg]
rho_liquid=0.001044;//[m^3/Kg]
rho_vapor=1.673;//[m^3/Kg]
del_m=750;//[Kg]

//using the eqn Q=(m2H2)tank-(m1H1)tank-(H*del_m)
   tank
m1_vapor=(V_vapor-(m1_liquid*rho_liquid))/rho_vapor;
//Term2=((m1H1)tank
Term2=(m1_liquid*H_liquid)+(m1_vapor*H_vapor);
mT=m1_liquid+del_m+m1_vapor;
//Solving Eqn By matrix Method
```

```
29 //m_vapor+m_liquid=mT  and  (rho_vapor*m_vapor)+(
   rho_liquid*rho_vapor)=V_vapor
30 A=[1,1;rho_vapor,rho_liquid];
31 B=[mT;V_vapor];
32 X=inv(A)*B;
33 m2_vapor=X(1,1);
34 m2_liquid=X(2,1);
35
36 Term1=(m2_liquid*H_liquid)+(m2_vapor*H_vapor);
37 Q=approx(Term1-Term2-del_m*H,0);
38
39 disp('KJ',Q,'Heat Required')
40 disp('Note: The Answer Varies With That of The Book
   because the calculations as in Book do not give
   the Answer the Book results')
41 //End
```

Scilab code Exa 6.8 Find V U S and H fo 1 butene

```
1  clear all;
2  clc;
3
4  //Example 6.8
5  //Caption : Program to Find V U S and H fo 1-butene
6
7  //To find Approx Value
8  function[A]=approx(V,n)
9    A=round(V*10^n)/10^n;//V-Value  n-To what place
10   funcprot(0)
11 endfunction
12
13 function[Q]=SRB(Tr,Pr,omega)
14   B0=0.083-(0.422/(Tr^1.6));
```

```
15    diffr_B0=0.675/(Tr^2.6);//dB0/dTr
16    B1=0.139-(0.172/(Tr^4.2));
17    diffr_B1=0.722/(Tr^5.2);//dB0/dTr
18    Q=-Pr*(diffr_B0+(omega*diffr_B1));
19    funcprot(0);
20  endfunction
21
22  function[H]=HRB(Tr,Pr,omega)
23    B0=0.083-(0.422/(Tr^1.6));
24    diffr_B0=0.675/(Tr^2.6);//dB0/dTr
25    B1=0.139-(0.172/(Tr^4.2));
26    diffr_B1=0.722/(Tr^5.2);//dB0/dTr
27    H=Pr*(B0-(Tr*diffr_B0)+(omega*(B1-(Tr*diffr_B1))))
        ;
28    funcprot(0);
29  endfunction
30
31  function[Q]=ICPS(T0,T,A,B,C,D)
32    t=T/T0;
33    Q=((A)*log(t))+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)
        ))*(t+1)/2))*(t-1))
34    funcprot(0);
35  endfunction
36
37  function[Q]=ICPH(T0,T,A,B,C,D)
38    t=T/T0;
39    Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
        /(t*T0*T0)))*(T-T0)
40    funcprot(0);
41  endfunction
42
43  //Example 6.8
44  //Caption : Program to Find V U S and H fo 1-butene
45
46  //Given values(from steam tables)
47  Tc=420;//[K]
48  Pc=40.43;//[bar]
49  omega=0.191;
```

```
Tn=266.9;//[K]
A0=1.967;
B0=31.630*10^-3;
C0=-9.837*10^-6;
D0=0;
T1=473.15;//[K]
P=70;//[bar]
R=8.314;
//From Table(E.3) And Table(E.4)
Z0=0.485;
Z1=0.142;

Tr=T1/Tc;
Pr=P/Pc;
Z=Z0+(omega*Z1);
V=approx((Z*R*T1*10^-2)/P,4);//[m^3/Kmol]

//step(a) vaporization at T1 and P1=P_saturated
//using eqn(6.70)  lnP_sat=A-(B/T)
//Solving eqn ln(1.0133)=A-(B/266.9) and ln(40.43)=A
   -(B/420)
a=[1,(-1/266.9);1,(-1/420)];
b=[log(1.0133);log(40.43)];
x=(a^-1*b);
A=x(1,1);
B=x(2,1);
//using eqn(4.12)  del_Hn/RTn=1.092*(ln Pc-1.013)
   /(0.930-Tr_n)
Tr_n=Tn/Tc;
del_Hn=R*Tn*(1.092*(log(Pc)-1.013)/(0.93-Tr_n));//[J
   /mol]
T2=273.15;//[K]
Tr=T2/Tc;
//Using Eqn(4.13)  del_H/del_Hn=((1-Tr)/(1-Tr_n))
   ^0.38
del_H_a=del_Hn*((1-Tr)/(1-Tr_n))^0.38;
del_S_a=approx(del_H_a/T2,2);

```

```
84 //Step(b)transition to ideal gas State at(T1,P1)
85 P_sat=exp(A-(B/273.15));
86 Pr=P_sat/Pc;
87 Tr=T2/Tc;
88 Hr_b=approx(R*Tc*HRB(Tr,Pr,omega),0)//[J/mol]
89 Sr_b=approx(R*SRB(Tr,Pr,omega),2)//[J/mol/K]
90
91 //Step(c) Change to (T2,P2) in ideal-gas state
92
93 H_c=approx(R*ICPH(T2,T1,A0,B0,C0,D0),0);//[J/mol]
94 S=R*ICPS(T2,T1,A0,B0,C0,D0);//[J/mol/K]
95 del_S_c=approx(S-(R*log(P/P_sat)),2);//[J/mol/K]
96
97 //Step(d) Transition to actual final state at(T2,P2)
98 //Using eqn(6.76) and eqn(6.77)
99 //Hr/RTc=Hr0/RTc+(omega*Hr1/RTc)
100 //Sr/R=Sr0/R+(omega*Sr1/R)  Sr0,Sr1 from Tables(E.5)
101 Tr=T1/Tc;
102 Pr=P/Pc;
103 Hr_d=R*Tc*(-2.294+(omega*-0.713));
104 Sr_d=R*(-1.566+(omega*-0.726));
105
106 H=approx(del_H_a-Hr_b+H_c+Hr_d,0);
107 S=approx(del_S_a-Sr_b+del_S_c+Sr_d,2);
108 U=approx(H-(P*V*10^2),0);
109
110 disp('m^3/Kmol',V,'Volume(V)=')
111 disp('J/mol',U,'Internal energy(U)=')
112 disp('J/mol',H,'Enthalpy(H)=')
113 disp('J/mol/K',S,'Entropy(S)=')
114
115 disp('Note: The Answer here Slightly Varies with
      That of Book because of the different
      approximation')
116
117 //End
```

Scilab code Exa 6.9 Find Residual Enthalpy and Entropy and V

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10
11 //Example 6.9
12 //Caption : Program to Find Residual Enthalpy and
       Entropy and V by Lee/Kesler
13
14
15 //Given Values
16 T=450;//[K]
17 P=140;//[bar]
18
19 //pseudo parameters
20 Tc1=304.2;//[K]
21 Tc2=369.8;//[K]
22 Pc1=73.83;//[bar]
23 Pc2=42.48;//[bar]
24 Tpc=(0.5*Tc1)+(0.5*Tc2);
25 Ppc=(0.5*Pc1)+(0.5*Pc2);
26
27 Tpr=T/Tpc;
28 Ppr=P/Ppc;
29
```

```
30 Z0=0.697;
31 Z1=0.205;
32
33 omega1=0.224;
34 omega2=0.152;
35 omega=(0.5*omega1)+(0.5*omega2);
36
37 Z=Z0+(omega*Z1);
38
39 V=approx(Z*R*T*10/P,1);//[cm^3/mol]
40
41 //(H/RT)0=-1.73  (H/RT)1=-0.169
42 H=approx(R*Tpc*(-1.73+(omega*-0.169)),0);//[J/mol]
43 S=approx(R*(-0.967+(omega*-0.330)),2);//[J/mol/K]
44
45 disp('cm^3/mol',V,'Volume(V)=')
46 disp('J/mol',H,'Residual Enthaply(H)=')
47 disp('J/mol/K',S,'Residual Entropy(S)=')
48
49 //End
```

Chapter 7

Applications Of Thermodynamics To Flow Process

Scilab code Exa 7.2 Find the Ratio of Area in a Nozzle

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10
11 //Example 7.2
12 //Caption : program to find the Ratio of Area's in a
       Nozzle for Diff Pressures
13
14 //Given Values
```

```
15 T=573.15;//[K]
16 P=[700,600,500,400,300,200];//[KPa]
17 //values for H,V,S for various P from steam tables
18 H=[3059.8,3020.4,2975.71,2923.5,2859.9,2777.35];//[
     KJ/Kg]
19 V=[371.39,418.25,481.26,571.23,711.93,970.04];//[cm
     ^3/g]
20 S=7.29997*ones(1,6);//[KJ/Kg/K]  Isentropic
21 u0=30;//[m/s]
22 u=zeros(1,6);
23
24 //Using Eq(7.3)
25 //u^2=u1^2-2(H-H1)
26 u=approx(sqrt((u0^2-2*(H-H(1))*10.^3)),1);
27
28 //Using Eq(2.27)
29 //A/A1=u1*V/V1*u;
30 c=u(1)./V(1);
31 K=approx((c*V./u),3);//K=A/A1  c=u1/V1
32
33 Ans=[P',V',u',K'];
34 disp(Ans,' P/[KPa]  V/[cm^3/g]  u/[m/s]  A/A1')
35
36 //End
```

Scilab code Exa 7.3 Find Critical Pressure and Discharge Pressure

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
```

```
7    funcprot(0)
8  endfunction
9
10
11 //Example 7.3
12 //Caption : Program to Find the Critical Pressure
       and the Discharge Pressure
13
14 //Given Values
15 T1=573.15;//[K]
16 R=8314;
17 P1=700;//[KPa]
18 M=18.015;
19 Gamma=1.3;
20 u0=30;//[m/s]
21
22 //(a)
23 //Using Eqn(7.12)
24 //K=P2/P1=(2/(Gamma+1))^(Gamma/(Gamma-1))
25 K=approx((2/(Gamma+1))^(Gamma/(Gamma-1)),2);//
       rounding to 2 decimal places
26
27 P1V1=round(R*T1/M);//m^2/s^2
28 //Using Eqn(7.11)
29 //u_throat^2=u^2+2(Gamma)(P1V1)/(Gamma-1)[1-(P2/P1)
       ^((Gamma-1)/Gamma))]
30 u_throat=approx(sqrt(u0^2+((2*Gamma*P1V1/(Gamma-1))
       *(1-(K^((Gamma-1)/Gamma))))),2);
31
32 disp(K,'(a)Critical Pressure ratio(P2/P1)')
33 disp(' m/s',u_throat,' Velocity at the throat')
34
35 //(b)Mach No 2.0
36 u=2*u_throat;
37 K=(1-((u^2-u0^2)*(Gamma-1)/(2*Gamma*P1V1)))^(Gamma/(
       Gamma-1));//K=P2/P1
38 P2=round(K*P1);
39
```

```
40 disp('KPa',P2,'(b) Discharge Pressure for Mach Number
       of 2.0')
41
42 //End
```

Scilab code Exa 7.4 Find the final Temperature and its Entropy change

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[H]=HRB(Tr,Pr,omega)
11   B0=0.083-(0.422/(Tr^1.6));
12   diffr_B0=0.675/(Tr^2.6);//dB0/dTr
13   B1=0.139-(0.172/(Tr^4.2));
14   diffr_B1=0.722/(Tr^5.2);//dB0/dTr
15   H=Pr*(B0-(Tr*diffr_B0)+(omega*(B1-(Tr*diffr_B1))))
       ;
16   funcprot(0);
17 endfunction
18
19 function[Q]=SRB(Tr,Pr,omega)
20   B0=0.083-(0.422/(Tr^1.6));
21   diffr_B0=0.675/(Tr^2.6);//dB0/dTr
22   B1=0.139-(0.172/(Tr^4.2));
23   diffr_B1=0.722/(Tr^5.2);//dB0/dTr
24   Q=-Pr*(diffr_B0+(omega*diffr_B1));
25   funcprot(0);
```

```
26 endfunction
27
28 //Example 7.4
29 //Caption : Program to Find the final Temperature
      and its Entropy change
30
31 //Given Values
32 P1=20;//[bar]
33 T=400;//[K]
34 P2=1;//[bar]
35 R=8.314;
36
37 //Solution
38
39 //using Eq(6.84)
40 //del_H=Cp(T2-T1)+Hr2-Hr1=0  but Hr2=0
41 //T2=Hr1/Cp + T1
42 Tc=369.8;//[K]
43 Pc=42.48;//[bar]
44 omega=0.152;
45 a=T;//Initial
46 for i=1:2
47   Tr=a/Tc
48   Pr=P1/Pc;
49   Hr1=R*Tc*HRB(Tr,Pr,omega);//[J/mol]
50   Cp=R*(1.213+(28.785*10^-3*a)-(8.824*10^-6*a*a));//
       [J/mol/K]
51   T2=(Hr1/Cp)+a;
52   Tm=(a+T2)/2;
53   i=i+1;
54   a=Tm;
55 end
56 Tm=a;
57 T2=round(Tm)//[K]
58 Tr=T/Tc;
59 Sr=R*SRB(Tr,Pr,omega);
60
61 del_S=approx((Cp*log(T2/T))-(R*log(P2/P1))-Sr,2);
```

```
62
63 disp('J/mol/K',del_S,'Entropy')
64 disp('Positive Entropy represents the
      irreversibility of Throttling Process')
65 disp('K',T2,'Final Temperature')
66
67 //End
```

Scilab code Exa 7.6 Find the state of Steam at Discharge and Mass Rate

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 7.6
11 //Caption : Program to Find the state of Steam at
      Discharge & Mass Rate of Flow
12
13 //Given Values
14 P1=8600;//[KPa]
15 T1=773.15;//[K]
16 //values of Enthalpy and Entropy from Steam tables
17 H1=3391.6;//[KJ/Kg]
18 S1=6.6858;//[KJ/Kg/K]
19 eta=0.75;
20 P2=10000;//[KPa]
21 rW=56400;//[KW] or [KJ/s]
22 S2i=S1;//Isentropic
```

```
23
24 S2_liquid=0.6493;
25 S2_vapor=8.1511;
26 H2_liquid=191.8;
27 H2_vapor=2584.8;
28
29 x2=(S2i-S2_liquid)/(S2_vapor-S2_liquid);
30
31 H2i=H2_liquid+(x2*(H2_vapor-H2_liquid));
32 del_Hs=H2i-H1;//[KJ/Kg]
33 del_H=eta*del_Hs;
34
35 H2=approx(H1+del_H,0);//[KJ/Kg]
36 x2=(H2-H2_liquid)/(H2_vapor-H2_liquid);
37 S2=approx(S2_liquid+(x2*(S2_vapor-S2_liquid)),4);
38
39 rm=approx(-rW/(H2-H1),2);//[Kg/s]
40 disp('KJ/Kg',H2,'Enthalpy')
41 disp('KJ/Kg/K',S2,'Entropy')
42 disp('Kg/s',rm,'Rate of mass change')
43
44 //End
```

Scilab code Exa 7.7 Find the isentropic Work Produced

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
```

```
9
10 function[Q]=MCPS(T0,T,A,B,C,D)
11   t=T/T0;
12   Q=(A)+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)))*(t+1)
       /2))*((t-1)/log(t)))
13   funcprot(0);
14 endfunction
15
16 function[Q]=MCPH(T0,T,A,B,C,D)
17   t=T/T0;
18   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))
19   funcprot(0);
20 endfunction
21
22 function[H]=HRB(Tr,Pr,omega)
23   B0=0.083-(0.422/(Tr^1.6));
24   diffr_B0=0.675/(Tr^2.6);//dB0/dTr
25   B1=0.139-(0.172/(Tr^4.2));
26   diffr_B1=0.722/(Tr^5.2);//dB0/dTr
27   H=Pr*(B0-(Tr*diffr_B0)+(omega*(B1-(Tr*diffr_B1))))
       ;
28   funcprot(0);
29 endfunction
30
31 function[Q]=SRB(Tr,Pr,omega)
32   B0=0.083-(0.422/(Tr^1.6));
33   diffr_B0=0.675/(Tr^2.6);//dB0/dTr
34   B1=0.139-(0.172/(Tr^4.2));
35   diffr_B1=0.722/(Tr^5.2);//dB0/dTr
36   Q=-Pr*(diffr_B0+(omega*diffr_B1));
37   funcprot(0);
38 endfunction
39
40 //Example 7.7
41 //Caption : Program to Find the isentropic Work
     Produced
42
```

```
43 //Given Values
44
45 T1=573.15;//[K]
46 P1=45;//[bar]
47 P2=2;//[bar]
48 Tc=282.3;//[K]
49 Pc=50.4;//[bar]
50 omega=0.087;
51 A=1.424;
52 B=14.394*10^-3;
53 C=-4.392*10^-6;
54 D=0;
55 R=8.314;
56
57 //Using Eqn(6.84)
58 //del_H=<Cp>h (T2-T1)+Hr2-Hr1
59 //Using Eqn(6.85))
60 //del_S=<Cp>s ln(T2/T1) - R*ln(P2/P1)+Sr2-Sr1
61
62 //(a) equations for Ideal gas
63 //No residuals terms, whence
64
65 //del_H=<Cp>h(T2-T1)
66 //del_S=<Cp>s ln(T2/T1) - R*ln(P2/P1)
67
68 del_S=0//isentropic
69 //Whence K = <Cp>s/R ln(T2/T1) = ln(P2/P1)
70 K=log(P2/P1);
71 //let c = <Cp>s/R
72 //T2=exp(K/c+ln(T1))
73 i=-1;
74 a=round(T1);//Initial
75 while (i==-1)
76   b=MCPS(T1,a,A,B,C,D);
77   temp=exp((K/b)+log(T1));
78   flag=a-temp;
79   if(flag<=0.1) then
80     T2=a;
```

```
81      i=1;
82    else
83      a=temp-0.1;
84      i=-1;
85    end
86  end
87  disp('(a)by Equations for an Ideal gas')
88  disp('K',approx(T2,1),'Temp = ')
89  Cp_h=R*MCPH(T1,T2,A,B,C,D);
90  del_Hs=Cp_h*(T2-T1);
91  Ws_a=approx(del_Hs,0);
92  disp('J/mol',Ws_a,'Work')
93
94  //(b)-Appropriate Generalized correlations
95
96  Tr1=T1/Tc;
97  Pr1=P1/Pc;
98
99  Hr1=R*Tc*HRB(Tr1,Pr1,omega);//[J/mol]
100 Sr1=R*SRB(Tr1,Pr1,omega);//[J/mol/K]
101
102 Tr2=T2/Tc;
103 Pr2=P2/Pc;
104
105 Sr2=R*SRB(Tr2,Pr2,omega);
106
107 //Using Eqn(6.85))
108 //del_S=<Cp>s ln(T2/T1) - R*ln(P2/P1)+Sr2-Sr1
109 //del_S=0 isentropic
110 //K=<Cp>s ln(T2/T1)=Rln(P2/P1)-Sr2+Sr1
111 K=R*log(P2/P1)-Sr2+Sr1;
112 //T2=exp((K/<Cp>s)+ln T1)
113 i=-1;
114 a=round(T1);//Initial
115 while (i==-1)
116   b=R*MCPS(T1,a,A,B,C,D);
117   temp=exp((K/b)+log(T1));
118   flag=a-temp;
```

