EBOOK

Michael Andre Franiatte

Programming Equation of State Resolution for the Study of Fluids

EoSResol.exe

EBOOK

Programming Equation of State Resolution for the Study of Fluids

EoSResol.exe

Michael Franiatte

07/06/2017

Before the past nobody had created a tool to calculate the behavior of fluids without the need of using experimental measures. The C# solution presented here can correlate all data on fluids with a simple program. Information about license, EULA and contract for using these following works can be found at https://michaelfraniatte.wordpress.com.

Programming Equation of State Resolution for the Study of Fluids

Michael Franiatte*

Abstract

It's possible with this book and the book "An Equation of State Resolution for the Study of Fluids" by the same author to understand the behaviour of fluids at high temperature and high pressure in geological and industrial processes. The properties *PVTX* and of reactions for fluids can be correlated with numerous measurement and fitting studies and the program presented here which use simple parameters in an only equation of state (EOS). The book with the C# codes put in a solution form with Visual Studio 2010 Express Edition or higher, allows resolving *PVTX* data of fluids combined with only four parameters describing a gas (molar mass and the three critical parameters). The readers whom apply the instructions can find and verify the good agreement with the volumes observed by the studies made by the Scientifics in the past. The properties of fluids as well as the properties of the reactions occurring in the fluids are deduced from the program made with it. The studies of Scientifics whom were working on EOS can be compared with the solution named EoSResol by everyone. The equation of state resolution with this easy access by everyone is important to understand and correlate all the data on the fluids and the reactions acquired until this day, involving a lot of new works in chemical and petroleum industries.

Keywords: *fluids, PVTX properties, reactions, codes, program, equation of state*

* Author correspondence: michael.franiatte@gmail.com

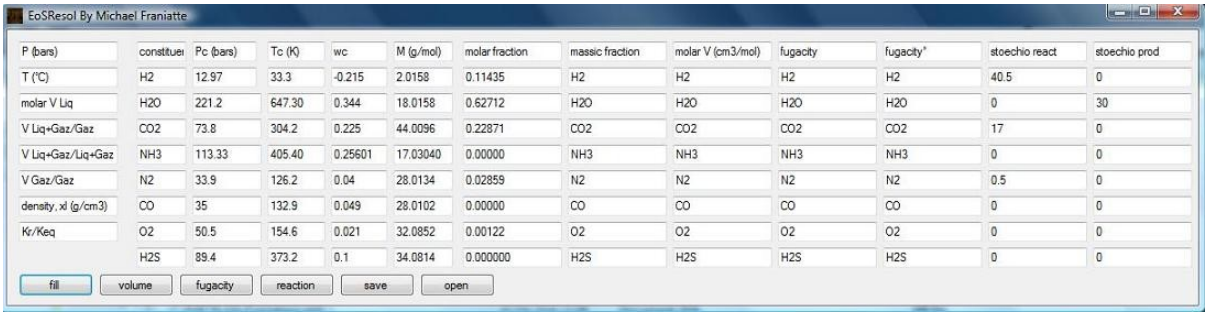
1. Introduction

For more informations on *PVTX* data and reactions, see "An Equation of State Resolution for the Study of Fluids" by the same author.

First, you must download and install C# Visual Studio 2010 Express Edition or Desktop Edition for the higher version.

Create a new solution with a form application, both named EoSResol. Then copy the codes in the chapter 2, in the sheet of the form which is open by clicking on "view code".

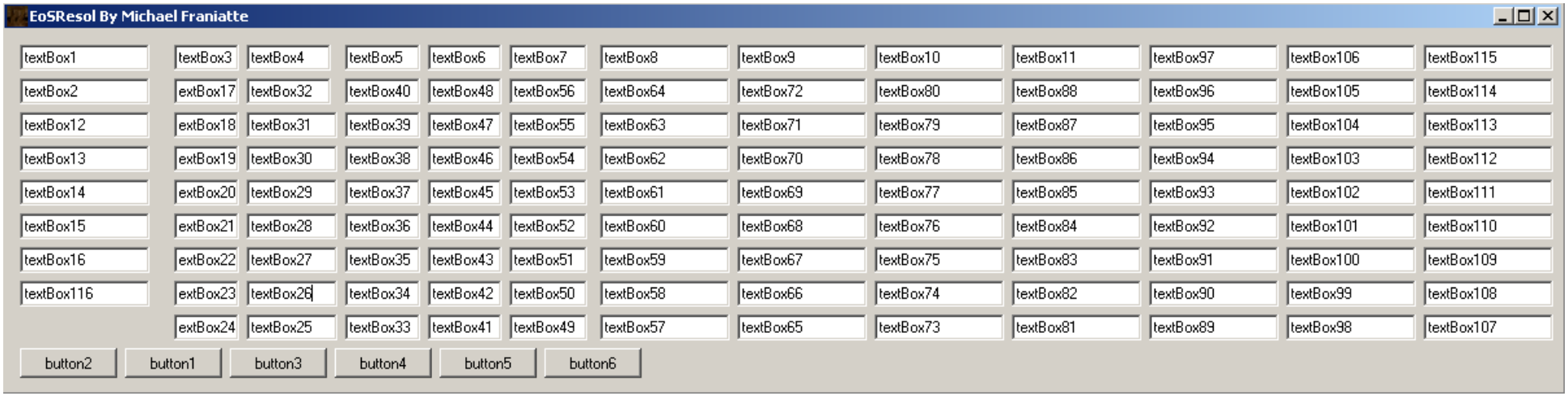
Add textboxes and command buttons as following:



Let the textboxes with blank. The button 2 is to fill the textboxes. After put a pressure and a temperature, click the button 1 for calculation.

To dispose textboxes follow this instruction:

Add the first textbox for pressure, the second for temperature, then the nine textboxes on the first line (after pressure), then the five textboxes under temperature (first row), then add 8 textboxes, the ones with the names of the 8 constituents, then copy these 8 textboxes, 8 times (under critical pressure, critical temperature, acentric factor, molar mass…).


EoSResol By Michael Franiatte
textBox1
textBox3
textBox4
textBox5
textBox6
textBox7
textBox8
textBox9
textBox10
textBox11
textBox97
textBox106
textBox115
textBox2
extBox17
textBox32
textBox40
textBox48
textBox56
textBox64
textBox72
textBox80
textBox88
textBox96
textBox105
textBox114
textBox12
extBox18
textBox31
textBox39
textBox47
textBox55
textBox63
textBox71
textBox79
textBox87
textBox95
textBox104
textBox113
textBox13
extBox19
textBox30
textBox38
textBox46
textBox54
textBox62
textBox70
textBox78
textBox86
textBox94
textBox103
textBox112
textBox14
extBox20
textBox29
textBox37
textBox45
textBox53
textBox61
textBox69
textBox77
textBox85
textBox93
textBox102
textBox111
textBox15
extBox21
textBox28
textBox36
textBox44
textBox52
textBox60
textBox68
textBox76
textBox84
textBox92
textBox101
textBox110
textBox16
extBox22
textBox27
textBox35
textBox43
textBox51
textBox59
textBox67
textBox75
textBox83
textBox91
textBox100
textBox109
textBox116
extBox23
textBox26
textBox34
textBox42
textBox50
textBox58
textBox66
textBox74
textBox82
textBox90
textBox99
textBox108
extBox24
textBox25
textBox33
textBox41
textBox49
textBox57
textBox65
textBox73
textBox81
textBox89
textBox98
textBox107
button2
button1
button3
button4
button5
button6

2. C# codes

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace EoSResol
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        //int text;
        double xH2S;
        double xO2;
        double xCO;
        double xN2;
        double xNH3;
        double xCO2;
        double xH2O;
        double xH2;
        double yH2S;
        double yO2;
        double yCO;
        double yN2;
        double yNH3;
        double yCO2;
        double yH2O;
        double yH2;
        double MH2S;
        double MO2;
        double MCO;
        double MN2;
        double MNH3;
        double MCO2;
        double MH2O;
        double MH2;
        double sumy;
        double XH2Obis;
        double XCO2bis;
        double XCObis;
        double XH2bis;
        double XN2bis;
        double XCH4bis;
        double XNH3bis;
        double XMGbis;
        double Pb;
        double P;
        double T;
        double TcH2O;//température critique de H2O dans la cellule J8
        double PcH2O;//pression critique de H2O
        double TcCO2;
        double PcCO2;
        double TcCO;
        double PcCO;
```

```
double TcH2;
double PcH2;
double TcN2;
double PcN2;
double TcCH4;
double PcCH4;
double TcNH3;
double PcNH3;
double TcMG;
double PcMG;
double R;
double wH2O;
double nH2O;
double alphaH2O;
double wCO2;
double nCO2;
double alphaCO2;
double wCO;
double nCO;
double alphaCO;
double wH2;
double nH2;
double alphaH2;
double wN2;
double nN2;
double alphaN2;
double wCH4;
double nCH4;
double alphaCH4;
double wNH3;
double nNH3;
double alphaNH3;
double wMG;
double nMG;
double alphaMG;
double AH2;
double BH2;
double ACO2;
double BCO2;
double AN2;
double BN2;
double AH2O;
double BH2O;
double ACO;
double BCO;
double ACH4;
double BCH4;
double ANH3;
double BNH3;
double AMG;
double BMG;
double grAbis;
double grAsuite;
double GRA;
double GRB;
double test;
double ZN;
double FZ;
double FpZ;
double ZN1;
double VN;
double V;
```