```
119   if(flag<=0.1) then
120     T2=a;
121     i=1;
122   else
123     a=temp-0.1;
124     i=-1;
125   end
126 end
127
128 disp('(b)by Appropriate generalized correlations')
129 disp('K',approx(T2,1),'Temp = ')
130 Tr2=T2/Tc;
131
132 Sr2=R*SRB(Tr2,Pr2,omega);//[J/mol/K]
133 Hr2=R*Tc*HRB(Tr2,Pr2,omega);//[J/mol]
134 Cp_h=R*MCPH(T1,T2,A,B,C,D);
135 del_Hs=Cp_h*(T2-T1)+Hr2-Hr1;
136 Ws_b=approx(del_Hs,-1);
137 disp('J/mol',Ws_b,'Work')
138
139 //End
```

Scilab code Exa 7.8 Find the Work Required and Properties of Steam

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
```

```
10 //Example 7.8
11 //Caption : Program to Find the Work Required and
       Properties of Discharge Steam
12
13 //Given Values
14
15 P1=100;//[KPa]   (Tsat/tsat)=327.78K/99.63'C)
16
17 //From Steam Tables @ 100KPa
18 S1=7.3598;//[KJ/Kg/K]
19 H1=2675.4;//[KJ/Kg]
20
21 P2=300;//[KPa]
22 //From Steam Tables @ 300KPa
23 S2=S1;//Isentropic
24 H2i=2888.8;//[KJ/Kg]
25
26 eta=0.75;//Efficiency
27
28 del_H=H2i-H1;
29 del_H=del_H/eta;
30 H2=approx(H1+del_H,1);//[KJ/Kg]
31
32 //From Steam Tables w.r.t H2
33 T2=519.25;//[K]
34 S2=7.5019;//[KJ/Kg/K]
35
36 Ws=approx(del_H,1);//[KJ/Kg]   Work Reqd
37
38 disp('KJ/Kg',H2,'Enthalpy')
39 disp('KJ/Kg/K',S2,'Entropy')
40 disp('K',T2,'Temperature')
41 disp('KJ/Kg',Ws,'Work Done')
42
43 //End
```

Scilab code Exa 7.9 Find Work Reqiured and Discharge Temperature

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

function[Q]=MCPH(T0,T,A,B,C,D)
  t=T/T0;
  Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
     /(t*T0*T0)))
  funcprot(0);
endfunction

function[Q]=MCPS(T0,T,A,B,C,D)
  t=T/T0;
  Q=(A)+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)))*(t+1)
     /2))*((t-1)/log(t)))
  funcprot(0);
endfunction

//Example 7.9
//Caption : Program to Find Work Reqiured and
   Discharge Temperature of Methane

//Given Values

R=8.314;
```

```
28 T1=293.15;//[K]
29
30 P1=140;//[KPa]
31 P2=560;//[KPa]
32
33 eta=0.75;//[Efficiency]
34 A=1.702;
35 B=9.081*10^-3;
36 C=-2.164*10^-6;
37 D=0;
38
39 i=-1;
40 a=round(T1);//Initial
41 while (i==-1)
42   b=MCPS(T1,a,A,B,C,D);
43   b=b^-1;
44   c=T1*((P2/P1)^b);
45   flag=c-a;
46   if(flag<=0.0001) then
47     T2i=a;
48     i=1;
49   else
50     a=a+0.01;
51     i=-1;
52   end
53 end
54
55 Cps=R*MCPS(T1,T2i,A,B,C,D);
56 Cph=approx(R*MCPH(T1,T2i,A,B,C,D),3);
57
58 //from Eqn(7.19)
59 Ws=approx(Cph*(T2i-T1),0)//[J/mol]
60 Ws=approx(Ws/eta,0)//Actual work
61 del_H=Ws;
62
63 //From eqn(7.21) Actual discharge Temperature
64 //T2=T1+(del_H/Cph)
65 i=-1;
```

```
a=round(T2i);//Initial
chk=1;
while (i==-1)
  b=R*MCPH(T2i,a,A,B,C,D);
  c=del_H/(a-T1);
  flag=c-b;
  if(flag<=0.001) then
      T2=a;
      i=1;
  else
      a=a+0.001;
      i=-1;
  end
end
Cph_T2=approx(R*MCPH(T2i,T2,A,B,C,D),2);
disp('K',T2,'Temperature')
disp('J/mol/K',Cph_T2,'Enthalpy')
disp('J/mol',Ws,'Actual Work')

disp('Note: The answer in the Book varies with that
   of this code because the Calculation in the Book
   does not leads to the answer given')

//End
```

Scilab code Exa 7.10 Find Work Temperature Change and Entropy Change

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
```

```
7    funcprot(0)
8  endfunction
9
10 //Example 7.10
11 //Caption : Program to Find Work,Temperature Change
       and Entropy Change in Pump
12
13 //Given Values
14 T1=318.15;//[K]
15 P1=10;//[KPa]
16 P2=8600;//[KPa]
17 eta=0.75;//Efficiency
18
19 //Properties of saturated liquid water @ 318.15K
20 V=1010;//[cm^3/Kg]
21 V=1010*10^-6;//[m^3/Kg]
22 Beta=425*10^-6;//[K^-1]
23 Cp=4.178;//[KJ/Kg/K]
24
25 //From Eqn(7.24)
26 Ws=V*(P2-P1);//[KPa m^3/Kg]
27 del_H=Ws;
28 //From Eqn(7.17)
29 del_H=del_H/eta;
30 Ws=approx(del_H,2);
31
32 //From Eqn(7.25)
33 del_T=approx((del_H-(V*(1-(Beta*T1))*(P2-P1)))/Cp,2)
       ;
34
35 //From Eqn(7.26)
36 T2=T1+del_T;
37 del_S=approx(Cp*log(T2/T1)-(Beta*V*(P2-P1)),3);
38
39 disp('KJ/Kg',Ws,'Work Done')
40 disp('K',del_T,'Change in Temperature')
41 disp('KJ/Kg/K',del_S,'Change in Entropy')
42
```

```
43 //End
```

Chapter 8

Production Of Power From Heat

Scilab code Exa 8.1 Find the Thermal efficiency in a Steam Turbine

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 8.1
11 //Caption : Program to Find the Thermal efficiency
       in a Steam Turbine
12
13 //Given Values
14 //(a)-As in Example(7.6)
15 P1=8600;//[KPa]
16 T1=773.15;//[K]
```

```
17 //values of Enthalpy and Entropy from Steam tables
18 H1=3391.6;//[KJ/Kg]
19 S1=6.6858;//[KJ/Kg/K]
20
21 P2=10;//[KPa]
22 S2i=S1;//Isentropic
23
24 S2_liquid=0.6493;
25 S2_vapor=8.1511;
26 H2_liquid=191.8;
27 H2_vapor=2584.8;
28
29 x2=(S2i-S2_liquid)/(S2_vapor-S2_liquid);
30
31 H2i=H2_liquid+(x2*(H2_vapor-H2_liquid));
32 del_Hs_1=approx((H2i-H1),1);//[KJ/Kg]
33 Ws=del_Hs_1;
34 H3i=H2i;
35 H4=H2_liquid;
36 //Applying Eqn(8.2)
37 Q_condenser=approx((H4-H3i),1);//heat Of condenser
     [KJ/Kg]
38 //From Example(7.10)
39 //Properties of saturated liquid water @ 318.15K
40 V=1010;//[cm^3/Kg]
41 V=1010*10^-6;//[m^3/Kg]
42 Beta=425*10^-6;//[K^-1]
43 Cp=4.178;//[KJ/Kg/K]
44
45 //From Eqn(7.24)
46 Ws_2=approx((V*(P1-P2)),1)//[KPa m^3/Kg]
47 del_Hs_2=Ws_2;
48 H1=H4+del_Hs_2;
49 //Enthalpy Of saturated steam at 8600KPa and 773.15K
50 H2=3391.6;//[KJ/Kg]
51 //Applying Eqn(8.2)
52 Q_boiler=H2-H1;
53
```

```
Ws_Rankine=-Q_boiler-Q_condenser;
eta=approx((abs(Ws_Rankine)/Q_boiler),3);
disp('(a)Rankine Cycle')
disp(eta,'Thermal Efficiency')

//(b)

eta_b=0.75;

del_H_1=del_Hs_1*eta_b;
Ws_turbine=del_H_1;
H3=H2+del_H_1;
Q_condenser=H4-H3;

//By Example 7.10 for the pump
Ws_pump=del_Hs_2/eta_b;
del_H_2=Ws_pump;
Ws_net=Ws_turbine+Ws_pump;
H1=H4+del_H_2;

Q_boiler=H2-H1;
efficiency=approx(abs(Ws_net)/Q_boiler,4);
disp('(b)Practical cycle with 0.75 efficiency')
disp(efficiency,'Thermal Efficiency')

//(c)
//By rating of Power Cycle
rWs_net=-80000;//[KJ/s]  Power Rating
rm=approx(rWs_net/Ws_net,2);

rQ_boiler=approx(rm*Q_boiler/1000,1);//[MW]
rQ_condenser=approx(rm*Q_condenser/1000,1);//[MW]
disp('(c)By rating of Power Cycle');
disp('kg/s',rm,'Steam Rate')
disp('MW',rQ_boiler,'Heat Transfer rate in boiler')
disp('MW',rQ_condenser,'Heat Transfer rate in
   condenser')

```

```
91 //End
```

Scilab code Exa 8.4 Find the Efficiency in a Gas Turbine

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 8.4
11 //Caption : Program to Find the Efficiency in
      Various Cycles in a Gas turbine
12
13 //Given Values
14 K=6;//Pb/Pa
15 T1=298.15;//[K]
16 Tmax=1033.15;//[K]
17 Gamma=1.4;
18
19 //(a)  Gamma=1.4
20 //From Eqn(8.12)
21 eta_a=approx(1-((1/K)^((Gamma-1)/Gamma)),1);
22 disp('(a)Efficiency of an ideal air cycle')
23 disp(eta_a,'Efficiency')
24
25 //(b)  eta_c=0.83  eta_t=0.86
26 eta_c=0.83;
27 eta_t=0.86;
28 K2=Tmax/T1;
```

```
29 alpha=(K)^((Gamma-1)/Gamma);
30
31 //Using Eqn(8.13)
32 eta_b=approx(((eta_t*eta_c*K2*(1-(1/alpha)))-(alpha
      -1))/((eta_c*(K2-1))-(alpha-1)),3);
33 disp('(b)Thermal efficiency of an air cycle if the
      Compressor and Turbine Operate adiabatically')
34 disp(eta_b,'Thermal efficiency')
35
36 //End
```

Chapter 9

Refrigerator And Liquifaction

Scilab code Exa 9.1 Find the COP of a Refrigerator

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 9.1
11 //Caption : Program to Find the COP of a
       Refrigerator
12
13 //Given Values
14 T1=261.15;//[K] Temgeratureof refrigerated space
15 T2=294.15;//[K] Temperature of cooling water
16 dT=5.6;//[K]    Temperature Difference
17 Qc=35.2;//[kW]  Refrigerant Capacity
18 eta=0.8;//Efficeincy(b)
```

```
19
20 //(a) Carnot Refrigerator
21 Tc=T1-dT;
22 Th=T2+dT;
23 //Using Eqn(9.3)
24 w=approx(Tc/(Th-Tc),2);
25 disp(w,'(a)Coefficient of Performance for Carnot
      Refrigerator')
26
27 //(b)  Vapor Compression Cycle
28
29 //From Table(9.1)
30 //@ Tc=255.55K
31 H2=388.13;//[KJ/Kg]
32 S2=1.7396;//[KJ/Kg/K]
33
34 //@ Th=299.75K
35 H4=236.76;//[KJ/Kg]
36
37 S3=S2;//Isentropic
38 //Hence
39 H3=420.27;//[KJ/Kg]
40 //Step 2 --> 3
41 del_Hs=H3-H2;
42 //But Compressor Efficiency = 0.80
43 del_Hs=del_Hs/eta
44 //Step 1 --> 4
45 H1=H4;//isenthalpic
46 w=approx((H2-H4)/del_Hs,2);
47 disp(w,'(b)Coefficient of Performance for Vapor
      Compression Cycle')
48 rm=approx(Qc/(H2-H4),4);//[Kg/s]
49 disp('kg/s',rm,'Circulation rate');
50
51 //End
```

Scilab code Exa 9.2 Find Power Requirement For Various seasons

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 9.2
//Caption : Program to Find Power Requirement For
   Various seasons

//Given Values
Qh_Winter=30;//[KW]
Qc_Summer=60;//[KW]
Tc_Winter=283.15;//[K]
Th_Winter=303.15;//[K]
Tc_Summer=278.15;//[K]
Th_Summer=298.15;//[K]

//For WINTER's
//Using Eqn(5.7)
Qc_Winter=Qh_Winter*(Tc_Winter/Th_Winter);
//Using Eqn(9.1)
W_Winter=approx((Qh_Winter-Qc_Winter),2);//[KW]
disp('KW',W_Winter,'Power Requirement for WINTER''s'
   )

//For SUMMER's
```

```
29 //Combining Eqn(9.2) And Eqn(9.3)
30 W_Summer=approx((Qc_Summer*((Th_Summer-Tc_Summer)/
      Tc_Summer)),2);
31 disp('KW',W_Summer,'Power Requirement for SUMMER''s'
      )
32
33 //End
```

Scilab code Exa 9.3 Find the Temperature of the High Pressure steam

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 9.3
11 //Caption : Program to Find the Temperature of the
      High Pressure steam
12
13 //Given Values
14
15 x=0.25;
16 //For Superheated Methane
17 //By Perry and Green  by linear interpolation
18 H4=1140;//[KJ/Kg]  @ 300K and 60 Bar
19 H15=1188.9;//[KJ/Kg]  @ 295K and 1 Bar
20 //By interpolation based on lnP
21 T_sat=111.5;//[K]
22 H9=285.4;//[KJ/Kg]  Saturated Liquid
```

```
23 H12=769.9;//[KJ/Kg] Saturated Vapor
24 S12=9.521;//[KJ/Kg/K] Saturated vapor
25
26 T5=253.6;//[K]
27 H5=1009.8;//[KJ/Kg]  @ 60 Bar
28
29 //From Eqn(9.7)
30 z=((x*(H12-H5))+H4-H15)/(H9-H15);
31
32 H14=((H5-H4)/(1-z))+H15;//[KJ/Kg]
33 //Whence
34 T14=227.2;//[K]  @ 1Bar
35
36 H7=H5-((1-z)/(1-x)*(H14-H12));//[KJ/Kg]
37 T7=197.6;//[K]  @ 60Bar
38
39 //From Eqn(9.8)
40 z=approx((H4-H15)/(H9-H15),4);
41
42 H7=H4-((1-z)*(H15-H12));
43 T7=206.6;//[K]
44
45 disp('%',z*100,'Fraction of methane liquefied')
46 disp('K',T7,'Temperature of High Pressure steam
      entering the throttle valve')
47
48 //End
```

Chapter 10

Vapor Liquid Equillibrium Introduction

Scilab code Exa 10.1 Plot the Graphs of P vs x1 y1 and t vs x1 y1

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 10.1
11 //Caption : Program to Plot the Graphs of P vs x1,y1
       and t vs x1,y1
12
13 //Antoinie Equations
14
15 //ln P1_sat=14.2724-(2945.47/(T-49.15))  [KPa]
16 //ln P2_sat=14.2043-(2972.64/(T-64.15))  [KPa]
```

```
17
18 //(a) Graph Showing P vs x1 and P vs y1 for T=348.15
     K
19 T=348.15;//[K]
20 //using BUBL P calculations
21
22 //Calculation of P1_sat and P2_sat at T=348.15K
23 P1_sat=approx(exp(14.2724-(2945.47/(T-49.15))),2)
     //KPa
24 P2_sat=approx(exp(14.2043-(2972.64/(T-64.15))),2)
     //KPa
25
26 //Using Eqn P=P2_sat+(P1_sat-P2_sat)x1
27
28 x=[0:0.2:1];
29 P=approx(P2_sat+((P1_sat-P2_sat)*x),2);
30 y=approx(x*P1_sat./P,4);
31
32 disp('Explanations Of graph')
33
34 Ans=[x',y',P'];
35 disp(Ans,'   x1       y1       P/PKa')
36
37 y1=0.6;
38 y2=1-y1;
39 P_dew=approx(1/((y1/P1_sat)+(y2/P2_sat)),2)
40 x1=approx(y1*P_dew/P1_sat,4)
41
42 // Plotting the graph
43 subplot(1,2,1);
44 T=348.15;//[K]
45
46 P1_sat=approx(exp(14.2724-(2945.47/(T-49.15))),2)
     //KPa
47 P2_sat=approx(exp(14.2043-(2972.64/(T-64.15))),2)
     //KPa
48
49 x=[0:0.2:1];
```

```
P=approx(P2_sat+((P1_sat-P2_sat)*x),2);
y=approx(x*P1_sat./P,4);

mtlb_axis('auto');
plot(x,P,'g-')  //P vs x1
plot(y,P,'b-')  //P vs y1
x=[0,0.1];
P=[P2_sat,P2_sat];
plot(x,P,'--')  //P2_sat
x=[0.9,1];
P=[P1_sat,P1_sat];
plot(x,P,'r--') //P1_sat

x1=0.6;
P_b=approx(P2_sat+((P1_sat-P2_sat)*x1),2);
y1=approx(x1*P1_sat/P_b,4);
x=[x1,y1];
P=[P_b,P_b];
plot(x,P,'bo-')  //b--b'

y1=0.6;
y2=1-y1;
P_c=approx(1/((y1/P1_sat)+(y2/P2_sat)),2)
x1=approx(y1*P_c/P1_sat,4)

x=[x1,y1];
P=[P_c,P_c];
plot(x,P,'ro-')  //c'--c

P=[(P_b+10),P_b,P_c,(P_c-10)];
x=[0.6,0.6,0.6,0.6];
plot(x,P,'go-')  //a--b--c--d--0.6

P=[(P_c-10),30];
x=[0.6,0.6];
plot(x,P,'yo--')

P=[110,80];
```

```
x=[0.6,0.6];
plot(x,P,'w')
legend('P vs x1(Liquid)','P vs y1(Vapor)','P2_sat','
   P1_sat',"b--b''","c''--c",'a--b--c--d--0.6')
xtitle('(a)T/t=348.15K','x1,y1','P/Kpa')

disp("This is the liquid-phase composition at point
   c''")

//(b) Graph showing (t vs x1) and (t vs y1) for a
   pressure of 70KPa
//Example 10.2(b)
P=70;//[KPa]

T1_sat=approx(2945.47/(14.2724-log(P))+49.15,2);
T2_sat=approx(2972.64/(14.2043-log(P))+64.15,2);

T=[T1_sat,347.15,351.15,355.15,359.15,T2_sat];

P1_sat=approx(exp(14.2724-(2945.47./(T-49.15))),2);
    //KPa
P2_sat=approx(exp(14.2043-(2972.64./(T-64.15))),2);
    //KPa

x=approx((P-P2_sat)./(P1_sat-P2_sat),3);
y=approx((x.*P1_sat)/P,3);

Ans=[x',y',T'];
disp(Ans,'   x1        y1        T/t(K/C`)')

//at x1=0.6;
x1_b=0.6;
x2_b=1-x1_b;

T_a=347.15;// Intermediate Temperature  (Point a in
   graph)
P1_sat_a=approx(exp(14.2724-(2945.47./(T_a-49.15)))
   ,2);  //KPa
```

```
P2_sat_a=approx(exp(14.2043-(2972.64./(T_a-64.15)))
   ,2);  //KPa
alpha=P1_sat_a/P2_sat_a;  //Initial
a=T_a;
i=-1;
while(i==-1)
  P2_sat_b=P/((x1_b*alpha)+x2_b);
  b=approx(2972.64/(14.2043-log(P2_sat_b))+64.15,2);
  dT=abs(a-b);
  if(dT==0)
    i=0;
    T_b=b;
  end
  alpha=exp(0.0681-(2945.47/(b-49.15))+(2972.64/(b
     -64.15)));  //Eqn C
  a=b;
end
P1_sat_b=approx(exp(14.2724-(2945.47./(T_b-49.15)))
   ,2);  //KPa
y1_b=approx((x1_b*P1_sat_b)/P,4);  //b'

disp('K',T_b,'Hence by iteration Temp(Temp at b) at
   x1=0.6 is ')
disp('KPa',P1_sat_b,'Hence by iteration P1_sat at x1
   =0.6 is ')
disp(y1_b,'Composition of Vapor(b') at x1=0.6')


//At y1=0.6
y1_c=0.6;
y2_c=1-y1_c;
T_d=355.15;// Intermediate Temperature  (Point a in
   graph)
P1_sat_d=approx(exp(14.2724-(2945.47./(T_d-49.15)))
   ,2);  //KPa
P2_sat_d=approx(exp(14.2043-(2972.64./(T_d-64.15)))
   ,2);  //KPa
alpha=P1_sat_d/P2_sat_d;  //Initial
```

```
d=T_d;
i=-1;
while(i==-1)
  P1_sat_c=P*(y1_c+(y2_c*alpha));
  c=approx(2945.47/(14.2724-log(P1_sat_c))+49.15,2);
  dT=abs(d-c);
  if(dT==0)
    i=0;
    T_c=c;
  end
  alpha=exp(0.0681-(2945.47/(c-49.15))+(2972.64/(c
     -64.15)));  //Eqn C
  d=c;
end
P1_sat_c=approx(exp(14.2724-(2945.47./(T_c-49.15)))
   ,2);  //KPa
x1_c=approx((y1_c*P)/P1_sat_c,4);  //c'

disp('K',T_c,'Hence by iteration Temp(Temp at b) at
   y1=0.6 is ')
disp('KPa',P1_sat_c,'Hence by iteration P1_sat at y1
   =0.6 is ')
disp(x1_c,'Composition of liqiud(c') at y1=0.6')

//Graph
subplot(1,2,2);
T=linspace(T1_sat,T2_sat,10);

P1_sat=approx(exp(14.2724-(2945.47./(T-49.15))),2);
    //KPa
P2_sat=approx(exp(14.2043-(2972.64./(T-64.15))),2);
    //KPa

x=approx((P-P2_sat)./(P1_sat-P2_sat),3);
y=approx((x.*P1_sat)/P,3);

plot(x,T,'g-');
plot(y,T,'b-');
```

```
181
182 xsat=[0,0.1];
183 T2sat=[T2_sat,T2_sat];
184 plot(xsat,T2sat,'--')   //T2_sat
185
186 xsat=[0.9,1];
187 T1sat=[T1_sat,T1_sat];
188 plot(xsat,T1sat,'r--') //T1_sat
189
190 Tcc=[T_c,T_c];
191 xc=[x1_c,y1_c];
192 plot(xc,Tcc,'ro-')   //c--c'
193
194 Tbb=[T_b,T_b];
195 xb=[x1_b,y1_b];
196 plot(xb,Tbb,'bo-')   //b--b'
197
198 Tabcd=[T_d,T_c,T_b,T_a];
199 xabcd=[0.6,0.6,0.6,0.6];
200 plot(xabcd,Tabcd,'go-')   //a--b--c--d--0.6
201
202 Tao=[T_a,340];
203 xao=[0.6,0.6];
204 plot(xao,Tao,'yo--')
205 legend('T vs x1(Liquid)','T vs y1(Vapor)','T2_sat','
    T1_sat',"c''--c","b--b''",'d--c--b--a--0.6')
206 xtitle('(b)P=70KPa','x1,y1','T(K)')
207
208 //End
```