```
double BiH2;
double BiCO2;
double BiN2;
double BiH2O;
double BiCO;
double BiCH4;
double BiNH3;
double BiMG;
double A;
double B;
double ArH2;
double ArCO2;
double ArN2;
double ArH2O;
double ArCO;
double ArCH4;
double ArNH3;
double ArMG;
double SB;
double DVDXH2;
double DVDXCO2;
double DVDXN2;
double DVDXH2O;
double DVDXCO;
double DVDXCH4;
double DVDXNH3;
double DVDXMG;
double VCO2M;
double VCOM;
double VH2M;
double VN2M;
double VCH4M;
double VNH3M;
double VH2OM;
double VMGM;
double logFIH2O;
double FIH2Oinc;
double FUH2Oinc;
double FUH2Oi;
double logFIH2;
double FIH2inc;
double FUH2inc;
double FUH2i;
double logFICO;
double FICOinc;
double FUCOinc;
double FUCOi;
double logFICO2;
double FICO2inc;
double FUCO2inc;
double FUCO2i;
double logFIN2;
double FIN2inc;
double FUN2inc;
double FUN2i;
double logFICH4;
double FICH4inc;
double FUCH4inc;
double FUCH4i;
double logFINH3;
double FINH3inc;
double FUNH3inc;
```

```
        double FUNH3i;
        double logFIMG;
        double FIMGinc;
        double FUMGinc;
        double FUMGi;
        double M;
        double Vlgg;
        double Vlglg;
        double Vgg;
        double xl;
        double LogKr;
        double LogKeq;
        double LogKrLogkeq;
        private void button1_Click(object sender, EventArgs e)
        {
            //text = 10;
            //textBox1.Text = text.ToString();
            //text = Int16.Parse(textBox1.Text.ToString()) + 10;
            //textBox1.Text = text.ToString();
            xH2S = Double.Parse(textBox57.Text.ToString());
            xO2 = Double.Parse(textBox58.Text.ToString());
            xCO = Double.Parse(textBox59.Text.ToString());
            xN2 = Double.Parse(textBox60.Text.ToString());
            xNH3 = Double.Parse(textBox61.Text.ToString());
            xCO2 = Double.Parse(textBox62.Text.ToString());
            xH2O = Double.Parse(textBox63.Text.ToString());
            xH2 = Double.Parse(textBox64.Text.ToString());
            xH2S          =          Double.Parse(textBox57.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xO2           =          Double.Parse(textBox58.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xCO           =          Double.Parse(textBox59.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xN2           =          Double.Parse(textBox60.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xNH3          =          Double.Parse(textBox61.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xCO2          =          Double.Parse(textBox62.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            xH2O          =          Double.Parse(textBox63.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
```

```
            xH2          =          Double.Parse(textBox64.Text.ToString())          /
(Double.Parse(textBox57.Text.ToString())  +  Double.Parse(textBox58.Text.ToString())  +
Double.Parse(textBox59.Text.ToString())  +  Double.Parse(textBox60.Text.ToString())  +
Double.Parse(textBox61.Text.ToString())  +  Double.Parse(textBox62.Text.ToString())  +
Double.Parse(textBox63.Text.ToString()) + Double.Parse(textBox64.Text.ToString()));
            textBox57.Text = xH2S.ToString(); //"H2";//xH2S
            textBox58.Text = xO2.ToString(); //"H2O";//xO2
            textBox59.Text = xCO.ToString(); //"CO2";//xCO
            textBox60.Text = xN2.ToString(); //"NH3";//xN2
            textBox61.Text = xNH3.ToString(); //"N2";//xNH3
            textBox62.Text = xCO2.ToString(); //"CO";//xCO2
            textBox63.Text = xH2O.ToString(); //"O2";//xH2O
            textBox64.Text = xH2.ToString(); //"H2S";//xH2
            MH2S = Double.Parse(textBox49.Text.ToString());//MH2S
            MO2 = Double.Parse(textBox50.Text.ToString());//MO2
            MCO = Double.Parse(textBox51.Text.ToString());//MCO
            MN2 = Double.Parse(textBox52.Text.ToString());//MN2
            MNH3 = Double.Parse(textBox53.Text.ToString());//MNH3
            MCO2 = Double.Parse(textBox54.Text.ToString());//MCO2
            MH2O = Double.Parse(textBox55.Text.ToString());//MH2O
            MH2 = Double.Parse(textBox56.Text.ToString());//MH2
            yH2S = xH2S * MH2S / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 +
xN2 * MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yO2 = xO2 * MO2 / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 + xN2
* MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yCO = xCO * MCO / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 + xN2
* MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yN2 = xN2 * MN2 / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 + xN2
* MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yNH3 = xNH3 * MNH3 / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 +
xN2 * MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yCO2 = xCO2 * MCO2 / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 +
xN2 * MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yH2O = xH2O * MH2O / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 +
xN2 * MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            yH2 = xH2 * MH2 / (xH2S * MH2S + xO2 * MO2 + xCO * MCO + xCO2 * MCO2 + xN2
* MN2 + xCO2 * MCO2 + xH2O * MH2O + xH2 * MH2);
            sumy = yH2S + yO2 + yCO + yN2 + yNH3 + yCO2 + yH2O + yH2;
            yH2S = yH2S / sumy;
            yO2 = yO2 / sumy;
            yCO = yCO / sumy;
            yN2 = yN2 / sumy;
            yNH3 = yNH3 / sumy;
            yCO2 = yCO2 / sumy;
            yH2O = yH2O / sumy;
            yH2 = yH2 / sumy;
            textBox65.Text = yH2S.ToString();//yH2S
            textBox66.Text = yO2.ToString();//yO2
            textBox67.Text = yCO.ToString();//yCO
            textBox68.Text = yN2.ToString();//yN2
            textBox69.Text = yNH3.ToString();//yNH3
            textBox70.Text = yCO2.ToString();//yCO2
            textBox71.Text = yH2O.ToString();//yH2O
            textBox72.Text = yH2.ToString();//yH2
            ////////////////////////////////////////
            XH2Obis = Double.Parse(textBox63.Text.ToString());
            XCO2bis = Double.Parse(textBox62.Text.ToString());
            XCObis = Double.Parse(textBox61.Text.ToString());
            XH2bis = Double.Parse(textBox64.Text.ToString());
            XN2bis = Double.Parse(textBox58.Text.ToString());
            XCH4bis = Double.Parse(textBox60.Text.ToString());
            XNH3bis = Double.Parse(textBox57.Text.ToString());
```

```
            XMGbis = Double.Parse(textBox59.Text.ToString());
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
```

```
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
```

```
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
```

```
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
```

```
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            textBox78.Text = VCO2M.ToString();
            textBox77.Text = VCOM.ToString();
            textBox80.Text = VH2M.ToString();
            textBox74.Text = VN2M.ToString();
            textBox76.Text = VCH4M.ToString();
            textBox73.Text = VNH3M.ToString();
            textBox79.Text = VH2OM.ToString();
            textBox75.Text = VMGM.ToString();
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
```

```
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
```

```
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
```

```
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////////
            textBox87.Text = FUH2Oi.ToString();
            textBox86.Text = FUCO2i.ToString();
            textBox85.Text = FUCOi.ToString();
            textBox88.Text = FUH2i.ToString();
            textBox82.Text = FUN2i.ToString();
            textBox84.Text = FUCH4i.ToString();
            textBox81.Text = FUNH3i.ToString();
            textBox83.Text = FUMGi.ToString();
            ////////////////////////
            M = MH2 * xH2 + MH2O * xH2O + MCO2 * xCO2 + MNH3 * xNH3 + MN2 * xN2 + MCO
* xCO + MO2 * xO2 + MH2S * xH2S;
            textBox12.Text = M.ToString();
            Vlgg = VMGM + VH2OM + VNH3M + VCH4M + VN2M + VH2M + VCOM + VCO2M;
            textBox13.Text = Vlgg.ToString();
            Vlglg = Vlgg / 2;
            textBox14.Text = Vlglg.ToString();
            Vgg = Vlgg - M;
            textBox15.Text = Vgg.ToString();
            xl = M / Vlgg;
            textBox16.Text = xl.ToString();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            //textBox1.Text = "Hello";
            textBox1.Text = "P (bars)";
            textBox2.Text = "T (°C)";
            textBox3.Text = "constituent";
            textBox4.Text = "Pc (bars)";
            textBox5.Text = "Tc (K)";
            textBox6.Text = "wc";
            textBox7.Text = "M (g/mol)";
            textBox8.Text = "molar fraction";
            textBox9.Text = "massic fraction";
            textBox10.Text = "molar V (cm3/mol)";
            textBox11.Text = "fugacity";
            textBox12.Text = "molar V Liq";
            textBox13.Text = "V Liq+Gaz/Gaz";
            textBox14.Text = "V Liq+Gaz/Liq+Gaz";
            textBox15.Text = "V Gaz/Gaz";
            textBox16.Text = "density, xl (g/cm3)";
            textBox17.Text = "H2";
            textBox18.Text = "H2O";
            textBox19.Text = "CO2";
            textBox20.Text = "NH3";
            textBox21.Text = "N2";
            textBox22.Text = "CO";
            textBox23.Text = "O2";
            textBox24.Text = "H2S";
            textBox25.Text = "89.4"; //"H2";//PcH2S
            textBox26.Text = "50.5"; //"H2O";//PcO2
            textBox27.Text = "35"; //"CO2";//PcCO
            textBox28.Text = "33.9"; //"NH3";//PcN2
            textBox29.Text = "113.33"; //"N2";//PcNH3
            textBox30.Text = "73.8"; //"CO";//PcCO2
            textBox31.Text = "221.2"; //"O2";//PcH2O
            textBox32.Text = "12.97"; //"H2S";//PcH2
            textBox33.Text = "373.2";//TcH2S
```

```
textBox34.Text = "154.6";//TcO2
textBox35.Text = "132.9";//TcCO
textBox36.Text = "126.2";//TcN2
textBox37.Text = "405.40";//TcNH3
textBox38.Text = "304.2";//TcCO2
textBox39.Text = "647.30";//TcH2O
textBox40.Text = "33.3";//TcH2
textBox41.Text = "0.1";//wcH2S
textBox42.Text = "0.021";//wcO2
textBox43.Text = "0.049";//wcCO
textBox44.Text = "0.04";//wcN2
textBox45.Text = "0.25601";//wcNH3
textBox46.Text = "0.225";//wcCO2
textBox47.Text = "0.344";//wcH2O
textBox48.Text = "-0.215";//wcH2
textBox49.Text = "34.0814";//MH2S
textBox50.Text = "32.0852";//MO2
textBox51.Text = "28.0102";//MCO
textBox52.Text = "28.0134";//MN2
textBox53.Text = "17.03040";//MNH3
textBox54.Text = "44.0096";//MCO2
textBox55.Text = "18.0158";//MH2O
textBox56.Text = "2.0158";//MH2
textBox57.Text = "0.000000"; //"H2";//xH2S
textBox58.Text = "0.00122"; //"H2O";//xO2
textBox59.Text = "0.00000"; //"CO2";//xCO
textBox60.Text = "0.02859"; //"NH3";//xN2
textBox61.Text = "0.00000"; //"N2";//xNH3
textBox62.Text = "0.22871"; //"CO";//xCO2
textBox63.Text = "0.62712"; //"O2";//xH2O
textBox64.Text = "0.11435"; //"H2S";//xH2
textBox65.Text = "H2S"; //"H2";//yH2S
textBox66.Text = "O2"; //"H2O";//yO2
textBox67.Text = "CO"; //"CO2";//yCO
textBox68.Text = "N2"; //"NH3";//yN2
textBox69.Text = "NH3"; //"N2";//yNH3
textBox70.Text = "CO2"; //"CO";//yCO2
textBox71.Text = "H2O"; //"O2";//yH2O
textBox72.Text = "H2"; //"H2S";//yH2
textBox73.Text = "H2S"; //"H2";//vH2S
textBox74.Text = "O2"; //"H2O";//vO2
textBox75.Text = "CO"; //"CO2";//vCO
textBox76.Text = "N2"; //"NH3";//vN2
textBox77.Text = "NH3"; //"N2";//vNH3
textBox78.Text = "CO2"; //"CO";//vCO2
textBox79.Text = "H2O"; //"O2";//vH2O
textBox80.Text = "H2"; //"H2S";//vH2
textBox81.Text = "H2S"; //"H2";//fH2S
textBox82.Text = "O2"; //"H2O";//fO2
textBox83.Text = "CO"; //"CO2";//fCO
textBox84.Text = "N2"; //"NH3";//fN2
textBox85.Text = "NH3"; //"N2";//fNH3
textBox86.Text = "CO2"; //"CO";//fCO2
textBox87.Text = "H2O"; //"O2";//fH2O
textBox88.Text = "H2"; //"H2S";//fH2
textBox89.Text = "H2S"; //"H2";//f°H2S
textBox90.Text = "O2"; //"H2O";//f°O2
textBox91.Text = "CO"; //"CO2";//f°CO
textBox92.Text = "N2"; //"NH3";//f°N2
textBox93.Text = "NH3"; //"N2";//f°NH3
textBox94.Text = "CO2"; //"CO";//f°CO2
textBox95.Text = "H2O"; //"O2";//f°H2O
```

```
            textBox96.Text = "H2"; //"H2S";//f°H2
            textBox97.Text = "fugacity°";
            textBox98.Text = "0"; //"H2";//RH2S
            textBox99.Text = "0"; //"H2O";//RO2
            textBox100.Text = "0"; //"CO2";//RCO
            textBox101.Text = "0.5"; //"NH3";//RN2
            textBox102.Text = "0"; //"N2";//RNH3
            textBox103.Text = "17"; //"CO";//RCO2
            textBox104.Text = "0"; //"O2";//RH2O
            textBox105.Text = "40.5"; //"H2S";//RH2
            textBox106.Text = "stoechio react";
            textBox107.Text = "0"; //"H2";//PH2S
            textBox108.Text = "0"; //"H2O";//PO2
            textBox109.Text = "0"; //"CO2";//PCO
            textBox110.Text = "0"; //"NH3";//PN2
            textBox111.Text = "0"; //"N2";//PNH3
            textBox112.Text = "0"; //"CO";//PCO2
            textBox113.Text = "30"; //"O2";//PH2O
            textBox114.Text = "0"; //"H2S";//PH2
            textBox115.Text = "stoechio prod";
            textBox116.Text = "Kr/Keq";
        }

        private void button3_Click(object sender, EventArgs e)
        {
            MH2S = Double.Parse(textBox49.Text.ToString());//MH2S
            MO2 = Double.Parse(textBox50.Text.ToString());//MO2
            MCO = Double.Parse(textBox51.Text.ToString());//MCO
            MN2 = Double.Parse(textBox52.Text.ToString());//MN2
            MNH3 = Double.Parse(textBox53.Text.ToString());//MNH3
            MCO2 = Double.Parse(textBox54.Text.ToString());//MCO2
            MH2O = Double.Parse(textBox55.Text.ToString());//MH2O
            MH2 = Double.Parse(textBox56.Text.ToString());//MH2
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 1;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
```

```
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
```

```
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
```

```
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis * Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis * XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0) * XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 * (1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis * Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O, 0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) * XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis * Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 - 0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 * (1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2 * (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis * XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 * AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) + Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5) + 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis * XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 * AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
```

```
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
```

```
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
```

```
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
```

```
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////////
            textBox89.Text = FUNH3i.ToString();
            ////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 1;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
```

```
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
```

```
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
```

```
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
```

```
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
```

```
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
```

```
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
```

```
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////
            textBox90.Text = FUN2i.ToString();
            ////////////////////////////////////////////:
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 1;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O  =  Double.Parse(textBox31.Text.ToString());  //pression  critique  de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
```

```
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
```

```
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
```

```
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
```

```
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
```

```
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
```

```
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
```

```
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////
            textBox91.Text = FUMGi.ToString();
            ////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 0;
            XCH4bis = 1;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
```

```
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
```

```
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
```

```
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis * Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis * XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0) * XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 * (1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis * Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O, 0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) * XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis * Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 - 0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 * (1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2 * (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis * XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 * AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) + Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5) + 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis * XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
```

```
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
```

```
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
```

```
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
```

```
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////////
            textBox92.Text = FUCH4i.ToString();
            ////////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 1;
            XH2bis = 0;
```

```
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
```

```
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
```

```
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
```

```
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
```

```
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
```

```
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
```

```
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////
            textBox93.Text = FUCOi.ToString();
```

```
            ////////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 1;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O  =  Double.Parse(textBox31.Text.ToString());  //pression  critique  de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2  =  0.42748  *  alphaH2  *  Math.Pow(TcH2,  2)  /  (PcH2  *  100000)  *  P  /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
```

```
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
```

```
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis * Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis * XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0) * XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 * (1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis * Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O, 0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) * XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis * Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 - 0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 * (1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2 * (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis * XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 * AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) + Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5) + 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis * XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
```

```
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
```

```
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
```

```
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
```

```
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
```

```
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////////
            textBox94.Text = FUCO2i.ToString();
            ////////////////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 1;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 0;
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O = Double.Parse(textBox31.Text.ToString()); //pression critique de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
```

```
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
```

```
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis * Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis * XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0) * XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 * (1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis * Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O, 0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) * XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis * Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 - 0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 * (1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2 * (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis * XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 * AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) + Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
```

```
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
```

```
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
```

```
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
```

```
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) /
ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O
/ GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 /
GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO /
GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2
/ GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 /
GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
```