Scilab code Exa 10.2 Find the Composition of the vapor and Liquid phase

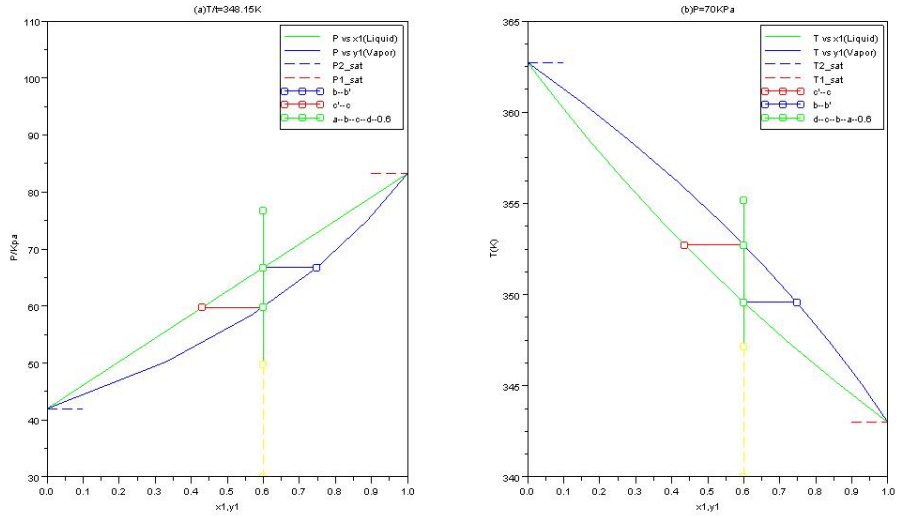

Figure 10.1: Plot the Graphs of P vs x1 y1 and t vs x1 y1

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 10.2
//Caption : Program to Find the Composition of the
   vapor and Liquid phases

//Taking CO2 as Species 1 and Water as Species 2
H=990;//[Bar] Henry's Law const
T=283.15;//[K]
P2_sat=0.01227;//[Bar]  from Steam Tables
x1=0.01;//Assumed
x2=1-x1;
```

```
y1=1;
P=approx((x1*H)+(x2*P2_sat),3);
x1=approx((y1*P)/H,4);
x2=1-x1;
y2=approx((x2*P2_sat)/P,4);
y1=1-y2;
disp(x1,'Composition in liquid Phase')
disp(y1,'Composition in vapor Phase')
disp('Bar',P,'Pressure Exerted on Can')
disp('Hence Vapor phase chosen is nearly pure')

//End
```

Scilab code Exa 10.3 Find Pressure Temperature and Composition

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 10.3
//Caption : Program to Find Pressure,Temperature and
    Composition for a system

//Equations to be Used
// ln v1=A*(x2^2)  ln v2=A*(x1^2)  Where A
   =2.771-0.00523T

//Antoine Equations
```

```
17 //ln P1_sat = 16.59158-(3643.31/(T-33.424))
18 //ln P2_sat = 14.25326-(2665.54/(T-53.424))
19 //P = E(xi * Vi * Pi_sat)  E--Summation          Eqn
      10.6
20 //P = 1/E(yi / (vi * Pi_sat))  E--Summation     Eqn
      10.7
21
22 //(a) Calculate P and (yi) , for T=318.15K and x1
      =0.25
23
24 //BULB P Calculation
25 T=318.15;//[K]  Given
26 x1=0.25; //Given
27 x2=1-x1;
28 P1_sat = approx(exp(16.59158-(3643.31/(T-33.424)))
      ,2);//[KPa]
29 P2_sat = approx(exp(14.25326-(2665.54/(T-53.424)))
      ,2);//[KPa]
30 A=approx(2.771-(0.00523*T),3);
31 v1=approx(exp(A*(x2^2)),3);
32 v2=approx(exp(A*(x1^2)),3);
33
34 //Form Eqn(10.6)
35 P_a=approx((x1*v1*P1_sat)+(x2*v2*P2_sat),2);//[KPa]
36 y1_a=approx((x1*v1*P1_sat)/P_a,3);
37 y2_a=approx((x2*v2*P2_sat)/P_a,3);
38
39 disp('(a)P and [yi] for T=318.15K and x1=0.25')
40 disp('BUBL P calculations')
41 disp('KPa',P_a,'P = ')
42 disp(y1_a,'y1 = ')
43 disp(y2_a,'y2 = ')
44
45 //(b) Calculate P and (xi) , for T=318.15K and y1
      =0.60
46
47 //DEW P calculation
48 y1=0.6;
```

```
y2=1-y1;
T=318.15;//[K]
P1_sat = approx(exp(16.59158-(3643.31/(T-33.424)))
   ,2);//[KPa]
P2_sat = approx(exp(14.25326-(2665.54/(T-53.424)))
   ,2);//[KPa]
A=approx(2.771-(0.00523*T),3);
v1=0.1;//Assumed
v2=0.1;//Assumed
a1=v1;
a2=v2;
i=-1;
while(i==-1)
  P=approx(1/((y1/(a1*P1_sat))+(y2/(a2*P2_sat))),2);
  x1=approx(y1*P/(a1*P1_sat),4);
  x2=1-x1;
  b1=approx(exp(A*(x2^2)),4);
  b2=approx(exp(A*(x1^2)),4);
  dt=abs(b1-a1);
  if(dt==0)
    i=0;
    v1=b1;
    v2=b2;
    break;
  end
  a1=b1;
  a2=b2;
end
x1_b=x1;
x2_b=1-x1_b;
P_b=P;
v1_b=v1;
v2_b=v2;
disp('(b)P and [xi] for T=318.15K and y1=0.60')
disp('DEW P calculations')
disp('kPa',P_b,'P = ')
disp(x1_b,'x1 = ')
disp(x2_b,'x2 = ')
```

```
85
86 //(c) Calculate T and (yi) for P = 101.33 KPa and x1
       =0.85
87
88 //BUBL T calculation
89 P=101.33;
90 x1=0.85;
91 x2=1-x1;
92 T1_sat=approx((3643.31/(16.59158-log(P)))+33.424,2);
93 T2_sat=approx((2665.54/(14.25326-log(P)))+53.424,2);
94 T=(x1*T1_sat)+(x2*T2_sat);
95 a=T;//Initial
96 i=-1;
97 while(i==-1)
98   A=approx(2.771-(0.00523*a),4);
99   v1=approx(exp(A*(x2^2)),4);
100  v2=approx(exp(A*(x1^2)),4);
101  P1_sat = approx(exp(16.59158-(3643.31/(a-33.424)))
       ,2);//[KPa]
102  P2_sat = approx(exp(14.25326-(2665.54/(a-53.424)))
       ,2);//[KPa]
103  alpha=P1_sat/P2_sat;
104  P1_sat=approx(P/((x1*v1)+(x2*v2/alpha)),2);
105  b=approx((3643.31/(16.59158-log(P1_sat)))
       +33.424,2);
106  dt=abs(b-a);
107  if(dt==0)
108    i=0;
109    T=b;
110    break;
111  end
112  a=b;
113 end
114 T_c=T;
115 y1_c=approx(x1*v1*P1_sat/P,3);
116 y2_c=1-y1_c;
117 disp('(c)T and [yi] for P=101.33kPa and x1=0.')
118 disp('BUBL T calculations')
```

```
disp('K',T_c,'Temperature = ')
disp(y1_c,'y1 = ')
disp(y2_c,'y2 = ')

//(d) Calculate T and (xi) for P = 101.3 KPa and y1 =0.4
P=101.3;
y1=0.4;
y2=1-y1;
T1_sat=approx((3643.31/(16.59158-log(P)))+33.424,2);
T2_sat=approx((2665.54/(14.25326-log(P)))+53.424,2);
T=(y1*T1_sat)+(y2*T2_sat);
v1=1;  //Initially
v2=1;  //Initially
a=T;//Initial
i=-1;
while(i==-1)
  A=approx(2.771-(0.00523*a),4);
  P1_sat = approx(exp(16.59158-(3643.31/(a-33.424))),2);//[KPa]
  P2_sat = approx(exp(14.25326-(2665.54/(a-53.424))),2);//[KPa]
  alpha=P1_sat/P2_sat;
  x1=approx((y1*P)/(v1*P1_sat),4);
  x2=1-x1;
  v1=approx(exp(A*(x2^2)),4);
  v2=approx(exp(A*(x1^2)),4);
  P1_sat=P*((y1/v1)+(y2*alpha/v2));
  b=approx((3643.31/(16.59158-log(P1_sat)))+33.424,2);
  dt=abs(a-b);
  if(dt==0)
    T=a;
    i=0;
    break;
  end
  a=b;
end
```

```
153 T_d=T;
154 x1_d=x1;
155 x2_d=x2;
156 disp('(d)T and [xi] for P=101.33kPa and y1=0.40')
157 disp('DEW T calculations')
158 disp('K',T,'T = ')
159 disp(x1_d,'x1 = ')
160 disp(x2_d,'x2 = ')
161
162 //(e) Taz , (xi_az) and (yi_az)  for T = 318.15K
163 T = 318.15;
164 // Relative Volatility   alpha_12=(y1/x1)/(y2/x2)
165 //At Azeotrope y1=x1 and y2=x2 and alpha_12=1
166 P1_sat = approx(exp(16.59158-(3643.31/(T-33.424)))
      ,2);//[KPa]
167 P2_sat = approx(exp(14.25326-(2665.54/(T-53.424)))
      ,2);//[KPa]
168 //From eqn (10.5) alpha_12=(v1*P1_sat)/(v2*P2_sat)
169 A=approx(2.771-(0.00523*T),4);
170
171 //When x1=0 v2=1 and v1=exp(A)
172 alpha_12_x10=P1_sat*exp(A)/P2_sat;
173
174 //When x1=1 v1=1 and v2=exp(A)
175 alpha_12_x11=P1_sat/(P2_sat*exp(A));
176
177 //But this is not Azeotrope  (at Azeotrope alpha_12
      =1)
178
179 //v1_az/v2_az=(P2_sat/P1_sat)=K
180 K=P2_sat/P1_sat;
181
182 //ln(v1/v2)=ln(K)=A(1-(2*x1))
183 x1_az=approx((A-log(K))/(2*A),3);
184 x2_az=1-x1_az;
185 y1_az=x1_az;
186 y2_az=x2_az;
187 v1_az=approx(exp(A*(x2_az^2)),3);
```

```
188 v2_az=approx(exp(A*(x1_az^2)),3);
189 P_az=approx(v1_az*P1_sat,2);
190
191 disp('Azeotropic Pressure and Azeotropic Composition
      for T = 318.15K')
192 disp('KPa',P_az,'Azeotropic Pressure = ')
193 disp(x1_az,'x1_az')
194 disp(y1_az,'y1_az')
195
196 //End
```

Scilab code Exa 10.4 Find the Dewpoint and Bubblepoint Pressure

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 10.4
11 //Caption : Program To Find the Dewpoint and
      Bubblepoint Pressure
12
13 T=283.15;//[K]
14 //(a) Dew Point Pressure
15 //Species=[" Methane ";" Ethane ";" Propane "];
16 y=[0.1,0.2,0.7];
17
18 P1=6.9;//[bar]
19 K1=[20,3.25,0.92];
```

```
20 x1=approx(y./K1,3);
21
22 P2=10.34;//[bar]
23 K2=[13.2,2.25,0.65];
24 x2=approx(y./K2,3);
25
26 P3=8.7;//[bar]
27 K3=[16,2.65,0.762];
28 x3=approx(y./K3,3);
29
30 i=1;
31 j=1;
32 P=[P1,P2,P3];
33 x=[x1;x2;x3];
34 E1=zeros(1,3);
35 for(i=1:3)
36   for(j=1:3)
37     E1(i)=E1(i)+x(i,j);//Summation
38     //j=j+1;
39   end
40   //i=i+1;
41 end
42 P_dew = 8.7;
43 Ans=[y',K1',x1',K2',x2',K3',x3';1,0,E1(1),0,E1(2),0,
     E1(3)];
44 disp(     '                P=6.9 bar       P=10.34 bar
         P=8.7 bar')
45 disp(Ans,'    yi       Ki      yi/Ki      Ki      yi/Ki
         Ki       yi/Ki')
46 disp('Last Row Represents the summation')
47 disp('KPa',P_dew,'The dew Point Pressure')
48
49 T=283.15;//[K]
50 //(b) Bubble Point Pressure
51 //Species=[" Methane ";" Ethane ";" Propane "];
52 x=[0.1,0.2,0.7];
53
54 P1=26.2;//[bar]
```

```
55 K1=[5.6,1.11,0.335];
56 y1=approx(x.*K1,3);
57
58 P2=27.6;//[bar]
59 K2=[5.25,1.07,0.32];
60 y2=approx(x.*K2,3);
61
62 P3=26.54;//[bar]
63 K3=[5.49,1.1,0.33];
64 y3=approx(x.*K3,3);
65
66 i=1;
67 j=1;
68 P=[P1,P2,P3];
69 y=[y1;y2;y3];
70 E2=zeros(1,3);
71 for(i=1:3)
72   for(j=1:3)
73     E2(i)=E2(i)+y(i,j);//Summation
74     //j=j+1;
75   end
76   //i=i+1;
77 end
78 P_Bubble = 26.54;
79 Ans=[x',K1',y1',K2',y2',K3',y3';1,0,E2(1),0,E2(2),0,
     E2(3)];
80 disp(     '                P=26.2 bar       P=27.6 bar
         P=26.54 bar')
81 disp(Ans,'   xi       Ki      xiKi      Ki      xiKi
         Ki       xiKi')
82 disp('Last Row Represents the summation')
83 disp('KPa',P_Bubble,'The Bubble Point Pressure')
84
85 //End
```

Scilab code Exa 10.5 Find L V xi and yi for a System

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 10.5
//Caption : Program to Find L V {xi} and {yi} for a
   System

z1=0.45;
z2=0.35;
z3=0.2;
P=110;//[KPa]
T=353.15;//[K]
P1_sat=195.75;//[KPa]
P2_sat=97.84;//[KPa]
P3_sat=50.32;//[KPa]

//BUBL Calculation
x1=z1;
x2=z2;
x3=z3;
P_BUBL=(x1*P1_sat)+(x2*P2_sat)+(x3*P3_sat);

//DEW Calculation
y1=z1;
```

```
30 y2=z2;
31 y3=z3;
32 P_Dew=1/((y1/P1_sat)+(y2/P2_sat)+(y3/P3_sat));
33
34 //Since P_Bubl<P<P_dew
35 //Flash Calculation
36 K1=P1_sat/P;
37 K2=P2_sat/P;
38 K3=P3_sat/P;
39
40 //Finding V from Eqn(10.17)
41 //E((zi*Ki)/(1+(V*(Ki-1))))=1
42
43 x=0;
44 F_x=(((z1*K1)/(1+((K1-1)*x)))+((z2*K2)/(1+((K2-1)*x)
      ))+((z3*K3)/(1+((K3-1)*x)))-1);
45 F_a=F_x;
46
47 x=0.9;
48 F_x=(((z1*K1)/(1+((K1-1)*x)))+((z2*K2)/(1+((K2-1)*x)
      ))+((z3*K3)/(1+((K3-1)*x)))-1);
49 F_b=F_x;
50 A=0;
51 B=0.9;
52 i=1;
53 while(i==1)
54   a=A;
55   F_a=(((z1*K1)/(1+((K1-1)*a)))+((z2*K2)/(1+((K2-1)*
        a)))+((z3*K3)/(1+((K3-1)*a)))-1);
56   b=B;
57   F_b=(((z1*K1)/(1+((K1-1)*b)))+((z2*K2)/(1+((K2-1)*
        b)))+((z3*K3)/(1+((K3-1)*b)))-1);
58   x1=((a*F_b)-(b*F_a))/(F_b-F_a);
59   F_x1=(((z1*K1)/(1+((K1-1)*x1)))+((z2*K2)/(1+((K2
        -1)*x1)))+((z3*K3)/(1+((K3-1)*x1)))-1);
60
61   if((F_a*F_x1)<0) then
62     flag=1;
```

```
63      A=a;
64      B=x1;
65    else((F_x1*F_b)<0)
66      flag=2;
67      A=x1;
68      B=b;
69    end
70    x1_a=approx(x1,4);
71    b_a=approx(b,4);
72    a_a=approx(a,4);
73    if(x1_a==b_a)
74      V=approx(x1,4);
75      i=0;
76      break;
77    elseif(x1_a==a_a)
78      root=approx(x1,4);
79      i=0;
80      break;
81    end
82
83  end
84  disp(V,'Hence By solving the polynomial V = ')
85  L=1-V;
86  //from eqn 10.16
87  //yi=(zi*Ki)/(1+(V*(Ki-1)))
88  y1=approx((z1*K1)/(1+((K1-1)*V)),4);
89  y2=approx((z2*K2)/(1+((K2-1)*V)),4);
90  y3=approx((z3*K3)/(1+((K3-1)*V)),4);
91  x1=approx(y1/K1,4);
92  x2=approx(y2/K2,4);
93  x3=approx(y3/K3,4);
94  y=[y1 y2 y3];
95  x=[x1 x2 x3];
96  disp(L,'Moles Of liquid')
97  disp(V,'Moles Of vapor')
98  disp(x,'Mole fraction Of liquid')
99  disp(y,'Mole fraction Of vapor')
100
```

```
101 //End
```

Scilab code Exa 10.6 Find the Composition of Vapor and Liquid Phases

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 10.6
11 //Caption : Program To Find the Composition of Vapor
       and Liquid Phases
12
13 //Given Values
14 z=[0.1,0.2,0.7];
15 K=[10,1.76,0.52];
16
17 V_01=[0.1:0.1:1]
18 for(i=1:10)
19   for(j=1:3)
20     y_01(i,:)=approx((z.*K)./(1+(V_01(i)*(K-1))),3);
21   end
22   Sum(i)=sum(y_01(i,:));
23 end
24
25 for(i=1:10)
26   if(Sum(i+1)<1)
27     a1=V_01(i);
28     i=-1;
```

```
29      break;
30    end
31  end
32
33  V_001=[a1:0.01:a1+0.1]
34  for(i=1:10)
35    for(j=1:3)
36      y_001(i,:)=approx((z.*K)./(1+(V_001(i)*(K-1)))
            ,3);
37    end
38    Sum(i)=sum(y_001(i,:));
39  end
40
41  for(i=1:10)
42    if(Sum(i+1)<1)
43      a01=V_001(i);
44      i=-1;
45      break;
46    end
47  end
48
49  V_0001=[a01:0.001:a01+0.01];
50
51  for(i=1:10)
52    for(j=1:3)
53      y_0001(i,:)=approx((z.*K)./(1+(V_0001(i)*(K-1)))
            ,3);
54    end
55    Sum(i)=sum(y_0001(i,:));
56  end
57
58  for(i=1:10)
59    if(Sum(i+1)<1)
60      a001=V_0001(i);
61      i=-1;
62      break;
63    end
64  end
```

```
65 V=a001;
66 y_02=approx((z.*K)./(1+(a1*(K-1))),3);
67 y_03=approx((z.*K)./(1+((a1+0.1)*(K-1))),3);
68 y_0273=approx((z.*K)./(1+(V*(K-1))),3);
69 x_0273=approx(y_0273./K,3);
70
71 Ans=[z',K',y_02',y_03',y_0273',x_0273';1,0,sum(y_02)
      ,sum(y_03),sum(y_0273),approx(sum(x_0273),2)];
72 disp(Ans,'   z       K       y,V=0.2   y,V=0.3   y,V
      =0.273  x,V=0.273')
73 disp('NOTE : Last Row represents the summation')
74 disp('Hence for V = 0.273 E(yi) = 0 and E(xi) = 0')
75 disp(V,'Fraction of Vapor (V) = ')
76
77 //End
```

Chapter 11

Solution Thermodynamics Theroy

Scilab code Exa 11.4 Find the Expression For Enthalpies

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 11.4
11 //Caption : Program To Find the Expression For
       Enthalpies
12
13 //H=400x1 + 600x2 + x1x2(40x1 + 20x2)  Given
14
15 //Substituting x2=1-x1
16
```

```
17 H=poly([600 -180 0 -20],'x1','c')      //(A)
18
19 //K=dH/dx1
20
21 K=poly([-180 0 -60],'x1','c')
22
23 //Using Eqn 11.15  H1=H+x2*K
24
25 //substituting x2=1-x1
26
27 H1=poly([420 0 -60 40],'x1','c')      //(B)
28
29 //Similarly for H2
30
31 H2=poly([600 0 0 40],'x1','c')        //(C)
32
33 //Now to calculate H1_inf and H2_inf
34
35 //x1=0 in (B)
36 H1_inf=420;//[J/mol]
37
38 //x2=0 in (C) i.e. x1=1
39 H2_inf=640;//[J/mol]
40
41 disp(H1,'H1 = ');
42 disp(H2,'H2 = ');
43 disp('J/mol',H1_inf,'H1_inf = ')
44 disp('J/mol',H2_inf,'H2_inf = ')
45
46 //End
```

Scilab code Exa 11.5 Plot the Fugacity and Fugacity Coefficient Vs P

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 11.5
//Caption : Program to Plot the Fugacity and
   Fugacity Coefficient Vs P

//Using Eqn 11.30

//G=t(T)+RT ln f
//G'=t(T)+rt ln f'

//Hence ln(f/f') = (1/RT)*(G-G')

//G=H-TS
//G'=H-TS'

//ln (f/f')=(n/R)*(((H-H')/T)-(S-S'))    (A)

R=8.314;
n=18.015;
f_=1;//[kPa]
P_=f_;
H_=3076.8;//[J/g]
S_=10.3450;//[J/g/K]
T=573.15;//[K]
P=[10 50 100 200 500 1000 1500 2000 2500 3000 3500
   4000 4500 5000 5500 6000 6500 7000 7600 8000
   8400];
H=[3076.6 3075.7 3074.2 3072.1 3064.8 3052.1 3038.9
   3025 3010.4 2995.1 2979 2962 2944.2 2925.5 2905.8
    2885 2862.9 2839.4 2808.8 2786.8 2763.1];
```

```
34 S=[9.282 8.5380 8.2166 7.8937 7.4614 7.1251 6.9207
      6.7696 6.6470 6.5422 6.4491 6.3642 6.2852 6.2105
      6.1388 6.0692 6.0008 5.9327 5.8503 5.7942
      5.7366];
35
36 K=approx(exp((n/R).*(((H-H_)/T)-(S-S_))),0);
37
38 f=K*f_;
39 P(22)=8592.7;//[kPa]
40 P_sat=P(22)
41 f(22)=6738.9;
42 f_sat=f(22)
43
44 si=approx(f./P,4);
45 si_sat=si(22)
46
47 Vl=approx(1.403*n,2)//[cm^3/mol]
48 //Using Eqn 11.41
49 P_new=linspace(8592.7,10000,10);
50 f_new=approx(f_sat.*exp(Vl.*(P_new-P_sat)/(R*1000*T)
      ),1);
51 si_new=f_new./P_new;
52 subplot(1,2,1)
53 plot(P/1000,f/1000,'b')
54 plot(P_new/1000,f_new/1000,'g')
55 dotsx=[0 P_sat/1000];
56 dotsy=[f_sat/1000 f_sat/1000];
57 plot(dotsx,dotsy,'b--')
58 dotsx=[0 f_sat/1000];
59 dotsy=[P_sat/1000 P_sat/1000];
60 plot(dotsy,dotsx,'g--')
61 dotsx=[11,8];
62 dotsx=[6,6];
63 plot(dotsx,dotsy,'w')
64 legend('f vs P(till P_sat)','f vs P(Beyond P_sat)','
      f_sat','P_sat')
65 xtitle('f vs P','P X 10^-3 /kPa','f X 10^-3 /kPa')
66 subplot(1,2,2)
```

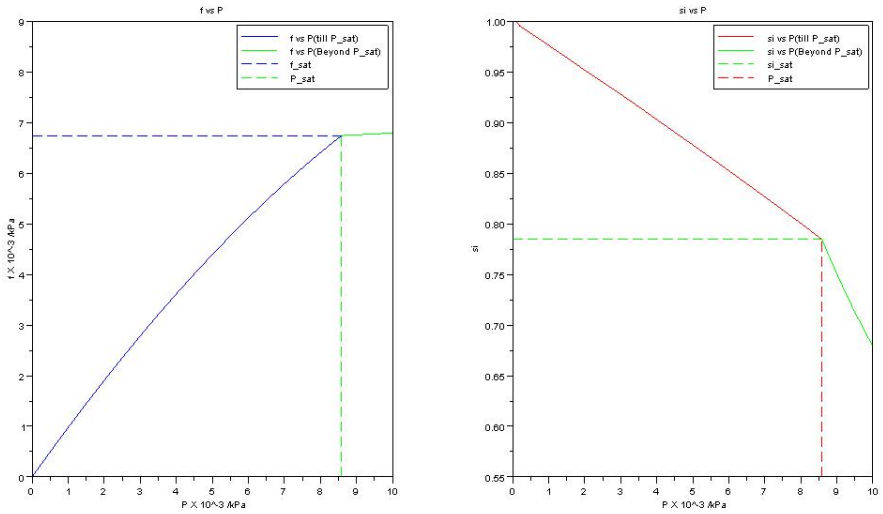

Figure 11.1: Plot the Fugacity and Fugacity Coefficient Vs P

```
67 plot(P/1000,si,'r')
68 plot(P_new/1000,si_new,'g')
69 dotsx=[0 P_sat/1000];
70 dotsy=[si_sat si_sat];
71 plot(dotsx,dotsy,'g--')
72 dotsx=[0.55 si_sat];
73 dotsy=[P_sat/1000 P_sat/1000];
74 plot(dotsy,dotsx,'r--')
75
76 legend('si vs P(till P_sat)','si vs P(Beyond P_sat)'
    ,'si_sat','P_sat')
77 xtitle('si vs P','P X 10^-3 /kPa','si')
78
79 //End
```

Scilab code Exa 11.7 Find the Fugacity Coefficient for the mixture

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 11.7
//Caption : Program to Find the Fugacity Coefficient
    for the mixture

T=200;//[K]
P=30;//[bar]
R=83.14;
x1=0.4;//[N2]
x2=1-x1;//[CH4]

B11=-35.2;//[cm^3/mol]
B22=-105;//[cm^3/mol]
B12=-59.8;//[cm^3/mol]

delta_12=approx((2*B12)-B11-B22,1);
si_1=approx(exp((P/(R*T))*(B11+(x2^2*delta_12))),4);
si_2=approx(exp((P/(R*T))*(B22+(x1^2*delta_12))),4);

B=approx((x1^2*B11)+(2*x1*x2*B12)+(x2^2*B22),2);
Z=approx(1+((B*P)/(R*T)),2);

```

```
disp(si_1,si_2,'Fugacity Coefficients are ')
disp(B,'Second Viral coefficient is ')
disp(Z,'Compressibility Factor is ')

//End
```

Scilab code Exa 11.8 Find the Fugacity of 1 butene vapor

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 11.8
//Caption : Program to Find the Fugacity of 1-butene
    vapor

T=473.15;//[K]
P=70;//[bar]
Tc=420;//[K]
Pc=40.43;//[bar]
omega=0.191;

//By interpolation in Tables E.15 and E.16
si_0=0.627;
si_1=1.096;
//Using Eqn(11.64)
si=approx(si_0*(si_1^omega),3);
f=approx(si*P,1);
```

```
25 disp(si,'Fugacity coefficient is ')
26 disp('bar',f,'fugacity is ')
27
28 //End
```

Scilab code Exa 11.9 Find the Fugacity Coefficients for mixture

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 11.9
11 //Caption : Program to Find the Fugacity
       Coefficients for mixture
12
13 P=25;//[KPa]
14 T=323.15;//[K]
15 R=83.14;
16 x1=0.5;
17 x2=1-x1;
18
19 ij=[11,22,12];
20 Tc_ij=[535.5,591.8,563.0];
21 Pc_ij=[41.5,41.1,41.3];
22 Vc_ij=[267,316,291];
23 Zc_ij=approx((Pc_ij.*Vc_ij)./(R*Tc_ij),3);
24 omega_ij=[0.323,0.262,0.293];
25
```

```
26 Tr_ij=approx(T./Tc_ij,3);
27 B0=approx(0.083-(0.422./(Tr_ij.^1.6)),3)
28 B1=approx(0.139-(0.172./(Tr_ij.^4.2)),3)
29 B_ij=round((R*Tc_ij./Pc_ij).*(B0+(omega_ij.*B1)));
30
31 delta_12=(2*B_ij(3))-B_ij(1)-B_ij(2);
32 R=8314;
33 si_1=approx(exp((P/(R*T))*(B_ij(1)+(x2^2*delta_12)))
      ,3);
34 si_2=approx(exp((P/(R*T))*(B_ij(3)+(x1^2*delta_12)))
      ,3);
35
36 Ans=[ij',Tc_ij',Pc_ij',Vc_ij',Zc_ij',omega_ij',Tr_ij
      ',B0',B1',B_ij'];
37 disp(Ans,'   ij      Tcij      Pcij    Vcij      Zcij
          Wij      Trij       B0        B1        Bij')
38
39 disp(si_1,si_2,'Fugacity Coefficients are ')
40
41 //End
```

Scilab code Exa 11.10 Find the Excess Properties for a mixture

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 11.10
```

```
11 //Caption : Program to Find the Excess Properties
      for a mixture
12
13 T0=298.15;//[K]
14 T=323.15;//[K]
15 Cp_E=-2.86;//[J/mol/K]
16 Ho_E=897.9;//[J/mol]
17 Go_E=384.5;//[J/mol]
18
19 //(a)  Derivations
20
21 //G_E=-a*(T ln T - T)+ bT + c
22 //S_E=a ln T - b
23 //H_E=aT + c
24
25 //Where
26
27 //a=Cp_E
28 //c=Ho_E-aT0
29 //b=((Go_E+a(T ln T0 - T0)-c)/T0)
30
31 //(b)
32 a=Cp_E;
33 c=approx(Ho_E-(a*T0),1);
34 b=approx((Go_E+(a*((T0*log(T0))-T0)-c))/T0,4);
35 G_E=approx((-a*(T*log(T)-T))+(b*T)+c,1);
36 S_E=approx((a*log(T))-b,3);
37 H_E=approx((a*T)+c,1);
38
39 disp('J/mol',G_E,'G_E = ')
40 disp('J/mol/K',S_E,'S_E = ')
41 disp('J/mol',H_E,'H_E = ')
42
43 //End
```

Chapter 12

Solution Thermodynamics Applications

Scilab code Exa 12.1 Reduce the set of VLE Data and Plot the Graphs

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 12.1
11 //Caption : Program to Reduce the set of VLE Data
       and Plot the Graphs
12
13 P
       =[90.15,91.78,88.01,81.67,78.89,76.82,73.39,66.45,62.95,57.70,50.
14 x1
```

```
    =[0.000,0.063,0.248,0.372,0.443,0.508,0.561,0.640,0.702,0.763,0.8
15 y1
    =[0.000,0.049,0.131,0.182,0.215,0.248,0.268,0.316,0.368,0.412,0.4
16 x2=1-x1;
17 y2=1-y1;
18 P1_sat=P(13);
19 P2_sat=P(1);
20 K=zeros(1,13);
21 for(i=1:13)
22   if(i ~= 1)
23     ln_V1(i)=approx(log(y1(i)*P(i)/(x1(i)*P1_sat))
         ,3);
24   end
25   if(i ~= 13)
26     ln_V2(i)=approx(log(y2(i)*P(i)/(x2(i)*P2_sat))
         ,3);
27   end
28 end
29 ln_V1(1)=%nan;
30 ln_V2(13)=%nan;
31 for(i=2:12)
32   K(i)=approx(((x1(i)*ln_V1(i))+(x2(i)*ln_V2(i)))/(
       x1(i)*x2(i)),3);   //K=G_E/(x1*x2*R*T)
33   k(i)=approx(((x1(i)*ln_V1(i))+(x2(i)*ln_V2(i))),3)
       ;   //K=G_E/(R*T)
34
35 end
36 K(1)=%nan;
37 k(1)=%nan;
38 K(13)=%nan;
39 k(13)=%nan;
40 A21=0.70;
41 A12=1.35;
42 K_new=approx((A21.*x1)+(A12.*x2),3);
43 //Using Eqn (12.10(a) and 12.10(b))
44 ln_V1_new=approx((x2.*x2).*(A12+(2*(A21-A12).*x1))
```

```
    ,3);
45 V1_new=approx(exp(ln_V1_new),3);
46 ln_V2_new=approx((x1.*x1).*(A21+(2*(A12-A21).*x2))
    ,3);
47 V2_new=approx(exp(ln_V2_new),3);
48 //Using Eqn (12.11)
49 P_new=(x1.*V1_new*P1_sat)+(x2.*V2_new*P2_sat);
50
51 A21_new=0.596;
52 A12_new=1.153;
53
54 K_new1=approx((A21_new.*x1)+(A12_new.*x2),3);
55 //Using Eqn (12.10(a) and 12.10(b))
56 ln_V1_new1=approx((x2.*x2).*(A12_new+(2*(A21_new-
    A12_new).*x1)),3);
57 V1_new1=approx(exp(ln_V1_new1),3);
58 ln_V2_new1=approx((x1.*x1).*(A21_new+(2*(A12_new-
    A21_new).*x2)),3);
59 V2_new1=approx(exp(ln_V2_new1),3);
60 //Using Eqn (12.11)
61 P_new1=(x1.*V1_new1*P1_sat)+(x2.*V2_new1*P2_sat);
62
63 subplot(1,2,1)
64 plot(x1,P,'bo')
65 plot(y1,P,'gs')
66
67 plot(x1,P_new,'b-')
68 plot(y1,P_new,'g-')
69
70 plot(x1,P_new1,'b--')
71 plot(y1,P_new1,'g--')
72
73 legend('Actual(P vs x1)','Actual(P vs y1)','By Gibbs
    Duhem(P vs x1)','By Gibbs Duhem(P vs y1)','By
    Barkers Method(Accurate)')
74
75 xtitle('(a)','x1,y1','P/kPa')
76
```

```
77 subplot(1,2,2)
78 plot(x1,ln_V1,'bs')
79 plot(x1,ln_V2,'gv')
80
81 plot(x1,K,'ro')
82
83 plot(x1,K_new,'r-')
84 plot(x1,ln_V1_new,'b-')
85 plot(x1,ln_V2_new,'g-')
86
87 plot(x1,K_new1,'r--')
88 plot(x1,ln_V1_new1,'b--')
89 plot(x1,ln_V2_new1,'g--')
90
91 legend('Actual(ln V1 vs x1)','Actual(ln V2 vs x1)','
     G_E/x1x2RT vs x1','By Gibbs Duhem(G_E/x1x2RT vs
     x1)','By Gibbs Duhem(ln V1 vs x1)','By Gibbs
     Duhem(ln V2 vs x1)','By Barkers Method(Accurate)'
     )
92 xtitle('(b)','x1')
93
94 //End
```

Scilab code Exa 12.2 Find the Excess Enthalpy as function of x1

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
```

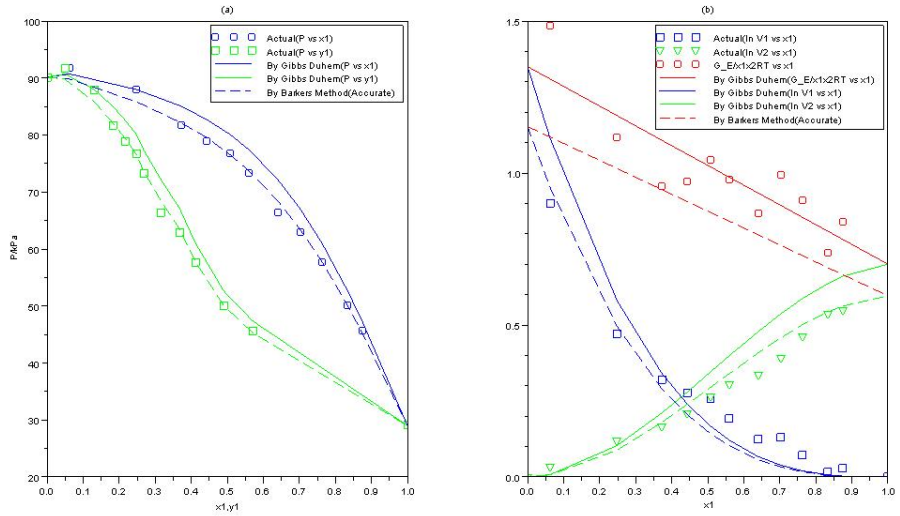

Figure 12.1: Reduce the set of VLE Data and Plot the Graphs

```
7    funcprot(0)
8  endfunction
9
10 //Example 12.2
11 //Caption : Program to Find the Excess Enthalpy as
       function of x1
12
13 //H_E=x1x2(40x1+20x2)    (A)
14 //Find H1_E and H2_E
15
16 //H1_E=H_E+((1-x1)*(dH_E/dx1))   let d=dH_E/dx1
17
18 //H2_E=H_E-(x1*(dH_E/dx1))   let d=dH_E/dx1
19
20 //Replacing x2=1-x1 in (A)
21
22 H_E=poly([0 20 0 -20],'x1','c');
23 d=poly([20 0 -60],'x1','c');
24 H1_E=poly([20 0 -60 40],'x1','c');
```

```
25 H2_E=poly([0 0 0 40],'x1','c');
26
27 disp(H1_E,'Expression For H1_E(x1) is ')
28 disp(H2_E,'Expression For H2_E(x1) is ')
29
30
31 //End
```

Scilab code Exa 12.4 Find the Heat of Formation of LiCl

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 12.4
11 //Caption : Program to Find the Heat of Formation of
       LiCl
12
13 //          Li + 0.5Cl2 --> LiCL(s)        (A)
14
15 //    LiCl(s) + 12H2O(l) --> LiCl(12H2O)    (B)
16
17 //Net Reaction
18 //Li + 0.5Cl2 +12H2O(l) --> LiCl(12H2O)    (Net)
19
20 //From Table C.4
21 Hf_A=-408610;//[J]
22 Hf_B=-33614;//[J]
```

```
23 Hf_Net=Hf_A+Hf_B;//[J]
24
25 disp('J',Hf_Net,'Heat of formation of LiCl in 12mol
     H2O at 298.15K is')
26
27 //End
```