```
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////////
            textBox95.Text = FUH2Oi.ToString();
            ////////////////////////////////////////
            ////////////////////////////////////////
            XH2Obis = 0;
            XCO2bis = 0;
            XCObis = 0;
            XH2bis = 1;
            XN2bis = 0;
            XCH4bis = 0;
            XNH3bis = 0;
            XMGbis = 0;
            Pb = Double.Parse(textBox1.Text.ToString());
            P = Pb * 100000; //passage de la pression de bar en Pa
            T = Double.Parse(textBox2.Text.ToString()) + 273.15;
            TcH2O = Double.Parse(textBox39.Text.ToString()); //température critique de
H2O dans la cellule J8
            PcH2O  =  Double.Parse(textBox31.Text.ToString());  //pression  critique  de
H2O
            TcCO2 = Double.Parse(textBox38.Text.ToString());
            PcCO2 = Double.Parse(textBox30.Text.ToString());
            TcCO = Double.Parse(textBox37.Text.ToString());
            PcCO = Double.Parse(textBox29.Text.ToString());
            TcH2 = Double.Parse(textBox40.Text.ToString());
            PcH2 = Double.Parse(textBox32.Text.ToString());
            TcN2 = Double.Parse(textBox34.Text.ToString());
            PcN2 = Double.Parse(textBox26.Text.ToString());
            TcCH4 = Double.Parse(textBox36.Text.ToString());
            PcCH4 = Double.Parse(textBox28.Text.ToString());
            TcNH3 = Double.Parse(textBox33.Text.ToString());
            PcNH3 = Double.Parse(textBox25.Text.ToString());
            TcMG = Double.Parse(textBox35.Text.ToString());
            PcMG = Double.Parse(textBox27.Text.ToString());
            R = 8.314472; //constante des gaz parfaits
            ////////////////////////////////////////////
            //calcul des facteurs acentriques
            wH2O = Double.Parse(textBox47.Text.ToString());
            nH2O = 0.48508 + 1.55171 * wH2O - 0.15613 * Math.Pow(wH2O, 2);
            alphaH2O = Math.Pow(1 + nH2O * (1 - Math.Pow(T / TcH2O, 0.5)), 2);
            wCO2 = Double.Parse(textBox46.Text.ToString());
            nCO2 = 0.48508 + 1.55171 * wCO2 - 0.15613 * Math.Pow(wCO2, 2);
            alphaCO2 = Math.Pow(1 + nCO2 * (1 - Math.Pow(T / TcCO2, 0.5)), 2);
            wCO = Double.Parse(textBox45.Text.ToString());
            nCO = 0.48508 + 1.55171 * wCO - 0.15613 * Math.Pow(wCO, 2);
            alphaCO = Math.Pow(1 + nCO * (1 - Math.Pow(T / TcCO, 0.5)), 2);
            wH2 = Double.Parse(textBox48.Text.ToString());
            nH2 = 0.48508 + 1.55171 * wH2 - 0.15613 * Math.Pow(wH2, 2);
            alphaH2 = Math.Pow(1 + nH2 * (1 - Math.Pow(T / TcH2, 0.5)), 2);
            wN2 = Double.Parse(textBox42.Text.ToString());
            nN2 = 0.48508 + 1.55171 * wN2 - 0.15613 * Math.Pow(wN2, 2);
            alphaN2 = Math.Pow(1 + nN2 * (1 - Math.Pow(T / TcN2, 0.5)), 2);
            wCH4 = Double.Parse(textBox44.Text.ToString());
```

```
            nCH4 = 0.48508 + 1.55171 * wCH4 - 0.15613 * Math.Pow(wCH4, 2);
            alphaCH4 = Math.Pow(1 + nCH4 * (1 - Math.Pow(T / TcCH4, 0.5)), 2);
            wNH3 = Double.Parse(textBox41.Text.ToString());
            nNH3 = 0.48508 + 1.55171 * wNH3 - 0.15613 * Math.Pow(wNH3, 2);
            alphaNH3 = Math.Pow(1 + nNH3 * (1 - Math.Pow(T / TcNH3, 0.5)), 2);
            wMG = Double.Parse(textBox43.Text.ToString());
            nMG = 0.48508 + 1.55171 * wMG - 0.15613 * Math.Pow(wMG, 2);
            alphaMG = Math.Pow(1 + nMG * (1 - Math.Pow(T / TcMG, 0.5)), 2);
            ////////////////////////
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 +
XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
```

```
            ////////////////////////////////
            test = 10;
            ZN = 1000.01; //initialisation NR à changer si plantage
            while (test > 0.000000001)
            {
                FZ = Math.Pow(ZN, 3) - Math.Pow(ZN, 2) + (GRA - Math.Pow(GRB, 2) -
GRB) * ZN - GRA * GRB;
                FpZ = 3 * Math.Pow(ZN, 2) - 2 * ZN + (GRA - Math.Pow(GRB, 2) - GRB);
                ZN1 = ZN - FZ / FpZ;
                test = Math.Abs(ZN1 - ZN);
                ZN = ZN1;
            }
            VN = (ZN * R * T / P);
            V = VN * 1000000;
            ////////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, aialphai et bialphai qui interviennent dans le calcul des coefficients de
fugacité
            AH2 = 0.42748 * alphaH2 * (R * Math.Pow(TcH2, 2)) / (PcH2 * 100000);
            BH2 = 0.08664 * R * TcH2 / (PcH2 * 100000);
            BiH2 = BH2; //stockage de bialphai
            ACO2 = 0.42748 * alphaCO2 * (R * Math.Pow(TcCO2, 2)) / (PcCO2 * 100000);
            BCO2 = 0.08664 * R * TcCO2 / (PcCO2 * 100000);
            BiCO2 = BCO2;
            AN2 = 0.42748 * alphaN2 * (R * Math.Pow(TcN2, 2)) / (PcN2 * 100000);
            BN2 = 0.08664 * R * TcN2 / (PcN2 * 100000);
            BiN2 = BN2;
            AH2O = 0.42748 * alphaH2O * (R * Math.Pow(TcH2O, 2)) / (PcH2O * 100000);
            BH2O = 0.08664 * R * TcH2O / (PcH2O * 100000);
            BiH2O = BH2O;
            ACO = 0.42748 * alphaCO * (R * Math.Pow(TcCO, 2)) / (PcCO * 100000);
            BCO = 0.08664 * R * TcCO / (PcCO * 100000);
            BiCO = BCO;
            ACH4 = 0.42748 * alphaCH4 * (R * Math.Pow(TcCH4, 2)) / (PcCH4 * 100000);
            BCH4 = 0.08664 * R * TcCH4 / (PcCH4 * 100000);
            BiCH4 = BCH4;
            ANH3 = 0.42748 * alphaNH3 * (R * Math.Pow(TcNH3, 2)) / (PcNH3 * 100000);
            BNH3 = 0.08664 * R * TcNH3 / (PcNH3 * 100000);
            BiNH3 = BNH3;
            AMG = 0.42748 * alphaMG * (R * Math.Pow(TcMG, 2)) / (PcMG * 100000);
            BMG = 0.08664 * R * TcMG / (PcMG * 100000);
            BiMG = BMG;
            ////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, a et b qui n'interviennent pas dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis *
Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5)
+ 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis *
XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG *
ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0)
* XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 *
(1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis *
XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2
* ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 -
0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis *
Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O,
0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) *
XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis *
Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5)
+ grAbis;
            A = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 -
```

```
0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 *
(1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2
* (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis *
XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 *
AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) +
Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5)
+ 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis *
XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO *
ACO2, 0.5) + grAsuite;
            B = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis
* BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            ////////////////////////////
            //calcul de dérivés de XiXj(1-Kji)racine(aialphai*akalphak)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            ////////////////////////////
```

```
            SB = BH2O + BH2 + BCO2 + BN2 + BCO + BNH3 + BCH4 + BMG;
            DVDXH2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArH2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXH2O = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArH2O) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCO2) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXCO = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArCO)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXN2 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArN2)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));
            DVDXCH4 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArCH4) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXNH3 = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) *
ArNH3) / (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) *
Math.Pow(VN - B, 2));
            DVDXMG = (-(R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * VN *
Math.Pow(VN - B, 2)) * SB + (VN - B) * VN * (Math.Pow(VN, 2) - Math.Pow(B, 2)) * ArMG)
/ (-R * T * Math.Pow(VN, 2) * Math.Pow(VN + B, 2) + A * (2 * VN + B) * Math.Pow(VN -
B, 2));

            VCO2M = (VN * 1 * (XCO2bis) - 1 * (XCO2bis - 0 * XCO2bis) * 1 / 3 / 8 / 2
* DVDXCO2) * 1000000;
            VCOM = (VN * 1 * (XCObis) - 1 * (XCObis - 0 * XCObis) * 1 / 3 / 8 / 2 *
DVDXCO) * 1000000;
            VH2M = (VN * 1 * (XH2bis) - 1 * (XH2bis - 0 * XH2bis) * 1 / 3 / 8 / 2 *
DVDXH2) * 1000000;
            VN2M = (VN * 1 * (XN2bis) - 1 * (XN2bis - 0 * XN2bis) * 1 / 3 / 8 / 2 *
DVDXN2) * 1000000;
            VCH4M = (VN * 1 * (XCH4bis) - 1 * (XCH4bis - 0 * XCH4bis) * 1 / 3 / 8 / 2
* DVDXCH4) * 1000000;
            VNH3M = (VN * 1 * (XNH3bis) - 1 * (XNH3bis - 0 * XNH3bis) * 1 / 3 / 8 / 2
* DVDXNH3) * 1000000;
            VH2OM = (VN * 1 * (XH2Obis) - 1 * (XH2Obis - 0 * XH2Obis) * 1 / 3 / 8 / 2
* DVDXH2O) * 1000000;
            VMGM = (VN * 1 * (XMGbis) - 1 * (XMGbis - 0 * XMGbis) * 1 / 3 / 8 / 2 *
DVDXMG) * 1000000;
            //calcul de somme de Xk(1-Kki)racine(aialphai*akalphak)  (avant le 2 dans
le calcul du coefficient de fugacité de l'espèce k)
            grAbis = (XMGbis) * Math.Pow(AH2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArH2 = ((XH2bis)) * AH2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO2, 0.5) + grAbis;
            ArCO2 = ((XCO2bis)) * ACO2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO2,
```