Scilab code Exa 12.5 Calculate the Heat transfer rate in th evaporator

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 12.5
11 //Caption : Program to Calculate the Heat transfer
       rate in th evaporator
12
13 M_LiCl=42.39;
14 M_H2O=18.015;
15 T1=298.15;//[K]
16 T2=405.15;//[K]
17 //Step a
18 m_LiCl=0.15*2;
19 m_H2O=2-m_LiCl;
20 n_LiCl=(m_LiCl*1000)/M_LiCl
21 n_H2O=(m_H2O*1000)/M_H2O;
22 dH_LiCl=-33800;
23 dH_a=-n_LiCl*dH_LiCl//[J/s]
```

```
24
25 //Step b
26 m_LiCl=0.15*2;
27 m_H2O=0.45;
28 n_LiCl=(m_LiCl*1000)/M_LiCl;
29 n_H2O=(m_H2O*1000)/M_H2O;
30 dH_LiCl=-23260;
31 dH_b=n_LiCl*dH_LiCl//[J/s]
32
33 //Step c
34 m_LiCl=0.75;
35 Cp=2.72;
36 dT=T2-T1;
37 dH_c=m_LiCl*Cp*dT*1000//[J/s]
38
39 //step d
40 m_H2O=2-m_LiCl;
41 dH_T2=2740.3;//[KJ/s/mol] form Steam Tables
42 dH_T1=104.8;//[KJ/s/mol]  from Steam Tables
43 dH_d=m_H2O*(dH_T2-dH_T1)*1000//[J/s]
44
45 dH=approx((dH_a+dH_b+dH_c+dH_d)/1000,1);
46
47 disp('kW or KJ/s',dH,'The required Heat Transfer
     rate is ')
48
49 //End
```

Scilab code Exa 12.6 Calculate Heat Transfer Rate in a evaporator

```
1 clear all;
2 clc;
3
```

```
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
9
10 //Example 12.6
11 //Caption : Program to Calculate Heat Transfer Rate
     in Single effect Evaporator
12
13 T0=298.15;//[K]
14 T=361.5;//[K]
15 mT=1.25;//[Kg/s]  10% NaOH
16 m_steam=1;//[Kg/s] at P=76 torr and 361.5K
17 m_50NaOH=mT-m_steam;//[Kg/s]  at 361.5K
18
19 //From Steam tables
20 //at 76 torr and 361.15K
21 H_steam=2666;//[KJ/kg]
22 //for 10% NaOH soln at 294.15K
23 H_10NaOH=79;//[KJ/Kg]
24 //for 50% NaOH soln at 361.15K
25 H_50NaOH=500;//[KJ/Kg]
26
27 dH=(m_steam*H_steam)+(m_50NaOH*H_50NaOH)-(mT*
     H_10NaOH);
28 Q=dH;
29 disp('kW or kJ/s',Q,'Heat Transfer rate')
30
31 //End
```

Scilab code Exa 12.9 Calculate the Heat Transformed per Kg of Solution

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 12.9
//Caption : Program to Calculate the Heat
   Transformed per Kg of Solution formed

T=294.15;//[K]
m_NaOH_soln=1;//[kg]
m_NaOH_solid=0.45*m_NaOH_soln;//[Kg]
m_H2O=0.55*m_NaOH_soln;//[Kg]

//From Steam Tables
H_NaOH_soln=216;//[kJ/Kg]
H_NaOH_solid=1113;//[kJ/Kg]
H_H2O=88;//[kJ/Kg]

dH=m_NaOH_soln*H_NaOH_soln-(m_NaOH_solid*
   H_NaOH_solid)-(m_H2O*H_H2O);
Q=dH;

disp('kW or kJ/kg',Q,'Heat Transferred per Kg of
   NaOH Soln')

//End
```

Chapter 13

Chemical Reaction Equilibria

Scilab code Exa 13.1 Determine the Expressions for mole Fractions yi

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 13.1
11 //Caption : Program to determine the Expressions for
       mole Fractions yi
12
13 //CH4 + H2O --> CO + 3H2
14
15 n_CH4=2;//Moles of CH4
16 n_H2O=1;//Moles of H20
17 n_CO=1;//Moles of CO
18 n_H2=4;//Moles of H2
```

```
19
20 v_CH4=-1;
21 v_H2O=-1
22 v_CO=1;
23 v_H2=3;
24
25 v=v_CH4+v_H2O+v_CO+v_H2;
26 n=n_CH4+n_H2O+n_CO+n_H2;
27
28 //y_CH4=(n_CH4+(v_CH4e)/n+(v*e))
29 //y_H2O=(n_H2O+(v_H2Oe)/n+(v*e))
30 //y_CO=(n_CO+(v_CO*e)/n+(v*e))
31 //y_H2=(n_H2+(v_H2*e)/n+(v*e))
32
33 y_CH4='(n_CH4+(v_CH4e)/n+(v*e))';
34 y_H2O='(n_H2O+(v_H2Oe)/n+(v*e))';
35 y_CO='(n_CO+(v_CO*e)/n+(v*e))';
36 y_H2='(n_H2+(v_H2*e)/n+(v*e))';
37
38
39
40 //Hence
41
42 y_CH4='(2-e/8+2e)';
43 y_H2O='(1-e/8+2e)';
44 y_CO='(1+e/8+2e)';
45 y_H2='(4+3e/8+2e)';
46
47 disp(y_CH4,'y_CH4 = ')
48
49 disp(y_H2O,'y_H2O = ')
50
51 disp(y_CO,'y_CO = ')
52
53 disp(y_H2,'y_H2 = ')
54
55 //End
```

Scilab code Exa 13.3 Determine the Expression for yi for two reactions

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 13.3
//Caption : Program to Determine the Expression for
   yi for two reactions

//CH4 + H2O  --> CO + 3H2     (A)

//CH4 + 2H2O --> CO2 + 4H2    (B)

Species=['CH4','H2O','CO','CO2','H2','sum'];
v_A=['-1','-1','1','0','3','2'];
v_B=['-1','-2','0','1','4','2'];
m_CH4=2;
m_H2O=3;
mt=m_CH4+m_H2O;
y=['(m_CH4+vie1+vje2)/(mt+vie1+vje2)','(m_H2O+vie1+
   vje2)/(mt+vie1+vje2)','(vie1)/(mt+vie1+vje2)','(
   vje2)/(mt+vie1+vje2)','(vie1+vje2)/(mt+vie1+vje2)
   ',' '];

//Hence

```

```
27 yf=['(2-e1-e2)/(5+2e1+2e2)','(3-e1-2e2)/(5+2e1+3e2)'
   ,'e1/(5+2e1+2e2)','e2/(5+2e1+2e2)','(3e1+4e2)
   /(5+2e1+2e2)',' '];
28
29 Ans=[Species',v_A',v_B',y',yf'];
30
31 disp(Ans,' i    v_A   v_B       y(Before substitution)
            y_species')
32
33 //End
```

Scilab code Exa 13.4.a Alternate Program to 13 iv

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 13.4  Alternate
11 //Caption : Alternate Program to 13.4
12
13 T=[298.15,418.15,593.15];
14 t=approx(T/T(1),4);
15 R=8.314;
16
17
18 //C2H4(g) + H2O(g) -->  C2H5OH(g)
19
20 //Values From Table C.1 At T=298.15K
```

```
21
22 A_ethanol=3.518;
23 A_ethene=1.424;
24 A_water=3.470;
25
26 B_ethanol=20.001*10^-3;
27 B_ethene=14.394*10^-3;
28 B_water=1.450*10^-3;
29
30 C_ethanol=-6.002*10^-6;
31 C_ethene=-4.392*10^-6;
32 C_water=0;
33
34 D_ethanol=0;
35 D_ethene=0;
36 D_water=0.121*10^5;
37
38 dA=A_ethanol-A_ethene-A_water
39 dB=B_ethanol-B_ethene-B_water
40 dC=C_ethanol-C_ethene-C_water
41 dD=D_ethanol-D_ethene-D_water
42
43 // Values from Table C.4 at T=298.15K
44 H_ethanol=-235100;//[J/mol]
45 H_ethene=52510;//[J/mol]
46 H_water=-241572;//[J/mol]
47
48 G_ethanol=-168490;//[J/mol]
49 G_ethene=68460;//[J/mol]
50 G_water=-228572;//[J/mol]
51
52 dHo=H_ethanol-H_ethene-H_water
53 dGo=G_ethanol-G_ethene-G_water
54
55 //Using Eqn(13.21)
56 Ko=approx(exp(-dGo./(R*T(1))),3)
57 K0=Ko*ones(1,3);
58 //Using Eqn(13.22)
```

```
59 K1=exp((dHo/(R*T0))*(1-(T(1)./T)));
60 //Using Eqn(13.24)
61 K2=approx(exp((dA.*(log(t)-((t-1)./t)))+(0.5*dB*T(1)
      .*((t-1).^2)./t)+((1/6)*dC*T(1)*T(1).*((t-1).^2)
      .*(t+2)./t)+(0.5*dD.*((t-1).^2)./((T(1)^2).*(t)
      .^2))),4);
62
63 K=K0.*K1.*K2;
64
65 Ans=[T',t',K0',K1',K2',K'];
66 disp(Ans,'     T/K          t              K0              K1
              K2           K')
67
68 //End
```

Scilab code Exa 13.4 Find the equillibrium constant for Vapor Phase

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=IDCPH(T0,T,dA,dB,dC,dD)
11   t=T/T0;
12   Q=(dA+((dB/2)*T0*(t+1))+((dC/3)*T0*T0*((t^2)+t+1))
       +(dD/(t*T0*T0)))*(T-T0)
13   funcprot(0);
14 endfunction
15
```

```
16 function[Q]=IDCPS(T0,T,dA,dB,dC,dD)
17   t=T/T0;
18   Q=((dA)*log(t))+(((dB*T0)+(((dC*T0*T0)+(dD/(t*t*T0
       *T0)))*(t+1)/2))*(t-1))
19   funcprot(0);
20 endfunction
21
22 //Example 13.4
23 //Caption : Program to Find the equillibrium
     constant for Vapor Phase Hydration
24
25 T0=298.16;//[K]
26 T1=418.15;//[K]
27 T2=593.15;//[K]
28 R=8.314;
29
30 //C2H4(g) + H2O(g) -->  C2H5OH(g)
31 //Values From Table C.1 At T=298.15K
32
33 A_ethanol=3.518;
34 A_ethene=1.424;
35 A_water=3.470;
36
37 B_ethanol=20.001*10^-3;
38 B_ethene=14.394*10^-3;
39 B_water=1.450*10^-3;
40
41 C_ethanol=-6.002*10^-6;
42 C_ethene=-4.392*10^-6;
43 C_water=0;
44
45 D_ethanol=0;
46 D_ethene=0;
47 D_water=0.121*10^5;
48
49 dA=A_ethanol-A_ethene-A_water
50 dB=B_ethanol-B_ethene-B_water
51 dC=C_ethanol-C_ethene-C_water
```

```
52 dD=D_ethanol-D_ethene-D_water
53
54 // Values from Table C.4 at T=298.15K
55 H_ethanol=-235100;//[J/mol]
56 H_ethene=52510;//[J/mol]
57 H_water=-241572;//[J/mol]
58
59 G_ethanol=-168490;//[J/mol]
60 G_ethene=68460;//[J/mol]
61 G_water=-228572;//[J/mol]
62
63 dHo=H_ethanol-H_ethene-H_water
64 dGo=G_ethanol-G_ethene-G_water
65
66 I1=approx(IDCPH(T0,T1,dA,dB,dC,dD),3)
67 I2=approx(IDCPS(T0,T1,dA,dB,dC,dD),5)
68
69 //Using Eqn 13.18
70 //dG_418/RT=((dGo - dHo)/RTo)+(dHo/RT)+((1/T)*I1)-I2
       c1=dG_418/RT
71
72 c1=approx(((dGo-dHo)/(R*T0))+(dHo/(R*T1))+((1/T1)*I1
     )-I2,4)
73
74 I3=approx(IDCPH(T0,T2,dA,dB,dC,dD),3)
75 I4=approx(IDCPS(T0,T2,dA,dB,dC,dD),5)
76
77 //Using Eqn 13.18
78 //dG_593/RT=((dGo - dHo)/RTo)+(dHo/RT)+((1/T)*I1)-I2
       c2=dG_593/RT
79
80 c2=approx(((dGo-dHo)/(R*T0))+(dHo/(R*T2))+((1/T2)*I3
     )-I4,4)
81
82 K_413=approx(exp(-c1),4);
83 K_593=exp(-c2);
84
85 disp('X 10^-1',K_413*10,'Equilibrium Constant at T =
```

```
    413.15K is ')
disp('X 10^-3',approx(K_593*1000,3),'Equilibrium
   Constant at T = 593.15K is ')

//End
```

Scilab code Exa 13.5 Calculate the Fraction of Heat Reacted

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 13.5
//Caption : Program to Calculate the Fraction of
   Heat Reacted for Various Cases

//CO(g) + H2O(g) --> CO2(g) + H2(g)

v_CO=-1;
v_H2O=-1;
v_CO2=1;
v_H2=1;
v=v_CO+v_H2O+v_CO2+v_H2;

//Calculate e(Fraction of Stream) in each case

//(a)
n_H2O_a=1;//mol
```

```
25 n_CO_a=1;//mol
26 T_a=1100;//[K]
27 P_a=1;//[bar]
28
29 x=10^4/T_a;
30 //at this x the value of ln K=0 form Graph
31 y=0;
32 nt=n_H2O_a+n_CO_a;
33 K=exp(y);
34
35 //y_H2O=(n_H2O+(v_H2O*e))/nt
36 //y_CO=(n_CO+(v_CO*e))/nt
37 //y_H2=(v_H2*e)/nt
38 //y_CO2=(v_CO2*e)/nt
39
40
41 y_H2O='(1-e)/2'
42 y_CO='(1-e)/2'
43 y_H2='e/2'
44 y_CO2='e/2'
45
46
47 //K=(y_H2*y_CO2)/(y_CO*y_H2O)
48
49 K='e^2/((1-e)^2)=1';
50
51
52 //Solving
53 e_a=0.5;
54
55 //(b) same as in (a) P_b=10bar
56
57 //Pressure has no effect on fraction of stream
58 e_b=e_a;
59
60 //(c) same as in (a) N2=2mols included
61
62 //Since N2 just act as diluent
```

```
63 //It only changes the total moles fraction remains
      the same
64 e_c=e_a;
65
66 //(d)
67 n_H2O_d=2;//mol
68 n_CO_d=1;//mol
69 T_d=1100;//[K]
70 P_d=1;//[bar]
71
72 x=10^4/T_d;
73 //at this x the value of ln K=0 form Graph
74 y=0;
75 nt=n_H2O_d+n_CO_d;
76 K=exp(y);
77
78 //y_H2O=(n_H2O+(v_H2O*e))/nt
79 //y_CO=(n_CO+(v_CO*e))/nt
80 //y_H2=(v_H2*e)/nt
81 //y_CO2=(v_CO2*e)/nt
82
83 y_H2O='(1-e)/3'
84 y_CO='(2-e)/3'
85 y_H2='e/3'
86 y_CO2='e/3'
87
88 //K=(y_H2*y_CO2)/(y_CO*y_H2O)
89
90 K='e^2/((1-e)(2-e))=1';
91
92 //Solving
93 e=approx(2/3,3);
94 e_d=approx(e/2,3);
95
96 //(e)
97 //Here the y_CO and y_H2O are interchanged
98
99 //No change in Fraction of stream
```

```
100 e_e=e_d;
101
102 //(f)
103 n_H2O_f=1;//mol
104 n_CO_f=1;//mol
105 n_CO2_f=1;;//[mol]
106 T_f=1100;//[K]
107 P_f=1;//[bar]
108
109 x=10^4/T_f;
110 //at this x the value of ln K=0 form Graph
111 y=0;
112 nt=n_H2O_f+n_CO_f+n_CO2_f;
113 K=exp(y);
114
115 //y_H2O=(n_H2O+(v_H2O*e))/nt
116 //y_CO=(n_CO+(v_CO*e))/nt
117 //y_CO2=(n_CO2+(v_CO2*e))/nt
118 //y_H2=(v_H2*e)/nt
119
120
121 y_H2O='(1-e)/3'
122 y_CO='(1-e)/3'
123 y_H2='(1+e)/3'
124 y_CO2='e/2'
125
126
127 //K=(y_H2*y_CO2)/(y_CO*y_H2O)
128
129 K='(e*(e+1))/((1-e)^2)=1';
130
131 //Solving
132 e_f=approx(1/3,3);
133
134 //(g)
135 n_H2O_g=1;//mol
136 n_CO_g=1;//mol
137 T_g=1650;//[K]
```

```
P_g=1;//[bar]

x=10^4/T_g;
//at this x the value of ln K=0 form Graph
y=-1.15;
nt=n_H2O_g+n_CO_g;
K=exp(y);

//y_H2O=(n_H2O+(v_H2O*e))/nt
//y_CO=(n_CO+(v_CO*e))/nt
//y_H2=(v_H2*e)/nt
//y_CO2=(v_CO2*e)/nt


y_H2O='(1-e)/2'
y_CO='(1-e)/2'
y_H2='e/2'
y_CO2='e/2'


//K=(y_H2*y_CO2)/(y_CO*y_H2O)

Exp='e^2/((1-e)^2)=0.316';

//Solving
p=poly([K -2*K K-1],'e','c');

root=roots(p);
e_g=approx(root(1),2);

//Other Root is negative and the Fraction of steam
   cannot be negative


disp('(a)1mol H2O and 1 mol CO T=1100K  P=1bar')
disp('(b)1mol H2O and 1 mol CO T=1100K  P=10bar')
disp('(c)1mol H2O and 1 mol CO 2mol N2 T=1100K  P=1
   bar')
```

```
174 disp('(d)2mol H20 and 1 mol CO T=1100K  P=1bar')
175 disp('(e)1mol H20 and 2 mol CO T=1100K  P=1bar')
176 disp('(f)1mol H20 and 1 mol CO 1mol CO2 T=1100K  P=1
      bar')
177 disp('(g)1mol H20 and 1 mol CO T=1650K  P=1bar')
178 e=[e_a e_b e_c e_d e_e e_f e_g];
179 disp(e,'Fraction Of Steam Reacted in each case')
180
181 //End
```

Scilab code Exa 13.6 Find the Maximum Conversion of Ethylene to ethanol

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=IDCPH(T0,T,dA,dB,dC,dD)
11   t=T/T0;
12   Q=(dA+((dB/2)*T0*(t+1))+((dC/3)*T0*T0*((t^2)+t+1))
       +(dD/(t*T0*T0)))*(T-T0)
13   funcprot(0);
14 endfunction
15
16 function[Q]=IDCPS(T0,T,dA,dB,dC,dD)
17   t=T/T0;
18   Q=((dA)*log(t))+(((dB*T0)+(((dC*T0*T0)+(dD/(t*t*T0
       *T0)))*(t+1)/2))*(t-1))
19   funcprot(0);
```

```
20 endfunction
21
22 function[si]=PHIB(Tr,Pr,omega)
23   B0=0.083-(0.422/(Tr^1.6));
24   B1=0.139-(0.172/(Tr^4.2));
25   si=exp((Pr/Tr)*(B0+(omega*B1)));
26   funcprot(0);
27
28 endfunction
29
30
31 //Example 13.6
32 //Caption : Program to find the Maximum Conversion
      of Ethylene to ethanol
33
34 T0=298.16;//[K]
35 T1=523.15;//[K]
36 P=35;//[bar]
37 R=8.314;
38
39 //C2H4(g) + H2O(g) -->  C2H5OH(g)
40 //Values From Table C.1 At T=298.15K
41
42 A_ethanol=3.518;
43 A_ethene=1.424;
44 A_water=3.470;
45
46 B_ethanol=20.001*10^-3;
47 B_ethene=14.394*10^-3;
48 B_water=1.450*10^-3;
49
50 C_ethanol=-6.002*10^-6;
51 C_ethene=-4.392*10^-6;
52 C_water=0;
53
54 D_ethanol=0;
55 D_ethene=0;
56 D_water=0.121*10^5;
```

```
57
58 dA=A_ethanol-A_ethene-A_water
59 dB=B_ethanol-B_ethene-B_water
60 dC=C_ethanol-C_ethene-C_water
61 dD=D_ethanol-D_ethene-D_water
62
63 // Values from Table C.4 at T=298.15K
64 H_ethanol=-235100;//[J/mol]
65 H_ethene=52510;//[J/mol]
66 H_water=-241572;//[J/mol]
67
68 G_ethanol=-168490;//[J/mol]
69 G_ethene=68460;//[J/mol]
70 G_water=-228572;//[J/mol]
71
72 dHo=H_ethanol-H_ethene-H_water
73 dGo=G_ethanol-G_ethene-G_water
74
75 I1=approx(IDCPH(T0,T1,dA,dB,dC,dD),3)
76 I2=approx(IDCPS(T0,T1,dA,dB,dC,dD),5)
77
78 //Using Eqn 13.18
79 //dG_418/RT=((dGo - dHo)/RTo)+(dHo/RT)+((1/T)*I1)-I2
       c1=dG_418/RT
80
81 c1=approx(((dGo-dHo)/(R*T0))+(dHo/(R*T1))+((1/T1)*I1
     )-I2,4)
82 K_523=approx(exp(-c1),4);
83 disp('X 10^-3',approx(K_523*1000,3),'Equilibrium
     Constant at T = 523.15K is ')
84
85 //Values Frm App B
86 Tc=[282.3,647.1,513.9];
87 Pc=[50.4,220.55,61.48];
88 omega=[0.087,0.345,0.645];
89
90 Tr=approx(T1./Tc,3);
91 Pr=approx(P./Pc,3);
```

```
92 si=approx(PHIB(Tr,Pr,omega),3);
93
94 //Using eqn
95 //(y_ETOH*si_ETOH)/(y_C2H4*si_C2H4*y_H20*si_H2O)=(P/
     Po)K
96 //y_ETOH/(y_C2H4*y_H20)=c=((si_C2H4*si_H2O)/si_ETOH)
     (P/Po)K
97 c=approx(((si(1)*si(2))/si(3))*(P*K_523),3)
98
99 //y_C2H4 = (1-e)/(6-e)
100 //y_ETOH = (5-e)/(6-e)
101 //y_H2O = (e)/(6-e)
102
103 //Solving we get a Eqn
104 poly([1.342 -6 1],'e','c')
105 root=approx(roots(poly([1.342 -6 1],'e','c')),3)
106
107 r=root(1)*100;
108 //Since e > 1 not possible so e=0.233
109
110 disp('%',r,'The Maximum Conversion of ethylene to
     ethanol by Vapor-Phase Hydration is ')
111
112 //End
```

Scilab code Exa 13.7 Find the Composition of Product Steam

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
```

```
7    funcprot(0)
8  endfunction
9
10 //Example 13.7
11 //Caption : Program to Find the Composition of
       Product Steam at Equillibrium
12
13 T=1393.15;//K
14 P=1;//[bar]
15 x=10^4/T;
16 //C2H2 --> 2C + H2  (I)
17 //2C + 2H2 --> C2H4 (II)
18
19 //Values Of ln K (at  1000/T )for Reactions I and II
       from Graph
20 K_I=exp(12.9);
21 K_II=exp(-12.9);
22 K=K_I*K_II;
23
24 //Application in Eqn (13.5)
25
26 //y_C2H4/(y_C2H2*y_H2)=c=(P/Po)K
27 c=P*K;
28 //y_H2=y_C2H2=(1-e)/(2-e)
29 //y_C2H4=e/(2-e)
30
31
32 //The Eqn comes out to be
33 poly([1 -4 2],'e','c')
34
35 root=approx(roots(poly([1 -4 2],'e','c')),3)
36 e=root(1);
37 //Since e > 1 not possible so e=0.293
38 y_C2H2=approx((1-e)/(2-e),3);
39 y_H2=y_C2H2;
40 y_C2H4=approx(e/(2-e),3);
41
42 disp(y_H2,y_C2H2,y_C2H4,'Equilibrium Composition of
```

```
    H2 C2H2 and C2H4 Respectively ')
43
44 //End
```

Scilab code Exa 13.8 Find the Mole fraction of Ethyl Acetate

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 13.8
11 //Caption : Program to Find the Mole fraction of
       Ethyl Acetate
12
13 T0=298.15;
14 T=373.15;//[K]
15 R=8.314;
16 //CH3COOH(l)+C2H5OH(l) --> CH3COOC2H5(l) + H2O(l)
17
18 //From Table C.4
19 dHo_EtAc=-480000;//[J]
20 dHo_H2O=-285830;//[J]
21 dHo_EtOH=-277690;//[J]
22 dHo_AcH=-484500;//[J]
23
24 dGo_EtAc=-332200;//[J]
25 dGo_H2O=-237130;//[J]
26 dGo_EtOH=-174780;//[J]
```

```
27 dGo_AcH=-389900;//[J]
28
29 dHo_298=dHo_EtAc+dHo_H2O-dHo_EtOH-dHo_AcH;
30 dGo_298=dGo_EtAc+dGo_H2O-dGo_EtOH-dGo_AcH;
31
32 K_298=approx(exp(-dGo_298/(R*T0)),4);
33
34 //Using Eqn(13.15)
35 //ln(K_373/K_298)=c=-(dHo_298/R)*((1/373.15)
      -(1/298.15))
36 c=approx(-(dHo_298/R)*((1/373.15)-(1/298.15)),4);
37 K_373=approx(K_298*exp(c),4);
38
39 //x_AcH=x_EtOH=(1-e)/2  and x_EtAc=x_H2O=e/2
40 //K=(x_EtAc*x_H2O)/(x_AcH*x_EtOH)
41
42 //Hence The Eqn is
43 q=poly([K_373 -2*K_373 K_373-1],'e','c')
44 root=approx(roots(q),4)
45 e=root(1);
46 //Since Other Root is > 1 hence e=0.6879
47 x_EtAc=approx(e/2,3);
48
49 disp(x_EtAc,'Composition of Ethyl Acetate in the
      Reacting Mixture')
50
51 //End
```

Scilab code Exa 13.9 Determine Composition and Temperature of Steam

```
1 clear all;
2 clc;
3
```

```
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=MCPH(T0,T,A,B,C,D)
11   t=T/T0;
12   Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
       /(t*T0*T0)))
13   funcprot(0);
14 endfunction
15
16 function[Q]=IDCPH(T0,T,dA,dB,dC,dD)
17   t=T/T0;
18   Q=(dA+((dB/2)*T0*(t+1))+((dC/3)*T0*T0*((t^2)+t+1))
       +(dD/(t*T0*T0)))*(T-T0)
19   funcprot(0);
20 endfunction
21
22 function[Q]=IDCPS(T0,T,dA,dB,dC,dD)
23   t=T/T0;
24   Q=((dA)*log(t))+(((dB*T0)+(((dC*T0*T0)+(dD/(t*t*T0
       *T0)))*(t+1)/2))*(t-1))
25   funcprot(0);
26 endfunction
27
28 //Example 13.9
29 //Caption : Program to Determine Composition and
     Temperature of Product Steam
30
31 P=1;//[bar]
32 T0=298.15;//[K]
33 R=8.314;
34
35 //SO2 + 0.5O2 --> SO3
36 dHo_298=-98890;//[J/mol]
37 dGo_298=-70866;//[J/mol]
```

```
38
39 n_O2_i=0.5*1.2;//  Moles O2 Entering
40 n_N2_i=n_O2_i*(79/21);//Moles N2 Entering
41
42 //n_SO2=1-e
43 //n_O2=0.6-(0.5*e)
44 //n_SO3=e
45 //n_N2=2.257
46
47 //By Energy Balance
48 //(dHo_298*e)+dHo_P = dH = 0
                                        (A)
49
50 //dHo_P=Cp*(T-298.15)  Cp=E nCp
                                    (B)
51
52 //Cp_SO2=R*MCPH(T0,T,5.699,0.801E-3,0,-1.015E+5)
53 //Cp_O2=R*MCPH(T0,T,3.639,0.506E-3,0,-0.227E+5)
54 //Cp_SO3=R*MCPH(T0,T,8.06,1.056E-3,0,-2.028E+5)
55 //Cp_N2=R*MCPH(T0,T,3.28,0.593E-3,0,0.04E+5)
56
57 //T=(-(dHo_298*e)/Cp)+T0
                                          (C)
58
59 //K=(e/(1-e))*((3.857-(0.5*e))/(0.6-(0.5*e)))^0.5
                    (D)
60
61 //ln K = ((dHo_298-dGo_298)/(R*T0))-(dHo_298/(R*T))+
      I1-(I2/T)   (E)
62
63 //I1=IDCPS(T0,T,0.5415,0.002E-3,0,-0.8995E+5)
64 //I2=IDCPH(T0,T,0.5415,0.002E-3,0,-0.8995E+5)
65
66 //Iteration
67 A1=300;//Initial
68 i=-1;
69
70 while(i==-1)
```

```
71
72      I1=IDCPS(T0,A1,0.5415,0.002E-3,0,-0.8995E+5);
73      I2=IDCPH(T0,A1,0.5415,0.002E-3,0,-0.8995E+5);
74      //Applying in Eqn (E)
75      K = exp(((dHo_298-dGo_298)/(R*T0))-(dHo_298/(R*
           A1))+I1-(I2/A1));
76      //Applying in Eqn (D)
77      if(isreal(K))
78        x=0;
79        // p=poly([-0.6*(K^2) 1.7*(K^2) 3.857-(1.6*(K
             ^2)) 0.5*((K^2)-1)],'e','c')
80        // (0.5*((K^2)-1)*(x^3))+((3.857-(1.6*(K^2)))
             *(x^2))+(1.7*(K^2)*x)+(-0.6*(K^2))
81        F_x=(0.5*((K^2)-1)*(x^3))+((3.857-(1.6*(K^2)))
             *(x^2))+(1.7*(K^2)*x)+(-0.6*(K^2));
82        F_a=F_x;
83
84        x=1;
85        F_x=(0.5*((K^2)-1)*(x^3))+((3.857-(1.6*(K^2)))
             *(x^2))+(1.7*(K^2)*x)+(-0.6*(K^2));
86        //F_x=(x^3)-(4*x)+1;
87        F_b=F_x;
88        root=-100;
89        A=0;
90        B=1;
91        i=1;
92        while(i==1)
93         a=A;
94         F_a=(0.5*((K^2)-1)*(a^3))+((3.857-(1.6*(K^2))
              )*(a^2))+(1.7*(K^2)*a)+(-0.6*(K^2));
95         //F_a=(a^3)-(4*a)+1;
96         b=B;
97         F_b=(0.5*((K^2)-1)*(b^3))+((3.857-(1.6*(K^2))
              )*(b^2))+(1.7*(K^2)*b)+(-0.6*(K^2));
98         //F_b=(b^3)-(4*b)+1;
99         x1=((a*F_b)-(b*F_a))/(F_b-F_a);
100        F_x1=(0.5*((K^2)-1)*(x1^3))+((3.857-(1.6*(K
              ^2)))*(x1^2))+(1.7*(K^2)*x1)+(-0.6*(K^2));
```

```
101         //F_x1=(x1^3)-(4*x1)+1;
102
103         if((F_a*F_x1)<0) then
104           flag=1;
105           A=a;
106           B=x1;
107         else((F_x1*F_b)<0)
108          flag=2;
109          A=x1;
110          B=b;
111       end
112       x1_a=approx(x1,4);
113       b_a=approx(b,4);
114       a_a=approx(a,4);
115       if(x1_a==b_a)
116         root=approx(x1,5);
117         i=0;
118         break;
119       elseif(x1_a==a_a)
120         root=approx(x1,5);
121         i=0;
122         break;
123       end
124
125     end
126     e=root;
127     Cp_SO2=R*MCPH(T0,A1,5.699,0.801E-3,0,-1.015E+5);
128     Cp_O2=R*MCPH(T0,A1,3.639,0.506E-3,0,-0.227E+5);
129     Cp_SO3=R*MCPH(T0,A1,8.06,1.056E-3,0,-2.028E+5);
130     Cp_N2=R*MCPH(T0,A1,3.28,0.593E-3,0,0.04E+5);
131
132     n_SO2=1-e;
133     n_O2=0.6-(0.5*e);
134     n_SO3=e;
135     n_N2=2.257;
136     if(n_SO2<0 | n_O2<0 | n_SO3<0)
137       e=0;
138     end
```

```
139      Cp=(n_SO2*Cp_SO2)+(n_O2*Cp_O2)+(n_SO3*Cp_SO3)+(
            n_N2*Cp_N2);
140      //Applying in Eqn (C)
141      B=(-(dHo_298*e)/Cp)+T0;
142      m=(A1+B)/2;
143      dT=approx(abs(m-A1),2);
144      if(dT<0.1)
145        i=0;
146        T=approx(A1,1);
147        e=approx(e,2);
148        break;
149     end
150      A1=m;
151      i=-1;
152    else
153      i=-1;
154      A1=A1+1;
155    end
156  end
157  disp(e,'Fraction')
158
159  n_SO2=1-e
160  n_O2=0.6-(0.5*e)
161  n_SO3=e
162  n_N2=2.257
163
164  nt=n_SO2+n_O2+n_SO3+n_N2;
165
166  y_SO2=approx(n_SO2/nt,4);
167  y_O2=approx(n_O2/nt,4);
168  y_SO3=approx(n_SO3/nt,4);
169  y_N2=approx(n_N2/nt,4);
170
171  disp(T,'Final Temperature')
172  disp(y_SO2,'Composition of SO2')
173  disp(y_O2,'Composition of O2')
174  disp(y_SO3,'Composition of SO3')
175  disp(y_N2,'Composition of N2')
```

```
176
177 //End
```

Scilab code Exa 13.12 Find the Product Composition for Two Reactions

```
1  clear all;
2  clc;
3
4
5  //To find Approx Value
6  function[A]=approx(V,n)
7    A=round(V*10^n)/10^n;//V-Value  n-To what place
8    funcprot(0)
9  endfunction
10
11 function[Q]=IDCPH(T0,T,dA,dB,dC,dD)
12   t=T/T0;
13   Q=(dA+((dB/2)*T0*(t+1))+((dC/3)*T0*T0*((t^2)+t+1))
       +(dD/(t*T0*T0)))*(T-T0)
14   funcprot(0);
15 endfunction
16
17 function[Q]=IDCPS(T0,T,dA,dB,dC,dD)
18   t=T/T0;
19   Q=((dA)*log(t))+(((dB*T0)+(((dC*T0*T0)+(dD/(t*t*T0
       *T0)))*(t+1)/2))*(t-1))
20   funcprot(0);
21 endfunction
22
23
24 //Example 13.12
25 //Caption : Program to Find the Product Composition
     for Two Reactions
```

```
26
27 T0=298.16;//[K]
28 T1=750;//[K]
29 R=8.314;
30 P=1.2;//[bar]
31
32 //C4H10 --> C2H4 + C2H6   (I)
33 //C4H10 --> C3H6 + CH4    (II)
34
35 //Values From Table C.1 At T=298.15K
36
37 A_butane=1.935;
38 A_ethene=1.424;
39 A_ethane=1.131;
40 A_propene=1.637;
41 A_methane=1.702;
42
43 B_butane=36.915*10^-3;
44 B_ethene=14.394*10^-3;
45 B_ethane=19.225*10^-3;
46 B_propene=22.706*10^-3;
47 B_methane=9.081*10^-3;
48
49 C_butane=-11.402*10^-6;
50 C_ethene=-4.392*10^-6;
51 C_ethane=-5.561*10^-6;
52 C_propene=-6.915*10^-6;
53 C_methane=-2.164*10^-6;
54
55 D_butane=0;
56 D_ethene=0;
57 D_ethane=0;
58 D_propene=0;
59 D_methane=0;
60
61 dA_I=A_ethene+A_ethane-A_butane;
62 dA_II=A_methane+A_propene-A_butane;
63
```

```
64  dB_I=B_ethene+B_ethane-B_butane;
65  dB_II=B_methane+B_propene-B_butane;
66
67  dC_I=C_ethene+C_ethane-C_butane;
68  dC_II=C_methane+C_propene-C_butane;
69
70  dD_I=D_ethene+D_ethane-D_butane;
71  dD_II=D_methane+D_propene-D_butane;
72
73  // Values from Table C.4 at T=298.15K
74  H_butane=-125790;//[J/mol]
75  H_ethene=52510;//[J/mol]
76  H_ethane=-83820;//[J/mol]
77  H_propene=19710;//[J/mol]
78  H_methane=-74520;//[J/mol]
79
80  G_butane=-16570;//[J/mol]
81  G_ethene=68460;//[J/mol]
82  G_ethane=-31855;//[J/mol]
83  G_propene=62205;//[J/mol]
84  G_methane=-50460;//[J/mol]
85
86  dHo_I=H_ethene+H_ethane-H_butane
87  dHo_II=H_methane+H_propene-H_butane
88
89  dGo_I=G_ethene+G_ethane-G_butane
90  dGo_II=G_methane+G_propene-G_butane
91
92  I1_I=approx(IDCPH(T0,T1,dA_I,dB_I,dC_I,dD_I),3)
93  I1_II=approx(IDCPH(T0,T1,dA_II,dB_II,dC_II,dD_II),3)
94  I2_I=approx(IDCPS(T0,T1,dA_I,dB_I,dC_I,dD_I),5)
95  I2_II=approx(IDCPS(T0,T1,dA_II,dB_II,dC_II,dD_II),5)
96
97  //Using Eqn 13.18
98  //dG_418/RT=((dGo - dHo)/RTo)+(dHo/RT)+((1/T)*I1)-I2
        c1=dG_418/RT
99
100 c1_I=approx(((dGo_I-dHo_I)/(R*T0))+(dHo_I/(R*T1))
```

```
    +((1/T1)*I1_I)-I2_I,4)
c1_II=approx(((dGo_II-dHo_II)/(R*T0))+(dHo_II/(R*T1)
    )+((1/T1)*I1_II)-I2_II,4)

K_I=approx(exp(-c1_I),4)
K_II=approx(exp(-c1_II),4)

k=(K_II/K_I)^0.5;
e_I=approx(((K_I/P)/(1+(K_I*(1/P)*(1+k)*(1+k))))
    ^0.5,4);

e_II=approx(k*e_I,4);

n_C4H10=1-e_I-e_II;
n_C2H4=e_I;
n_C2H6=e_I;
n_C3H6=e_II;
n_CH4=e_II;
nt=n_C4H10+n_C2H4+n_C2H6+n_C3H6+n_CH4;

y_C4H10=approx(n_C4H10/nt,4);
y_C2H4=approx(n_C2H4/nt,4);
y_C2H6=approx(n_C2H6/nt,4);
y_C3H6=approx(n_C3H6/nt,4);
y_CH4=approx(n_CH4/nt,4);

y=[y_C4H10 y_C2H4 y_C2H6 y_C3H6 y_CH4];
disp(y,'  Y_C4H10    Y_C2H4     Y_C2H6     Y_C3H6
    Y_CH4')

//End
```

Scilab code Exa 13.13 Find the Composition at different Temperatures

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 13.13
//Caption : Program to Find the Composition at
   different Temperatures

n_air=2.381//[mol]
n_O2=0.21*n_air;
n_N2=0.79*n_air;
R=8.314;

P=20;//[bar]
T=[1000 1100 1200 1300 1400 1500];
dG_H2O=[-192420 -187000 -181380 -175720 -170020
   -164310];
dG_CO=[-200240 -209110 -217830 -226530 -235130
   -243740];
dG_CO2=[-395790 -395960 -396020 -396080 -396130
   -396160];

KI='y_H2O/((y_O2)^0.5*y_H2)(P/Po)^-0.5'
KII='y_CO/((y_O2)^0.5)(P/Po)^0.5'
KIII='y_CO2/y_O2'

n='3.38+((e2-e1)/2)'
y_H2='-e1/n'
y_CO='e2/n'
y_O2='((0.5(1-e1-e2))-e3)/n'
y_H2O='(1+e1)/n'
y_CO2='e3/n'
y_N2='1.88/n'
```