```
0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO2 * AH2, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AN2 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            ArN2 = ((XN2bis)) * AN2 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AN2, 0.5)
+ (1 - 0) * (XH2bis) * Math.Pow(AN2 * AH2, 0.5) + (1 - 0) * (XCO2bis) * Math.Pow(AN2 *
ACO2, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AN2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(AH2O * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AH2O * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * AH2O, 0.5) + grAbis;
            ArH2O = (XH2Obis) * AH2O + (1 - 0) * (XH2bis) * Math.Pow(AH2O * AH2, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AH2O, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AH2O, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AH2O, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACO * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ACO * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACO, 0.5) + grAbis;
            ArCO = (XCObis) * ACO + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACO, 0.5) +
(1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACO, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2 *
ACO, 0.5) + (1 - 0) * (XH2bis) * Math.Pow(ACO * AH2, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ACH4 * AMG, 0.5);
            grAsuite = (1 - 0) * (XH2bis) * Math.Pow(AH2 * ACH4, 0.5) + (1 - 0) *
(XNH3bis) * Math.Pow(ANH3 * ACH4, 0.5) + grAbis;
            ArCH4 = ((XCH4bis)) * ACH4 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ACH4,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ACH4, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ACH4, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ACH4, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XMGbis) * Math.Pow(ANH3 * AMG, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(ANH3 * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(ANH3 * AH2, 0.5) + grAbis;
            ArNH3 = ((XNH3bis)) * ANH3 + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * ANH3,
0.5) + (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * ANH3, 0.5) + (1 - 0) * (XN2bis) *
Math.Pow(AN2 * ANH3, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * ANH3, 0.5) + grAsuite;
            grAbis = (1 - 0) * (XNH3bis) * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = (1 - 0) * (XCH4bis) * Math.Pow(AMG * ACH4, 0.5) + (1 - 0) *
(XH2bis) * Math.Pow(AMG * AH2, 0.5) + grAbis;
            ArMG = ((XMGbis)) * AMG + (1 - 0) * (XH2Obis) * Math.Pow(AH2O * AMG, 0.5)
+ (1 - 0) * (XCO2bis) * Math.Pow(ACO2 * AMG, 0.5) + (1 - 0) * (XN2bis) * Math.Pow(AN2
* AMG, 0.5) + (1 - 0) * (XCObis) * Math.Pow(ACO * AMG, 0.5) + grAsuite;
            //calculs des paramètres de repulsion et d'attraction de l'equation
d'etat, Ai et Bi qui interviennent dans le calcul du coefficient de fugacité
            AH2 = 0.42748 * alphaH2 * Math.Pow(TcH2, 2) / (PcH2 * 100000) * P /
Math.Pow(T, 2); //avec Tr=T/Tc et Pr=P/Pc
            BH2 = 0.08664 * TcH2 / (PcH2 * 100000) * P / (T);
            ACO2 = 0.42748 * alphaCO2 * Math.Pow(TcCO2, 2) / (PcCO2 * 100000) * P /
Math.Pow(T, 2);
            BCO2 = 0.08664 * TcCO2 / (PcCO2 * 100000) * P / (T);
            AN2 = 0.42748 * alphaN2 * Math.Pow(TcN2, 2) / (PcN2 * 100000) * P /
Math.Pow(T, 2);
            BN2 = 0.08664 * TcN2 / (PcN2 * 100000) * P / (T);
            AH2O = 0.42748 * alphaH2O * Math.Pow(TcH2O, 2) / (PcH2O * 100000) * P /
Math.Pow(T, 2);
            BH2O = 0.08664 * TcH2O / (PcH2O * 100000) * P / (T);
            ACO = 0.42748 * alphaCO * Math.Pow(TcCO, 2) / (PcCO * 100000) * P /
Math.Pow(T, 2);
            BCO = 0.08664 * TcCO / (PcCO * 100000) * P / (T);
            ACH4 = 0.42748 * alphaCH4 * Math.Pow(TcCH4, 2) / (PcCH4 * 100000) * P /
Math.Pow(T, 2);
            BCH4 = 0.08664 * TcCH4 / (PcCH4 * 100000) * P / (T);
            ANH3 = 0.42748 * alphaNH3 * Math.Pow(TcNH3, 2) / (PcNH3 * 100000) * P /
Math.Pow(T, 2);
            BNH3 = 0.08664 * TcNH3 / (PcNH3 * 100000) * P / (T);
            AMG = 0.42748 * alphaMG * Math.Pow(TcMG, 2) / (PcMG * 100000) * P /
```

```
Math.Pow(T, 2);
            BMG = 0.08664 * TcMG / (PcMG * 100000) * P / (T);
            ////////////////////////////////////
            //calculs des paramètres de repulsion et d'attraction de l'equation d'etat, A et B qui interviennent dans le calcul du coefficient de fugacité
            grAbis = Math.Pow(XMGbis, 2) * AMG + 2 * (1 - 0) * XMGbis * XH2bis * Math.Pow(AMG * AH2, 0.5) + 2 * (1 - 0) * XMGbis * XCO2bis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XN2bis * Math.Pow(AMG * AN2, 0.5) + 2 * (1 - 0) * XMGbis * XH2Obis * Math.Pow(AMG * AH2O, 0.5) + 2 * (1 - 0) * XMGbis * XCObis * Math.Pow(AMG * ACO2, 0.5) + 2 * (1 - 0) * XMGbis * XCH4bis * Math.Pow(AMG * ACH4, 0.5) + 2 * (1 - 0) * XMGbis * XNH3bis * Math.Pow(AMG * ANH3, 0.5);
            grAsuite = Math.Pow(XCH4bis, 2) * ACH4 + Math.Pow(XNH3bis, 2) * ANH3 + 2 * (1 - 0) * XCH4bis * XCObis * Math.Pow(ACO * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2Obis * Math.Pow(AH2O * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XCO2bis * Math.Pow(ACO2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XH2bis * Math.Pow(AH2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XN2bis * Math.Pow(AN2 * ACH4, 0.5) + 2 * (1 - 0) * XCH4bis * XNH3bis * Math.Pow(ANH3 * ACH4, 0.5) + 2 * (1 - 0) * XH2Obis * XNH3bis * Math.Pow(ANH3 * AH2O, 0.5) + 2 * (1 - 0) * XCO2bis * XNH3bis * Math.Pow(ANH3 * ACO2, 0.5) + 2 * (1 - 0) * XCObis * XNH3bis * Math.Pow(ANH3 * ACO, 0.5) + 2 * (1 - 0) * XH2bis * XNH3bis * Math.Pow(ANH3 * AH2, 0.5) + 2 * (1 - 0) * XN2bis * XNH3bis * Math.Pow(ANH3 * AN2, 0.5) + grAbis;
            GRA = Math.Pow(XH2Obis, 2) * AH2O + Math.Pow(XCO2bis, 2) * ACO2 + 2 * (1 - 0) * XH2Obis * XCO2bis * Math.Pow(AH2O * ACO2, 0.5) + Math.Pow(XH2bis, 2) * AH2 + 2 * (1 - 0) * XH2Obis * XH2bis * Math.Pow(AH2O * AH2, 0.5) + Math.Pow(XN2bis, 2) * AN2 + 2 * (1 - 0) * XH2Obis * XN2bis * Math.Pow(AH2O * AN2, 0.5) + 2 * (1 - 0) * XCO2bis * XH2bis * Math.Pow(ACO2 * AH2, 0.5) + 2 * (1 - 0) * XCO2bis * XN2bis * Math.Pow(ACO2 * AN2, 0.5) + 2 * (1 - 0) * XN2bis * XH2bis * Math.Pow(AN2 * AH2, 0.5) + Math.Pow(XCObis, 2) * ACO + 2 * (1 - 0) * XH2Obis * XCObis * Math.Pow(AH2O * ACO, 0.5) + 2 * (1 - 0) * XCObis * XH2bis * Math.Pow(ACO * AH2, 0.5) + 2 * (1 - 0) * XCObis * XN2bis * Math.Pow(ACO * AN2, 0.5) + 2 * (1 - 0) * XCObis * XCO2bis * Math.Pow(ACO * ACO2, 0.5) + grAsuite;
            GRB = XH2Obis * BH2O + XH2bis * BH2 + XCO2bis * BCO2 + XN2bis * BN2 + XCObis * BCO + XNH3bis * BNH3 + XCH4bis * BCH4 + XMGbis * BMG;
            //calculs des coefficients de fugacités
            //logFIH2Osoave = ZN - 1 - Log(ZN - GRB) - GRA / GRB * Log((ZN + GRB) / ZN)
            //FIH2Oincsoave = 10 ^ (logFIH2Osoave / 2.303)
            //Worksheets(1).Range("C31").Value = FIH2Oincsoave
            logFIH2O = (BH2O / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2O / GRB - 2 / A * ArH2O) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2Oinc = Math.Pow(10, logFIH2O);
            FUH2Oinc = FIH2Oinc * P * XH2Obis;
            FUH2Oi = FUH2Oinc * 0.00001;
            logFIH2 = (BH2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BH2 / GRB - 2 / A * ArH2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIH2inc = Math.Pow(10, logFIH2);
            FUH2inc = FIH2inc * P * XH2bis;
            FUH2i = FUH2inc * 0.00001;
            logFICO = (BCO / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO / GRB - 2 / A * ArCO) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICOinc = Math.Pow(10, logFICO);
            FUCOinc = FICOinc * P * XCObis;
            FUCOi = FUCOinc * 0.00001;
            logFICO2 = (BCO2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCO2 / GRB - 2 / A * ArCO2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICO2inc = Math.Pow(10, logFICO2);
            FUCO2inc = FICO2inc * P * XCO2bis;
            FUCO2i = FUCO2inc * 0.00001; //la même chose mais en bar
            logFIN2 = (BN2 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BN2 / GRB - 2 / A * ArN2) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIN2inc = Math.Pow(10, logFIN2);
            FUN2inc = FIN2inc * P * XN2bis;
```