```
35
36 KI='(1+e1)(2n)^0.5*(P/Po)^-0.5'
37 KII='(e3*(P/Po)^0.5)/(1-e1-e2-2e3)^0.5*(n/2)^0.5'
38 KIII='2e3/(1-e1-e2-2e3)'
39
40 K_I=approx(exp(-dG_H2O./(R.*T)),1)
41 K_II=approx(exp(-dG_CO./(R.*T)),1)
42 K_III=approx(exp(-dG_CO2./(R.*T)),1)
43
44 //Now since the values of KI KII KIII valyes are so
      High the mole fraction of O2 must be very small
45 //Hence We eleminate O2,Hence 2 Eqns are,
46
47 //C + CO2 --> 2CO       (a)
48 //H2O + C --> H2 + CO   (b)
49
50 Ka='(y_CO^2/y_CO2)*(P/Po)'
51 Kb='((y_H2*y_CO)/y_H2O)*(P/Po)'
52
53 n='3.38+(e_a+e_b)'
54 y_H2='e_b/n'
55 y_CO='(2e_a+e_b)/n'
56 y_H2O='(1-e_b)/n'
57 y_CO2='(0.5-e_a)/n'
58 y_N2='1.88/n'
59
60 Ka='(2e_a+e_b)^2/((0.5-e_a)*n)*(P/Po)'
61 Kb='e_b(2e_a+e_b)/((1-e_b)*n)*(P/Po)'
62
63 dG_new_a=(2*dG_CO)-dG_CO2;
64 dG_new_b=dG_CO-dG_H2O;
65
66 Ka=approx(exp((-dG_new_a./(R.*T))),3);
67 Kb=approx(exp((-dG_new_b./(R.*T))),3);
68
69 //Calculation of e_a and e_b
70
71 a=0.1;//Initial Value
```

```
72
73 b=0.7;//Initial Value
74
75 C1=Ka/20;
76 C2=Kb/20;
77
78 for(i=1:6)
79 c=-1;
80 while(c==-1)
81   fa=approx((((a^2)*(4+C1(i)))+(b^2)+((4+C1(i))*(a*b
       ))+(2.88*C1(i)*a)-(0.5*C1(i)*b)-(1.69*C1(i)))
       ,4);
82   dfax=approx(((2*a*(4+C1(i)))+((4+C1(i))*b)+(2.88*
       C1(i))),4);
83   dfay=approx((2*b)+((4+C1(i))*a)-(0.5*C1(i)),4);
84
85   fb=approx(((b^2*(1+C2(i)))+((2+C2(i))*a*b)-(C2(i)*
       a)+(2.38*C2(i)*b)-(3.38*C2(i))),4);
86   dfbx=approx((((2+C2(i))*b)-C2(i)),4);
87   dfby=approx(((2*b*(1+C2(i)))+((2+C2(i))*a)+(2.38*
       C2(i))),4);
88
89   A=[dfax dfay;dfbx dfby];
90   B=[-fa;-fb];
91   Ans=approx(A\B,4);
92   da=Ans(1);
93   db=Ans(2);
94
95   if(da==0 & db==0)
96     c=0;
97     e_a(i)=a;
98     e_b(i)=b;
99     break;
100  end
101  a=a+da;
102  b=b+db;
103 end
104 end
```

```
105
106 n=3.38+(e_a+e_b);
107 y_H2=approx(e_b./n,3);
108 y_CO=approx(((2*e_a)+e_b)./n,3);
109 y_H2O=approx((1-e_b)./n,3);
110 y_CO2=approx((0.5-e_a)./n,3);
111 y_N2=approx(1.88./n,3);
112
113 Ans=[T',Ka',Kb',e_a,e_b];
114 Ans1=[T',y_H2,y_CO,y_H2O,y_CO2,y_N2];
115
116 plot(T',y_H2,'r-')
117 plot(T',y_CO,'b-')
118 plot(T',y_H2O,'g-')
119 plot(T',y_CO2,'m-')
120 plot(T',y_N2,'y-')
121
122 legend('H2','CO','H2O','CO2','N2',)
123 xtitle('Equllibrium Compositions','T/K','yi')
124
125
126
127 disp(Ans,'     T/K          Ka            Kb            e_a
          e_b')
128
129 disp(Ans1,'     T/K          y_H2       y_CO      y_H2O
    y_CO2      y_N2')
130
131 //End
```

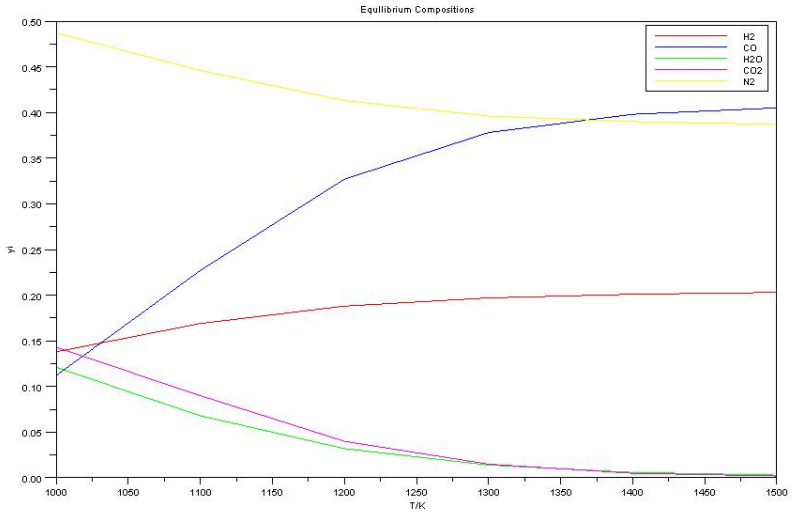

Figure 13.1: Find the Composition at different Temperatures

Chapter 14

Topics In Phase Equilibria

Scilab code Exa 14.1 Find the Fugacity Coefficients of N2 and CH4

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 14.1
//Caption : Program to Find the Fugacity
   Coefficients of Nitrogen and Methane

Tc=[126.2 190.6];
T=200;//[K]
Tr=T./Tc;
Pc=[34 45.99];
P=30;//[Bar]
Pr=P./Pc;
```

```
R=83.14;
bi=approx(0.08664.*R.*Tc./Pc,3);
ai=approx(((0.42748.*(R^2).*(Tc.^2).*(Tr.^(-0.5))./
   Pc)/(10^5)),3);   //10^-5ai
y=[0.4  0.6];
a=approx(((y(1)^2)*ai(1))+(2*y(1)*y(2)*sqrt((ai(1)*
   ai(2))))+((y(2)^2)*ai(2)),2);   //10^-5a
b=approx((y(1)*bi(1))+(y(2)*bi(2)),3);

q=(a*10^5)/(b*R*T);
Beta=0.051612;
//Z=1+Beta-(q*(Z-Beta)/(Z*(Z+Beta)))

Z=1;//initial
A=Z;
for i=0:10
  B=1+Beta-((q*Beta)*(A-Beta)/(A*(A+Beta)));
   if((B-A)==0.0001)
     break;
   end
   A=B;
   i=i+1;
end
Z=approx(B,5);

I=log((Z+Beta)/Z)
a1=approx((2*y(1)*ai(1))+(2*y(2)*sqrt((ai(1)*ai(2)))
   -a),2);    //10^-5a1
a2=approx((2*y(2)*ai(2))+(2*y(1)*sqrt((ai(2)*ai(1)))
   -a),2);    //10^-5a2
b1=bi(1);
b2=bi(2);
q1=approx(q*(((a1+a)/a)-(b1/b)),5);
q2=approx(q*(((a2+a)/a)-(b2/b)),5);
ln_si1=approx((((b1/b)*(Z-1))-(log(Z-Beta))-(q1*I))
   ,5);    //ln si1
ln_si2=approx((((b2/b)*(Z-1))-(log(Z-Beta))-(q2*I))
   ,5);    //ln si2
```

```
si1=approx(exp(ln_si1),5);
si2=approx(exp(ln_si2),5);
q=[q1 q2];
ln_si=[ln_si1 ln_si2];
si=[si1 si2];
a_=[a1 a2];
b_=[b1 b2];
q_=[q1 q2];

disp('bar cm^6 mol^-2',a,'a = ')
disp('cm^3 mol6-1',b,'b = ')
disp(q,'q = ')

Ans=[Tc',Tr',Pc',bi',ai',a_',b_',q_',ln_si',si'];
disp(Ans,'        Tc        Tr        Pc        bi
            ai        a         b         q         ln
   si        si')

//End
```

Scilab code Exa 14.5 Derive the equations from LLE data

```
clear all;
clc;

//To find Approx Value
function[A]=approx(V,n)
  A=round(V*10^n)/10^n;//V-Value  n-To what place
  funcprot(0)
endfunction

//Example 14.5
//Caption : Program to derive the equations from LLE
```

```
     data
12
13 function[Root]=RF(A,B,K)
14
15      //By Regula Falsi Method
16      i=1;
17     while(i==1)
18      a=A;
19      F_a=log((1-a)/a)+(2*a*K)-K;
20      b=B;
21      F_b=log((1-b)/b)+(2*b*K)-K;
22      x1=((a*F_b)-(b*F_a))/(F_b-F_a);
23      F_x1=log((1-x1)/x1)+(2*x1*K)-K;
24
25      if((F_a*F_x1)<0) then
26        flag=1;
27        A=a;
28        B=x1;
29      else((F_x1*F_b)<0)
30        flag=2;
31        A=x1;
32        B=b;
33      end
34      x1_a=approx(x1,4);
35      b_a=approx(b,4);
36      a_a=approx(a,4);
37
38      if(x1_a==b_a)
39        Root=approx(x1,4);
40        i=0;
41        break;
42      elseif(x1_a==a_a)
43        Root=approx(x1,4);
44        i=0;
45        break;
46      end
47    end
48 endfunction
```

```
49
50 //Example 14.5
51
52 //G_E/RT = Ax1x2                                          (A)
53
54 //ln V1=A*x2^2=A(1-x1)^2
55 //ln V2=A*x1^2
56
57 //A[(1-x1_a)^2 - (1-x1_b)^2] = ln(x1_b/x1_a)              (B)
58
59 //A[(x1_a)^2-(x1_b)^2] = ln((1-x1_b)/(1-x1_a))           (C)
60
61 //x1_b=1-x1_a                                            (D)
62
63 //A(1 - 2*x1) = ln((1-x1)/x1)                            (E)
64
65 //A=a/T + b - clnT                                       (F)
66
67 //H_E=R(a + cT)x1x2                                      (G)
68
69 //Cp_E=(dH_E/dT) = Rcx1x2                                (H)
70
71
72 T=linspace(250.1,450,20);
73 A=(-975.*(T.^-1)) + 22.4 - (3.*log(T));
74
75 subplot(3,2,1)
76 plot(T,A)
77 x=[250.001 450];
78 y=[2 2];
79 plot(x,y,'b--')
80
81 x=[272.9 272.9];
82 y=[1.9001 2.01];
83 plot(x,y,'r--')
84
85 x=[391.2 391.2];
86 y=[1.9001 2.01];
```

```
87  plot(x,y,'g--')
88
89  legend('A vs T(K)','T=272.9,A=2','T=391.2,A=2')
90
91  xtitle('(a)A vs T','T(K)','A')
92
93  subplot(3,2,2)
94
95  clear all;
96
97  T=linspace(272.9,391.2,100);
98  K=approx((-975*(T.^-1))+22.4+(-3.*log(T)),4);
99
100 root=zeros(100,1);
101 for(z=1:13)
102   root(z)=RF(0.4,0.49,K(z));
103 end
104
105 for(z=14:80)
106   root(z)=RF(0.01,0.49,K(z));
107 end
108
109 for(z=81:100)
110   root(z)=RF(0.4,0.49,K(z));
111 end
112 x1=root';
113 plot2d(x1,T,rect=[0,250,1,450])
114
115 x=[0 0.55];
116 y=[272.9 272.9];
117 plot2d(x,y,style=3)
118
119 x=[0 0.55];
120 y=[391.2 391.2];
121 plot2d(x,y,style=4)
122
123 legend('T vs x1','T=272.9K(LCST)','T=391.2K(UCST)')
124 xtitle('(b)T vs x1','x1','T(K)')
```

```
125
126 root=1-x1;
127 x1=root;
128 plot2d(x1,T,rect=[0,250,1,450])
129
130 x=[0.5 0.51];
131 y=[272.9 272.9];
132 plot2d(x,y)
133
134 x=[0.5 0.51];
135 y=[391.2 391.2];
136 plot2d(x,y)
137
138 clear all;
139 //xset('window',1)
140
141
142 T=linspace(250.1,450,20);
143 A=(-540.*(T.^-1)) + 21.1 - (3.*log(T));
144
145 subplot(3,2,3)
146 plot(T,A)
147 x=[250.001 450];
148 y=[2 2];
149 plot(x,y,'b--')
150
151 x=[346 346];
152 y=[1.51 2.2];
153 plot(x,y,'r--')
154
155 legend('A vs T(K)','A=2','T=346K(UCST)')
156 xtitle('(a)A vs T','T(K)','A')
157 subplot(3,2,4)
158 clear all;
159 T=linspace(250,346,100);
160 K=approx((-540*(T.^-1))+21.1+(-3.*log(T)),4);
161
162 root=zeros(100,1);
```

```
for(z=1:100)
  root(z)=RF(0.1,0.49,K(z));
end
x1=root';
plot2d(x1,T,rect=[0,250,1,450])
x=[0 0.55];
y=[346 346];
plot2d(x,y,style=3)
legend('T vs x1','T=346K(UCST)')
xtitle('(b)T vs x1','x1','T(K)')
root=1-x1;
x1=root;

plot2d(x1,T,rect=[0,250,1,450])
x=[0.49 0.51];
y=[345.4 345.4];
plot2d(x,y)

clear all;
//xset('window',2)


T=linspace(250.1,450,20);
A=(-1500.*(T.^-1)) + 23.9 - (3.*log(T));

subplot(3,2,5)
plot(T,A)
x=[250.001 450];
y=[2 2];
plot(x,y,'b--')

x=[339.7 339.7];
y=[1.35 2.2];
plot(x,y,'r--')

legend('A vs T(K)','A=2','T=339.7K(LCST)')
xtitle('(a)A vs T','T(K)','A')
subplot(3,2,6)
```

```
clear all;
T=linspace(339.7,450,100);
K=approx((-1500*(T.^-1))+23.9+(-3.*log(T)),4);

root=zeros(100,1);
for(z=1:100)
  root(z)=RF(0.1,0.49,K(z));
end
x1=root';
plot2d(x1,T,rect=[0,300,1,480])
x=[0 0.55];
y=[339.7 339.7];
plot2d(x,y,style=3)
legend('T vs x1','T=339.7K(LCST)')
xtitle('(b)T vs x1','x1','T(K)')
root=1-x1;
x1=root;

plot2d(x1,T,rect=[0,300,1,480])
x=[0.49 0.51];
y=[339.7 339.7];
plot2d(x,y)

//End
```

Scilab code Exa 14.8 Determine the Phase equlibrium data for the System

```
clear all;
clc;

```

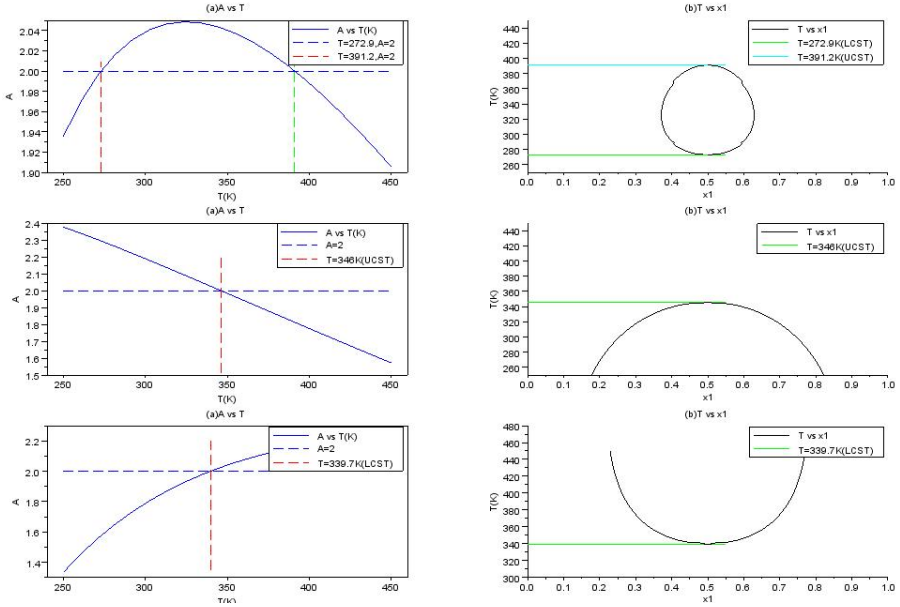

Figure 14.1: Derive the equations from LLE data

```
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Exampl_14.8
11 //Solution : Program to Determine the Phase
       equlibrium data for the System
12
13 A12=4.62424;
14 A21=3.35629;
15 alpha12=3.78608;
16 alpha21=1.81775;
17 B11=-996;
18 B22=-1245;
19 B12=-567;
20
21 P1_sat=103.264;//[kPa]
```

```
22 P2_sat=5.633;//[kPa]
23
24 T=308.15;//[K]
25 R=8314;
26
27 //G_E/RT = T1 = A21*x1 + A12*x2 - Q
28
29 //V1=exp[(x2^2)*[A12 + (2*(A21-A12)*x1) - Q - (x1*(
     dQ/dx1))]]
30
31 //V2=exp[(x1^2)*[A21 + (2*(A12-A21)*x2) - Q + (x2*(
     dQ/dx1))]]
32
33 //Q=(alpha12*x1*alpha21*x2)/((alpha12*x1) + (alpha21
     *x2))
34
35 //dQ/dx1=dQ_x1=(alpha12*alpha21*(alpha21*x2^2 -
     alpha12*x1^2))/((alpha12*x1 + alpha21*x2)^2)
36
37 //P=(x1*V1*P1_sat)/si1 + (x2*V2*P2_sat)/si2
38
39 //d12=2B12-B11-B22
40
41 //si1=exp[((B11*(P-P1_sat)) + (P*y2^2*d12)))/RT]
42
43 //si2=exp[((B22*(P-P2_sat)) + (P*y1^2*d12)))/RT]
44
45 //y1=(x1*V1*P1_sat)/(si1*P)
46
47 //y2=(x2*V2*P2_sat)/(si2*P)
48
49
50 //BUBL P
51
52 x1=[0.01:0.01:0.99];
53 x2=1-x1;
54
55 for(i=1:99)
```

```
56
57    si1=1;//Assumed
58    si2=1;//Assumed
59
60    dP=100;
61    while(dP>0.0001)
62
63      Q=approx(((alpha12*x1(i)*alpha21*x2(i))/((
           alpha12*x1(i)) + (alpha21*x2(i)))),4);
64      dQ_x1=approx((alpha12*alpha21*((alpha21*((x2(i))
           ^2)) - (alpha12*((x1(i))^2))))/(((alpha12*x1(
           i)) + (alpha21*x2(i)))^2),4);
65      V1=approx(exp((x2(i)^2)*(A12 + (2*(A21-A12)*x1(i
           )) - Q - (x1(i)*dQ_x1))),4);
66      V2=approx(exp((x1(i)^2)*(A21 + (2*(A12-A21)*x2(i
           )) - Q + (x2(i)*dQ_x1))),4);
67
68      Pi=approx((((x1(i)*V1*P1_sat)/si1) + ((x2(i)*V2*
           P2_sat)/si2)),4);
69
70      y1=approx((x1(i)*V1*P1_sat)/(si1*Pi),4);
71      y2=approx((x2(i)*V2*P2_sat)/(si2*Pi),4);
72
73      d12=(2*B12)-B11-B22;
74
75      si1=approx(exp(((B11*(Pi-P1_sat))+(Pi*(y2^2)*d12
           ))/(R*T)),4);
76      si2=approx(exp(((B22*(Pi-P2_sat))+(Pi*(y1^2)*d12
           ))/(R*T)),4);
77
78      Pf=approx(((x1(i)*V1*P1_sat)/si1) + ((x2(i)*V2*
           P2_sat)/si2),4);
79
80      dP=abs(Pf-Pi);
81    end
82    P(i)=Pf;
83    y(i)=y1;
84  end
```

```
85
86  for(i=1:99)
87    if(P(i)>104.61)
88      P(i)=%nan;
89    end
90  end
91  x1(100)=1;
92  y(100)=1;
93  P(100)=P1_sat;
94
95  subplot(1,2,1)
96  P_=[5.633 104.6];
97  x=[0 0.0117];
98  plot2d(x,P_,rect=[0,0,0.02,104.6])
99
100 P_=[104.6 104.6];
101 x=[0 0.02];
102 plot(x,P_,'r')
103
104 P_=[5.633 5.633];
105 y1=[0 0.02];
106 plot(y1,P_,'b')
107
108 P_=[104.6 120];
109 xa=[0.0117 0.0117];
110 plot(xa,P_,'g--')
111
112 legend('P vs x1','P* = 104.6kPa','P vs y1','x1_a
      =0.0117')
113 xtitle('P-x-y','x1,y1','P/kPa')
114
115 P_=[100 120];
116 y1=[0.02 0.02];
117 plot(y1,P_,'w')
118
119 subplot(1,2,2)
120
121 P_=[104.6 104.6];
```

```
122 x=[0.943 0.96];
123 plot(x,P_,'r')
124
125 P_=[104.3 104.6];
126 y1=[0.946 0.946];
127 plot(y1,P_,'b--')
128
129 P_=[104.6 104.8];
130 xb=[0.95 0.95];
131 plot(xb,P_,'g--')
132
133 plot2d(x1,P,rect=[0.943,103,1,105])
134 plot2d(y,P,style=3,rect=[0.943,103,1,105])
135
136 legend('P*=104.6kPa','y1*=0.946','x1_b=0.95','P vs
    x1','P vs y1')
137 xtitle('P-x-y(Ether Rich Region)','x1,y1','P/kPa')
138
139 //End
```

Scilab code Exa 14.9 Prepare a Table of Temperature and Compostion Data

```
1 clear all;
2 clc;
3
4 //To find Approx Value
5 function[A]=approx(V,n)
6   A=round(V*10^n)/10^n;//V-Value  n-To what place
7   funcprot(0)
8 endfunction
```

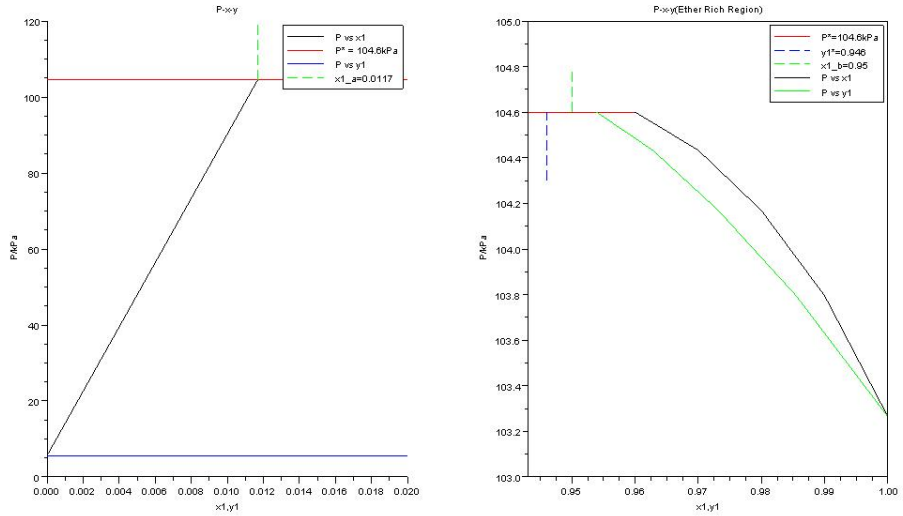

Figure 14.2: Determine the Phase equlibrium data for the System

```
9
10 //Example 14.9
11 //Caption : Program to Prepare a Table of
      Temperature/Compostion Data
12 subplot(1,2,1)
13 P=101.33;//[kPa]
14 T=[333.15 343.15 348.15 353.15 353.25 363.15
      373.15];
15 P1_sat=[52.22 73.47 86.40 101.05 101.33 136.14
      180.04];
16 P2_sat=[19.92 31.16 38.55 47.36 47.56 70.11 101.33];
17 P_=P1_sat+P2_sat;
18 plot(P_,T,'b');
19
20 Ans=[T',P1_sat',P2_sat',P_'];
21 disp(Ans,'     T        P1_sat     P2_sat    P1_sat+
      P2_sat')
22 //Hence P lies between T=333.15 and 343.15
23 P=[101.33 101.33];
```

```
Tp=[330.15 343.15];
xgrid();
plot(P,Tp,'r--')

//Thus by interpolation
P=[50.14 104.63];
T_=[342.15 342.15];

plot(P,T_,'g--');

legend('P* vs T','P=101.33kPa','T=342.15K')
xtitle('T vs P1_sat + P2_sat','P1_sat + P2_sat(kPa)'
   ,'T(K)')

T_=342.15;//[K]

subplot(1,2,2)

plot(P1_sat,T,'b')

//Hence P1_sat @ T=T_=342.15 by interpolation
P=[41 70];
Tp=[342.15 342.15];
xgrid();
plot(P,Tp,'r--')

//Thus by interpolation  P1_sat=71.3kPa
P=[71.3 71.3];
T_=[330.15 342.15];

plot(P,T_,'g--');


legend('P1_sat vs T','T=342.15K','P=71.3kPa')
xtitle('T vs P1_sat','P1_sat(kPa)','T(K)')

P_sat=71.3;
T(1)=342.15;
```

```
61 P=101.33;//[kPa]
62 P1_sat(1)=P_sat;
63 P2_sat(1)=P-P_sat;
64
65 y_I=approx(P1_sat/P,3);
66 y_II=approx(1-(P2_sat/P),3);
67 for(i=1:7)
68   if(y_I(i)>1)
69     y_I(i)=%nan;
70   elseif(y_II(i)>1)
71     y_II(i)=%nan;
72   end
73 end
74
75 Ans=[T',y_I',y_II'];
76 disp(Ans,'       T        y1_I        y1_II')
77
78 //End
```

Scilab code Exa 14.10 Illustrate the Concepts of Pure Gas Adsorption

```
1 clear all;
2 clc;
3
4 //Example 14.10
5 //Caption : Program to illustrate the Concepts of
     Pue Gas Adsorption
6
7 subplot(2,1,1)
8 m=4.7087;
9 b=2.1941;
```

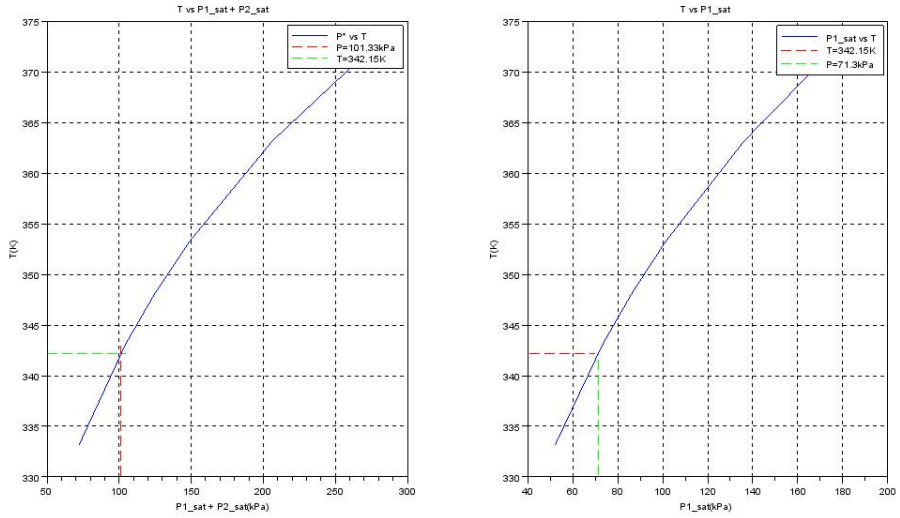

Figure 14.3: Prepare a Table of Temperature and Compostion Data

```
t=0.3984;
P=linspace(0,40,10);
N=(m.*P)./((b+(P.^t)).^(1/t));
plot(P,N)
m=0.6206;
b=1.5454;
t=1;
n=(m.*P)./((b+(P.^t)).^(1/t));
plot(P,n,'b--')
legend('Toth Equation','Langmuir Equation')
xtitle('Adsorption Isotherm(n vs P)','P(kPa)','n(mol
    /kg)')

subplot(2,1,2)
C0=0.4016;
C1=-0.6471;
C2=0.4567;
C3=-0.12;
n=linspace(0,1.6,20);
```

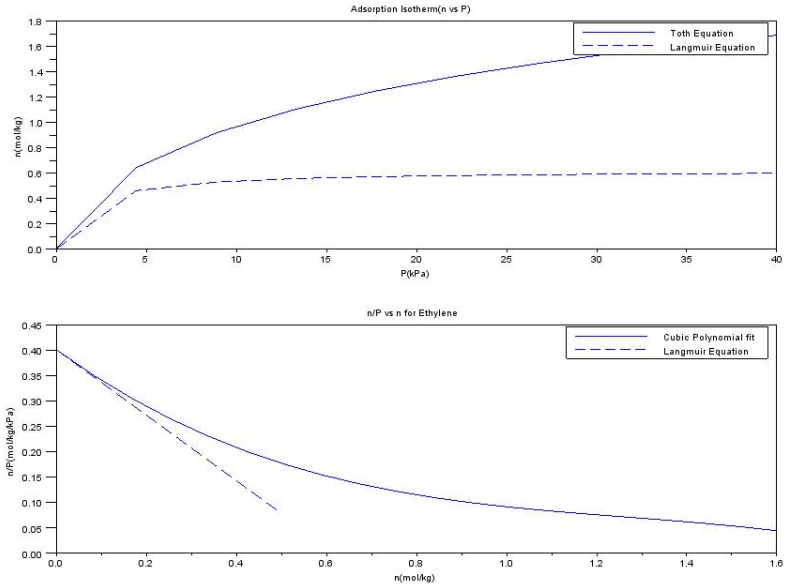

Figure 14.4: Illustrate the Concepts of Pure Gas Adsorption

```
28 K=C0+(C1*n)+(C2*(n^2))+(C3*(n^3));
29 plot(n,K)
30 n=linspace(0,0.5,20);
31 K=C0+(C1*n);
32 plot(n,K,'b--')
33 legend('Cubic Polynomial fit','Langmuir Equation')
34 xtitle('n/P vs n for Ethylene','n(mol/kg)','n/P(mol/
      kg/kPa)')
35
36 //End
```

Chapter 15

Thermodynamic Analysis Of Process

Scilab code Exa 15.1 Do a Thermodynamic Analysis of Steam Power Plant

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 function[Q]=MCPS(T0,T,A,B,C,D)
11   t=T/T0;
12   Q=(A)+(((B*T0)+(((C*T0*T0)+(D/(t*t*T0*T0)))*(t+1)
       /2))*((t-1)/log(t)))
13   funcprot(0);
14 endfunction
15
16 function[Q]=MCPH(T0,T,A,B,C,D)
```

```
17    t=T/T0;
18    Q=(A+((B/2)*T0*(t+1))+((C/3)*T0*T0*((t^2)+t+1))+(D
          /(t*T0*T0)))
19    funcprot(0);
20  endfunction
21
22  //Example 15.1
23  //Caption : Program to do a Thermodynamic Analysis
        of Steam Power Plant
24
25  State=['Supercooled Liquid','Superheated Vapor','Wet
        Vapor,x=0.9378','Saturated Liqiud'];
26  T=[318.98 773.15 318.98 318.98];
27  P=[8600 8600 10 10];
28  H=[203.4 3391.6 2436 191.8];
29  S=[0.6580 6.6858 7.6846 0.6493];
30  T0=298.15;
31  T1=460;//[K]
32  R=8.314;
33  T_sigma=T0;
34  //CH4 + 2O2 --> CO2 + 2H2O
35  dH_CO2=-393509;
36  dH_H2O=-241818;
37  dH_CH4=-74520;
38
39  dG_CO2=-394359;
40  dG_H2O=-228572;
41  dG_CH4=-50460;
42
43  dH_298=dH_CO2+(2*dH_H2O)-dH_CH4
44  dG_298=dG_CO2+(2*dG_H2O)-dG_CH4
45
46  dS_298=approx((dH_298-dG_298)/T0,3);
47
48  //Moles Entering
49  ni_O2=2*1.25;
50  ni_N2=approx(ni_O2*(79/21),3);
51  ni=ni_O2+ni_N2;
```

```
52
53 //Moles After Combustion
54 n_CO2=1;
55 n_H2O=2;
56 n_O2=0.5;
57 n_N2=ni_N2;
58 n=n_CO2+n_H2O+n_O2+n_N2;
59 m=[n_CO2 n_H2O n_N2 n_O2];
60
61 y_CO2=approx(n_CO2/n,4);
62 y_H2O=approx(n_H2O/n,4);
63 y_O2=approx(n_O2/n,4);
64 y_N2=approx(n_N2/n,4);
65
66 y=[y_CO2 y_H2O y_O2 y_N2];
67 yT=sum(y);
68
69 //Step(a)
70 dH_a=0
71 dS_a=approx(ni*R*((0.21*log(0.21))+(0.79*log(0.79)))
      ,3)//[J/K]
72
73 //Step(b)
74 dH_b=dH_298
75 dS_b=dS_298//[J/K]
76
77 //Step(c)
78 dH_c=0
79 dS_c=approx(-n*R*sum(y.*log(y)),3)//[J/K]
80
81 //Step(d)
82 //For CO2
83 CpH_CO2=approx(R*MCPH(T0,T1,5.457,1.045*(10^-3)
      ,0,-1.157*(10^5)),3);
84 //For H2O
85 CpH_H2O=approx(R*MCPH(T0,T1,3.470,1.450*(10^-3)
      ,0,0.121*(10^5)),3);
86 //For O2
```

```
87 CpH_O2=approx(R*MCPH(T0,T1,3.639,0.506*(10^-3)
      ,0,-0.227*(10^5)),3);
88 //For N2
89 CpH_N2=approx(R*MCPH(T0,T1,3.280,0.593*(10^-3)
      ,0,0.040*(10^5)),3);
90
91 //For CO2
92 CpS_CO2=approx(R*MCPS(T0,T1,5.457,1.045*(10^-3)
      ,0,-1.157*(10^5)),3);
93 //For H2O
94 CpS_H2O=approx(R*MCPS(T0,T1,3.470,1.450*(10^-3)
      ,0,0.121*(10^5)),3);
95 //For O2
96 CpS_O2=approx(R*MCPS(T0,T1,3.639,0.506*(10^-3)
      ,0,-0.227*(10^5)),3);
97 //For N2
98 CpS_N2=approx(R*MCPS(T0,T1,3.280,0.593*(10^-3)
      ,0,0.040*(10^5)),3);
99
100 CpH=[CpH_CO2 CpH_H2O CpH_N2 CpH_O2];
101 CpS=[CpS_CO2 CpS_H2O CpS_N2 CpS_O2];
102
103 Comp=['CO2' 'H2O' 'N2' 'O2'];
104
105 Ans=[CpH',CpS'];
106 disp(Ans,'    CpH       CpS',Comp')
107
108 CpHt=approx(sum(m.*CpH),3)//[J/K]
109 CpSt=approx(sum(m.*CpS),3)//[J/K]
110
111 dH_d=approx(CpHt*(T1-T0),0)//[J]
112 dS_d=approx((CpSt*log(T1/T0)),3)//[J/K]
113
114 dH=dH_a+dH_b+dH_c+dH_d//[J]
115 dS=dS_a+dS_b+dS_c+dS_d//[J/K]
116
117 rm=84.75;//[kg/s]
118
```

```
119 rn_CH4=approx((rm*(H(1)-H(2))*1000)/dH,2)//[mol/s]
120
121 rW_ideal=approx(rn_CH4*((dH/1000)-(T0*dS/1000))
      /1000,2)*1000//[KW]
122
123 //(a)  Furnace/Boiler
124 rS_a=approx((rn_CH4*dS/1000)+(rm*(S(2)-S(1))),2)//[
      kJ/s/K]
125
126 rW_a=approx(T_sigma*rS_a/1000,2)*1000//[kW]
127
128 //(b) Turbine
129 rS_b=approx(rm*(S(3)-S(2)),2)//[kW/K]
130
131 rW_b=approx(T_sigma*rS_b/1000,2)*1000//[kW]
132
133 //(c) Condenser
134 Q_c=H(4)-H(3);//[kJ/kg]
135 rQ_c=approx(rm*Q_c/1000,1)*1000//[kJ/s]
136 rS_c=approx((rm*(S(4)-S(3)))-(rQ_c/T_sigma),2)//[kW/
      K]
137 rW_c=approx(T_sigma*rS_c/1000,2)*1000//[kW]
138
139 //(d) Pump
140 rS_d=approx(rm*(S(1)-S(4)),2)//[kW/K]
141 rW_d=approx(T_sigma*rS_d/1000,2)*1000//[kW]
142
143 rS=[rS_a rS_b rS_c rS_d];
144 pS=approx(rS/sum(rS)*100,1);
145 T=[sum(rS) sum(pS)];
146 Process=['Furnace/boiler' 'Turbine' 'Condenser' '
      Pump'];
147 Ans=[rS',pS'];
148 disp(Ans,'   S(kW/K)    %',Process')
149 disp(T)
150 rW_ideal=80000;
151 rW=[rW_ideal rW_a rW_b rW_c rW_d]/1000;
152 pW=approx(rW/sum(rW)*100,1);
```

```
153 T=[sum(rW) sum(pW)];
154 Process=['Ideal' 'Furnace/boiler' 'Turbine' '
      Condenser' 'Pump'];
155 Ans=[rW',pW'];
156 disp(Ans,' W(kW/K)*10^-3    %',Process')
157 disp(T)
158
159 eta=pW(1);
160
161 disp('%',eta,'Efficiency of the power plant is')
162
163 //End
```

Scilab code Exa 15.2 Prepare a Thermodynamic Analysis of Linde System

```
1  clear all;
2  clc;
3
4  //To find Approx Value
5  function[A]=approx(V,n)
6    A=round(V*10^n)/10^n;//V-Value  n-To what place
7    funcprot(0)
8  endfunction
9
10 //Example 15.2
11 //Caption : Program to Prepare a Thermodynamic
      Analysis of Linde System
12
13
14 State=['Superheated Vapor','Superheated Vapor','
      Superheated Vapor','Saturated Liqiud','Saturated
      Vapor','Superheated Vapor'];
```

```
T=[300 300 207.1 111.5 111.5 295];
P=[1 60 60 1 1 1];
H=[1199.8 1140 772 285.4 796.9 1188.9];
S=[11.629 9.359 7.798 4.962 9.523 11.589];
Given=[T',P',H',S'];
disp(Given,'  T/K    P/kPa    H/kJ/Kg    S/kJ/kg/K'
   ,State')
T_sigma=300;//[K]
rQ=5;//[KJ]
rW=1000;//[KJ/s]
z=approx((H(6)-H(2)-rQ)/(H(6)-H(4)),4);

//rW_ideal = (dH*rm) - (T_sigma(dS*rm))
rW_ideal=approx(((z*H(4))+((1-z)*H(6))-H(1))-((
   T_sigma)*((z*S(4))+((1-z)*S(6))-S(1))),1);

//(a) Compression/cooling
rQ_a=(H(2)-H(1))-rW;//[kJ]
rS_a=approx((S(2)-S(1))-(rQ_a/T_sigma),4);//[kJ/Kg/K
   ]

rW_a=T_sigma*rS_a;//[KJ/Kg]

//(b) Exchanger
rS_b=approx(((S(6)-S(5))*(1-z))+(S(3)-S(2))-(rQ/
   T_sigma),4);//[kJ/Kg/K]
rW_b=T_sigma*rS_b;//[KJ/Kg]

//(c) Throttle
rS_c=approx(((S(4)*z)+(S(5)*(1-z))-S(3)),4);//[KJ/Kg
   /K]
rW_c=T_sigma*rS_c;//[KJ/kg]

S=[rS_a rS_b rS_c];
pS=approx((S/sum(S))*100,1);
ES=[sum(S) sum(pS)];

W=approx([rW_ideal rW_a rW_b rW_c],1);
```

```
48 pW=approx((W/sum(W))*100,1);
49 EW=[sum(W) sum(pW)];
50 Ans=[S',pS'];
51 Process=['Compression/Cooling';'Exchanger';'Throttle
       '];
52
53 disp(Ans,'       Si      %',Process)
54 disp(ES,'Sum')
55 Ans=[W',pW'];
56
57 Process=['Ideal';'Compression/Cooling';'Exchanger';'
       Throttle'];
58 disp(Ans,'       Wi      %',Process)
59 disp(EW,'Sum')
```