```
            FUN2i = FUN2inc * 0.00001;
            logFICH4 = (BCH4 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BCH4
/ GRB - 2 / A * ArCH4) * Math.Log(1 + GRB / ZN)) / 2.303;
            FICH4inc = Math.Pow(10, logFICH4);
            FUCH4inc = FICH4inc * P * XCH4bis;
            FUCH4i = FUCH4inc * 0.00001;
            logFINH3 = (BNH3 / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BNH3
/ GRB - 2 / A * ArNH3) * Math.Log(1 + GRB / ZN)) / 2.303;
            FINH3inc = Math.Pow(10, logFINH3);
            FUNH3inc = FINH3inc * P * XNH3bis;
            FUNH3i = FUNH3inc * 0.00001;
            logFIMG = (BMG / GRB * (ZN - 1) - Math.Log(ZN - GRB) + GRA / GRB * (BMG /
GRB - 2 / A * ArMG) * Math.Log(1 + GRB / ZN)) / 2.303;
            FIMGinc = Math.Pow(10, logFIMG);
            FUMGinc = FIMGinc * P * XMGbis;
            FUMGi = FUMGinc * 0.00001;
            ////////////////////////
            textBox96.Text = FUH2i.ToString();
        }

        private void button4_Click(object sender, EventArgs e)
        {
            LogKr  =  Math.Log(1  /  (Math.Pow(Double.Parse(textBox81.Text.ToString()),
Double.Parse(textBox98.Text.ToString()))                                          *
Math.Pow(Double.Parse(textBox82.Text.ToString()),
Double.Parse(textBox99.Text.ToString()))                                          *
Math.Pow(Double.Parse(textBox83.Text.ToString()),
Double.Parse(textBox100.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox84.Text.ToString()),
Double.Parse(textBox101.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox85.Text.ToString()),
Double.Parse(textBox102.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox86.Text.ToString()),
Double.Parse(textBox103.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox87.Text.ToString()),
Double.Parse(textBox104.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox88.Text.ToString()),
Double.Parse(textBox105.Text.ToString()))))                                       *
(Math.Pow(Double.Parse(textBox81.Text.ToString()),
Double.Parse(textBox107.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox82.Text.ToString()),
Double.Parse(textBox108.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox83.Text.ToString()),
Double.Parse(textBox109.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox84.Text.ToString()),
Double.Parse(textBox110.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox85.Text.ToString()),
Double.Parse(textBox111.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox86.Text.ToString()),
Double.Parse(textBox112.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox87.Text.ToString()),
Double.Parse(textBox113.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox88.Text.ToString()),
Double.Parse(textBox114.Text.ToString())))) / 2.303;
            LogKeq  =  Math.Log(1  /  (Math.Pow(Double.Parse(textBox89.Text.ToString()),
Double.Parse(textBox98.Text.ToString()))                                          *
Math.Pow(Double.Parse(textBox90.Text.ToString()),
Double.Parse(textBox99.Text.ToString()))                                          *
Math.Pow(Double.Parse(textBox91.Text.ToString()),
Double.Parse(textBox100.Text.ToString()))                                         *
Math.Pow(Double.Parse(textBox92.Text.ToString()),
Double.Parse(textBox101.Text.ToString()))                                         *
```

```
Math.Pow(Double.Parse(textBox93.Text.ToString()),
Double.Parse(textBox102.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox94.Text.ToString()),
Double.Parse(textBox103.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox95.Text.ToString()),
Double.Parse(textBox104.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox96.Text.ToString()),
Double.Parse(textBox105.Text.ToString()))))                                  *
(Math.Pow(Double.Parse(textBox89.Text.ToString()),
Double.Parse(textBox107.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox90.Text.ToString()),
Double.Parse(textBox108.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox91.Text.ToString()),
Double.Parse(textBox109.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox92.Text.ToString()),
Double.Parse(textBox110.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox93.Text.ToString()),
Double.Parse(textBox111.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox94.Text.ToString()),
Double.Parse(textBox112.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox95.Text.ToString()),
Double.Parse(textBox113.Text.ToString()))                                    *
Math.Pow(Double.Parse(textBox96.Text.ToString()),
Double.Parse(textBox114.Text.ToString()))))) / 2.303;
            LogKrLogkeq = LogKr - LogKeq;
            textBox116.Text = Math.Pow(10, LogKrLogkeq).ToString();
        }

        private void button5_Click(object sender, EventArgs e)
        {
            String charstore;
            System.Windows.Forms.SaveFileDialog       saveFileDialog1       =       new
System.Windows.Forms.SaveFileDialog();
            saveFileDialog1.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";
            saveFileDialog1.FilterIndex = 2;
            saveFileDialog1.RestoreDirectory = true;
            if (saveFileDialog1.ShowDialog() == System.Windows.Forms.DialogResult.OK)
            {
                charstore = saveFileDialog1.FileName;
                System.IO.StreamWriter file = new System.IO.StreamWriter(charstore);
                file.WriteLine(textBox1.Text);
                file.WriteLine(textBox2.Text);
                file.WriteLine(textBox3.Text);
                file.WriteLine(textBox4.Text);
                file.WriteLine(textBox5.Text);
                file.WriteLine(textBox6.Text);
                file.WriteLine(textBox7.Text);
                file.WriteLine(textBox8.Text);
                file.WriteLine(textBox9.Text);
                file.WriteLine(textBox10.Text);
                file.WriteLine(textBox11.Text);
                file.WriteLine(textBox12.Text);
                file.WriteLine(textBox13.Text);
                file.WriteLine(textBox14.Text);
                file.WriteLine(textBox15.Text);
                file.WriteLine(textBox16.Text);
                file.WriteLine(textBox17.Text);
                file.WriteLine(textBox18.Text);
                file.WriteLine(textBox19.Text);
                file.WriteLine(textBox20.Text);
                file.WriteLine(textBox21.Text);
                file.WriteLine(textBox22.Text);
```

```
file.WriteLine(textBox23.Text);
file.WriteLine(textBox24.Text);
file.WriteLine(textBox25.Text);
file.WriteLine(textBox26.Text);
file.WriteLine(textBox27.Text);
file.WriteLine(textBox28.Text);
file.WriteLine(textBox29.Text);
file.WriteLine(textBox30.Text);
file.WriteLine(textBox31.Text);
file.WriteLine(textBox32.Text);
file.WriteLine(textBox33.Text);
file.WriteLine(textBox34.Text);
file.WriteLine(textBox35.Text);
file.WriteLine(textBox36.Text);
file.WriteLine(textBox37.Text);
file.WriteLine(textBox38.Text);
file.WriteLine(textBox39.Text);
file.WriteLine(textBox40.Text);
file.WriteLine(textBox41.Text);
file.WriteLine(textBox42.Text);
file.WriteLine(textBox43.Text);
file.WriteLine(textBox44.Text);
file.WriteLine(textBox45.Text);
file.WriteLine(textBox46.Text);
file.WriteLine(textBox47.Text);
file.WriteLine(textBox48.Text);
file.WriteLine(textBox49.Text);
file.WriteLine(textBox50.Text);
file.WriteLine(textBox51.Text);
file.WriteLine(textBox52.Text);
file.WriteLine(textBox53.Text);
file.WriteLine(textBox54.Text);
file.WriteLine(textBox55.Text);
file.WriteLine(textBox56.Text);
file.WriteLine(textBox57.Text);
file.WriteLine(textBox58.Text);
file.WriteLine(textBox59.Text);
file.WriteLine(textBox60.Text);
file.WriteLine(textBox61.Text);
file.WriteLine(textBox62.Text);
file.WriteLine(textBox63.Text);
file.WriteLine(textBox64.Text);
file.WriteLine(textBox65.Text);
file.WriteLine(textBox66.Text);
file.WriteLine(textBox67.Text);
file.WriteLine(textBox68.Text);
file.WriteLine(textBox69.Text);
file.WriteLine(textBox70.Text);
file.WriteLine(textBox71.Text);
file.WriteLine(textBox72.Text);
file.WriteLine(textBox73.Text);
file.WriteLine(textBox74.Text);
file.WriteLine(textBox75.Text);
file.WriteLine(textBox76.Text);
file.WriteLine(textBox77.Text);
file.WriteLine(textBox78.Text);
file.WriteLine(textBox79.Text);
file.WriteLine(textBox80.Text);
file.WriteLine(textBox81.Text);
file.WriteLine(textBox82.Text);
file.WriteLine(textBox83.Text);
file.WriteLine(textBox84.Text);
```

```
                file.WriteLine(textBox85.Text);
                file.WriteLine(textBox86.Text);
                file.WriteLine(textBox87.Text);
                file.WriteLine(textBox88.Text);
                file.WriteLine(textBox89.Text);
                file.WriteLine(textBox90.Text);
                file.WriteLine(textBox91.Text);
                file.WriteLine(textBox92.Text);
                file.WriteLine(textBox93.Text);
                file.WriteLine(textBox94.Text);
                file.WriteLine(textBox95.Text);
                file.WriteLine(textBox96.Text);
                file.WriteLine(textBox97.Text);
                file.WriteLine(textBox98.Text);
                file.WriteLine(textBox99.Text);
                file.WriteLine(textBox100.Text);
                file.WriteLine(textBox101.Text);
                file.WriteLine(textBox102.Text);
                file.WriteLine(textBox103.Text);
                file.WriteLine(textBox104.Text);
                file.WriteLine(textBox105.Text);
                file.WriteLine(textBox106.Text);
                file.WriteLine(textBox107.Text);
                file.WriteLine(textBox108.Text);
                file.WriteLine(textBox109.Text);
                file.WriteLine(textBox110.Text);
                file.WriteLine(textBox111.Text);
                file.WriteLine(textBox112.Text);
                file.WriteLine(textBox113.Text);
                file.WriteLine(textBox114.Text);
                file.WriteLine(textBox115.Text);
                file.WriteLine(textBox116.Text);
                file.Close();
            }
        }

        private void button6_Click(object sender, EventArgs e)
        {
            String myRead;
            System.Windows.Forms.OpenFileDialog       openFileDialog1       =       new
System.Windows.Forms.OpenFileDialog();
            openFileDialog1.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";
            openFileDialog1.FilterIndex = 2;
            openFileDialog1.RestoreDirectory = true;
            if (openFileDialog1.ShowDialog() == System.Windows.Forms.DialogResult.OK)
            {
                myRead = openFileDialog1.FileName;
                System.IO.StreamReader file = new System.IO.StreamReader(myRead);
                textBox1.Text = file.ReadLine();
                textBox2.Text = file.ReadLine();
                textBox3.Text = file.ReadLine();
                textBox4.Text = file.ReadLine();
                textBox5.Text = file.ReadLine();
                textBox6.Text = file.ReadLine();
                textBox7.Text = file.ReadLine();
                textBox8.Text = file.ReadLine();
                textBox9.Text = file.ReadLine();
                textBox10.Text = file.ReadLine();
                textBox11.Text = file.ReadLine();
                textBox12.Text = file.ReadLine();
                textBox13.Text = file.ReadLine();
                textBox14.Text = file.ReadLine();
```

```
textBox15.Text = file.ReadLine();
textBox16.Text = file.ReadLine();
textBox17.Text = file.ReadLine();
textBox18.Text = file.ReadLine();
textBox19.Text = file.ReadLine();
textBox20.Text = file.ReadLine();
textBox21.Text = file.ReadLine();
textBox22.Text = file.ReadLine();
textBox23.Text = file.ReadLine();
textBox24.Text = file.ReadLine();
textBox25.Text = file.ReadLine();
textBox26.Text = file.ReadLine();
textBox27.Text = file.ReadLine();
textBox28.Text = file.ReadLine();
textBox29.Text = file.ReadLine();
textBox30.Text = file.ReadLine();
textBox31.Text = file.ReadLine();
textBox32.Text = file.ReadLine();
textBox33.Text = file.ReadLine();
textBox34.Text = file.ReadLine();
textBox35.Text = file.ReadLine();
textBox36.Text = file.ReadLine();
textBox37.Text = file.ReadLine();
textBox38.Text = file.ReadLine();
textBox39.Text = file.ReadLine();
textBox40.Text = file.ReadLine();
textBox41.Text = file.ReadLine();
textBox42.Text = file.ReadLine();
textBox43.Text = file.ReadLine();
textBox44.Text = file.ReadLine();
textBox45.Text = file.ReadLine();
textBox46.Text = file.ReadLine();
textBox47.Text = file.ReadLine();
textBox48.Text = file.ReadLine();
textBox49.Text = file.ReadLine();
textBox50.Text = file.ReadLine();
textBox51.Text = file.ReadLine();
textBox52.Text = file.ReadLine();
textBox53.Text = file.ReadLine();
textBox54.Text = file.ReadLine();
textBox55.Text = file.ReadLine();
textBox56.Text = file.ReadLine();
textBox57.Text = file.ReadLine();
textBox58.Text = file.ReadLine();
textBox59.Text = file.ReadLine();
textBox60.Text = file.ReadLine();
textBox61.Text = file.ReadLine();
textBox62.Text = file.ReadLine();
textBox63.Text = file.ReadLine();
textBox64.Text = file.ReadLine();
textBox65.Text = file.ReadLine();
textBox66.Text = file.ReadLine();
textBox67.Text = file.ReadLine();
textBox68.Text = file.ReadLine();
textBox69.Text = file.ReadLine();
textBox70.Text = file.ReadLine();
textBox71.Text = file.ReadLine();
textBox72.Text = file.ReadLine();
textBox73.Text = file.ReadLine();
textBox74.Text = file.ReadLine();
textBox75.Text = file.ReadLine();
textBox76.Text = file.ReadLine();
```

```
                textBox77.Text = file.ReadLine();
                textBox78.Text = file.ReadLine();
                textBox79.Text = file.ReadLine();
                textBox80.Text = file.ReadLine();
                textBox81.Text = file.ReadLine();
                textBox82.Text = file.ReadLine();
                textBox83.Text = file.ReadLine();
                textBox84.Text = file.ReadLine();
                textBox85.Text = file.ReadLine();
                textBox86.Text = file.ReadLine();
                textBox87.Text = file.ReadLine();
                textBox88.Text = file.ReadLine();
                textBox89.Text = file.ReadLine();
                textBox90.Text = file.ReadLine();
                textBox91.Text = file.ReadLine();
                textBox92.Text = file.ReadLine();
                textBox93.Text = file.ReadLine();
                textBox94.Text = file.ReadLine();
                textBox95.Text = file.ReadLine();
                textBox96.Text = file.ReadLine();
                textBox97.Text = file.ReadLine();
                textBox98.Text = file.ReadLine();
                textBox99.Text = file.ReadLine();
                textBox100.Text = file.ReadLine();
                textBox101.Text = file.ReadLine();
                textBox102.Text = file.ReadLine();
                textBox103.Text = file.ReadLine();
                textBox104.Text = file.ReadLine();
                textBox105.Text = file.ReadLine();
                textBox106.Text = file.ReadLine();
                textBox107.Text = file.ReadLine();
                textBox108.Text = file.ReadLine();
                textBox109.Text = file.ReadLine();
                textBox110.Text = file.ReadLine();
                textBox111.Text = file.ReadLine();
                textBox112.Text = file.ReadLine();
                textBox113.Text = file.ReadLine();
                textBox114.Text = file.ReadLine();
                textBox115.Text = file.ReadLine();
                textBox116.Text = file.ReadLine();
                file.Close();
            }
        }
    }
}
```

3. Save File Examples

Name of file: N2
1
25
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)

fugacity
17.9348751637485
24425.2962062259
12212.6481031129
24407.3613310621
0.00073427462301059
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
9.99900009999E-05
0
0

0.0835514872859433
0
0.916348522713057
0
0
0.000178880485690064
0
0
0.0793378953621404
0
0.92048322415217
0
0
2.44221992963484
0
0
2040.75088641287
0
22382.1030998834
0
0
0.000100684136591245
0
0
0.0829777953083594
0
0.902566890501131
0
0.992677644350632
0.999438468057453
0.999843701735916
0.999955978732042
0.992232189657145
0.99507318241478
0.984952723209613
1.00053754723327
fugacity°
0
0.75
0
0
1
0
0
0
stoechio react
0
0
0
0.5

0
0
1.5
0
stoechio prod
0

Name of file: HCN
400
300
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
15.4554220357796
44.8892313619191
22.4446156809596
29.4338093261395
0.344301329447377
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021

0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
0.000549994500054999
0
0.00652993470065299
0
0.00742992570074299
0.808831911680883
0.176658233417666
0
0.00114177946692832
0
0.0118356957396437
0
0.0211568508037137
0.942824720070829
0.0230409539188858
0
0.0246654566355312
0
0.292814804652348
0
0.333361491855866
36.3175880836934
7.92080152508192
0
1.0120527802123
0
19.152398530604
0
6.76305450259368
87.5218775924109
456.736037388385
316.034896966614
456.34029246617
487.847232288503
487.682749622029

312.887070592072
401.950821138628
88.5223393281903
462.143585832929
fugacity°
0
0
0
0.5
0
1
0.5
0
stoechio react
0
1.25
0
0
0
0
0
0
stoechio prod
0.145210896321047

Name of file: O2
400
300
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
15.4554220357796
44.8892313619191
22.4446156809596
29.4338093261395
0.344301329447377
H2
H2O
CO2
NH3
N2
CO
O2
H2S

89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
0.000549994500054999
0
0.00652993470065299
0
0.00742992570074299
0.808831911680883
0.176658233417666
0
0.00114177946692832
0
0.0118356957396437
0
0.0211568508037137
0.942824720070829
0.0230409539188858
0
0.0246654566355312

0
0.292814804652348
0
0.333361491855866
36.3175880836934
7.92080152508192
0
1.0120527802123
0
19.152398530604
0
6.76305450259368
87.5218775924109
456.736037388385
316.034896966614
456.34029246617
487.847232288503
487.682749622029
312.887070592072
401.950821138628
88.5223393281903
462.143585832929
fugacity°
0
0
0
0
0
0
0
2
stoechio react
0
0
0
0
0
0
0
2
0
stoechio prod
1.00080789890749

Name of file: acridineorange
140
250
constituent
Pc (bars)
Tc (K)
wc

M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
16.150858436
36.9177038997778
18.4588519498889
20.7668454637778
0.437482744859905
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158

0
0.00604
0
0.00645
0
0.0073
0.84245
0.13776
0
0.0119990283344961
0
0.0111874195861474
0
0.0198918268817151
0.939727802713557
0.0171939224840841
0
0.222754780933472
0
0.237851227979916
0
0.269356017142084
31.108414738907
5.07932713481533
0
14.4700443466546
0
29.2888279419566
0
6.31499975803889
38.1971306713667
559.568897652196
119.041244088169
145.812246286453
149.89337330544
150.014785698117
119.138155101844
133.903938008627
38.3324460037422
147.749865973146
fugacity°
0
0
0
0.5
0
13
0
30.5
stoechio react

0
0
0
0
0
0
26
0
stoechio prod
0.8304294476058

Name of file: THC
125
180
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
22.719490874
49.5709480152632
24.7854740076316
26.8514571412632
0.458322702785602
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2

647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
0.0012
0
4E-05
0
0.26005
0.60913
0.12958
0
0.00169467882064477
0
4.93204714055601E-05
0
0.503739126174575
0.483019813906064
0.0114970606273103
0
0.0594237511602639
0
0.00198057643504615
0
12.8855984658457
30.2092796160463
6.41466560577584
0
1.05260022944594
0
0.0557456963106998
0
76.4755076735677
15.8932844560085
207.516664869284

96.067764070559
128.556972398995
132.519748685857
132.797421223562
95.2168124514369
112.114319273202
10.3927648193701
131.894744361607
fugacity°
0
0
0
0
0
21
0
55
stoechio react
0
0
0
0
0
0
40
0
stoechio prod
1.10358482211514

Name of file: morphine
125
180
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
22.4342822348224
46.5567603080886
23.2783801540443
24.1224780732663
0.481869487618207
H2
H2O
CO2
NH3

N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
0.001220012200122
0
0.028590285902859
0
0.228712287122871
0.627126271262713
0.114351143511435
0
0.00174484456572428
0
0.0357003226904216
0
0.44866763135123

0.503612344694399
0.0102748566982257
0
0.0567427559397842
0
1.32959453687539
0
10.6437615613008
29.2098770010139
5.31678445295873
0
1.21946283870016
0
47.094563843629
0
73.3071301282756
15.2704364071243
214.16609401934
96.067764070559
128.556972398995
132.519748685857
132.797421223562
95.2168124514369
112.114319273202
10.3927648193701
131.894744361607
fugacity°
0
0
0
0.5
0
17
0
40.5
stoechio react
0
0
0
0
0
0
31
0
stoechio prod
1.03848300529299

Name of file: cocaine
125
180

constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
22.4342822348224
46.5567603080886
23.2783801540443
24.1224780732663
0.481869487618207
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6
132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134

17.03040
44.0096
18.0158
2.0158
0
0.001220012200122
0
0.028590285902859
0
0.228712287122871
0.627126271262713
0.114351143511435
0
0.00174484456572428
0
0.0357003226904216
0
0.44866763135123
0.503612344694399
0.0102748566982257
0
0.0567427559397842
0
1.32959453687539
0
10.6437615613008
29.2098770010139
5.31678445295873
0
1.21946283870016
0
47.094563843629
0
73.3071301282756
15.2704364071243
214.16609401934
96.067764070559
128.556972398995
132.519748685857
132.797421223562
95.2168124514369
112.114319273202
10.3927648193701
131.894744361607
fugacity°
0
0
0
0.5
0

17
0
40.5
stoechio react
0
0
0
0
0
0
30
0
stoechio prod
0.706820526036225

Name of file: adenine
400
300
constituent
Pc (bars)
Tc (K)
wc
M (g/mol)
molar fraction
massic fraction
molar V (cm3/mol)
fugacity
15.4554220357796
44.8892313619191
22.4446156809596
29.4338093261395
0.344301329447377
H2
H2O
CO2
NH3
N2
CO
O2
H2S
89.4
50.5
35
33.9
113.33
73.8
221.2
12.97
373.2
154.6

132.9
126.2
405.40
304.2
647.30
33.3
0.1
0.021
0.049
0.04
0.25601
0.225
0.344
-0.215
34.0814
32.0852
28.0102
28.0134
17.03040
44.0096
18.0158
2.0158
0
0.000549994500054999
0
0.00652993470065299
0
0.00742992570074299
0.808831911680883
0.176658233417666
0
0.00114177946692832
0
0.0118356957396437
0
0.0211568508037137
0.942824720070829
0.0230409539188858
0
0.0246654566355312
0
0.292814804652348
0
0.333361491855866
36.3175880836934
7.92080152508192
0
1.0120527802123
0
19.152398530604

0
6.76305450259368
87.5218775924109
456.736037388385
316.034896966614
456.34029246617
487.847232288503
487.682749622029
312.887070592072
401.950821138628
88.5223393281903
462.143585832929
fugacity°
0
0
0
2.5
0
5
2.5
0
stoechio react
0
6.25
0
0
0
0
0
0
stoechio prod
6.45648327960573E-05

4. Conclusion

The molecule picture in the first page of this book was made with these codes. The pressure and temperature determination are 125 bar and 180°C. Also the codes was used to determine the pressure of 140 bar for acridine orange synthesis at 250°C.

To calculate PT traject with homogenisation temperature (Th) and composition of a fluid inclusion, do as following:

Put the Th and change pressure until there is a hole for a bar of the volume. Take the pressure minimum under the hole, or the gap of the volume (the low volume). This volume correspond of the fluid inclusion volume. It must be the same volume for the PT traject.

5. Use and Agreement Contract

Owner: Michael Andre Franiatte.

Contact: michael.franiatte@gmail.com.

Owning: All works from scratch of the owner.

Proof of Owning: Works published, and writings/speakings all over.

Requirements of Use: Pay the owner, quote the owner, agreement of the owner.

Availability of Works: Only under the shapes of the owner built, only for personal use.

Subjects of Claims: Works published by the owner on Google Play and Google Books.

Concerning Author Rights: Equations and codes from scratch of the owner, softwares built from it, all things of people arising from it.

End User License Agreement: A commercial license is required to use in personal manner. Do not redistributing in any manner, including by computer media, a file server, an email attachment, etc. Do not embedding in or linking it to another programs, source codes and assistances including internal applications, scripts, batch files, etc. Do not use for any kind of technical support including on customer or retailer computer, hardware or software development, research, discovery, teachery, talk, speech, write, etc. Do not use for win money or for commercialisation of any products arising from my programs, source codes and assistances. Do not use and do not copy the way it run in other programs, source codes and assistances. Do not use without pay me, quote me and my agreement. Do not steal or copy or reproduce or modify or peer or share. Do not use in other manner than personal. It stand for my programs, source codes and assistances or programs, source codes and assistances stealing or copying or reproducing or modifying or peering or sharing my programs, source codes, and assistances. If you aren't agree you shall not use.

Terms of License and Price: The present contract acceptance is required to use works of the owner and built from it in all kind of manner. The price for each user shall be defined with the owner by contacting him and this for each subject of works the owner claims. Each user shall contact the owner for asking his agreement. It can be refused by the owner depending who asking and the price defined. People don’t respecting the present contract shall not use the works of the owner.