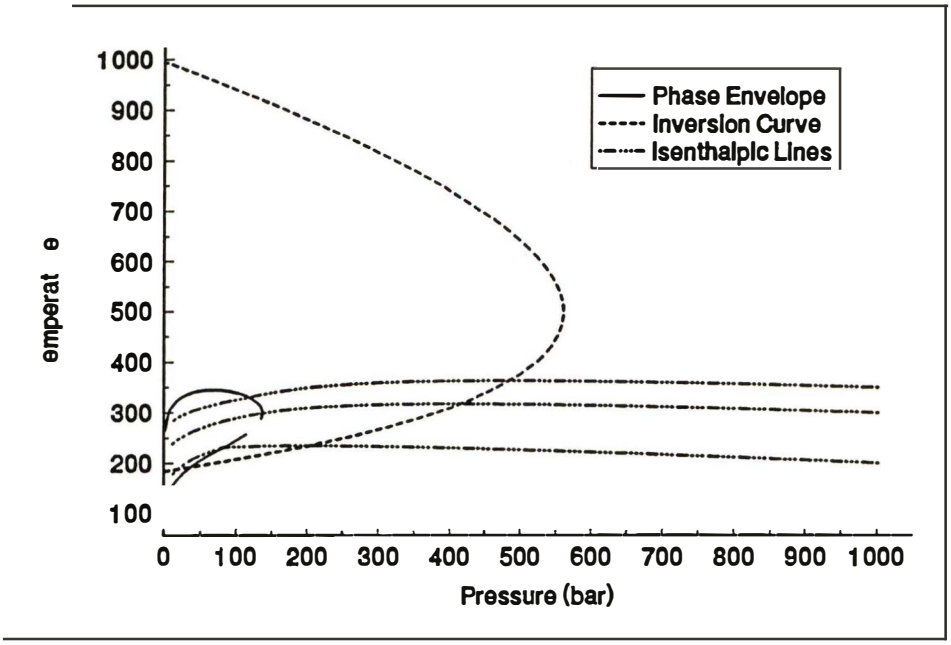
Joule-Thomson Expansion of Gas-Condensates

Predictions using Equations of State



November 1994

Walter G. Kortekaas

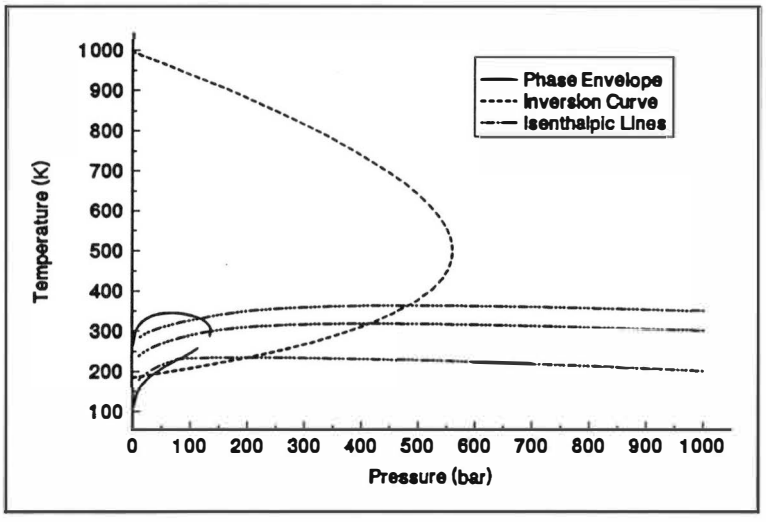
TU Delft

Technische Universiteit Delft

Faculteit der Scheikundige Technologie en der Materiaalkunde

Joule-Thomson Expansion of Gas-Condensates

Predictions using Equations of State



Master thesis performed in the period April through November 1994

W.G. Kortekaas

Supervisors:

Dr Ir C.J. Peters
Prof Dr Ir J. de Swaan Arons

November 1994

Laboratory of Applied Thermodynamics and Phase Equilibria
Faculty of Chemical Technology and Materials Science
Delft University of Technology

Preface

You are about to read the report on the investigation of Joule-Thomson expansion of high-pressure high-temperature gas-condensates, which I performed to obtain my masters degree in chemical engineering at the Delft University of Technology.

During these last eight months of my studies I have worked on this subject at the Laboratory of Applied Thermodynamics and Phase Equilibria headed by Prof Dr Ir J. de Swaan Arons. After my first experience with thermodynamics I was impressed by the variation of problems that thermodynamics provides a solution for, and when I had the opportunity to combine my interest for thermodynamics with my interest for petroleum engineering, I did not have to hesitate one moment.

I have enjoyed my stay here at the laboratory very much and it has been a great learning experience to me. Therefore, I would like to thank everyone who made my stay here a pleasant one. I would like express my special thanks to Prof Dr Ir J. de Swaan Arons for giving me the opportunity to perform my work here at the laboratory and Dr Ir Cor Peters for his enthusiastic supervision and valuable comments. Further more I would like to thank Peter Cornelisse for helping me out with some problems on Fortran and numerical receipts, Mark Wijtkamp for his inexhaustible supply of coffee coins, Louwe Kooijmans for allhis latest updates on software, and Peter Jan Smits for his introductory course in stock exchange and the fact that he is the only one who knows how to appreciate the Vegas mode.

Walter Kortekaas

November 1994

Summary

This work deals with the Joule-Thomson expansion of high-pressure high-temperature gas-condensates. After observing unexpected heating of the produced condensate at pressure relief in a North Sea gas-condensate reservoir, it seemed to be necessary to investigate the possibility of the Joule-Thomson inversion effect to occur in these reservoir fluids.

No experimental data is available on Joule-Thomson effects in gas-condensate reservoirs and, therefore, this work has an entirely theoretical basis. Calculated Joule-Thomson coefficients, inversion curves, and isenthalpic lines for some lighter constituents of gas-condensate mixtures, e.g. nitrogen, carbon dioxide, and methane, were compared with experimental data to get insight in the applicability of cubic equations of state. The equations used are the Redlich-Kwong, the Soave-Redlich-Kwong and the Peng-Robinson equations of state.

The RK equation gave the best overall predictions for the Joule-Thomson coefficient despite its simplicity, but failed in the characterization of the inversion curve because the locus of this curve is strongly affected by the acentric factor, which is not taken into account by the RK equation. Both the SRK and the PR equation gave satisfactory results for the lower part of the inversion curve. The SRK equation gives the better description of the peak of the curve (at high pressures) and the PR equation the better description of the upper part (at high temperatures).

Isenthalpic expansion was modelled for several gas-condensate systems found in the literature with varying compositions. These simulations showed that it is possible to observe heating of gas-condensates at expansion. This heating is not caused by high reservoir temperatures, but by high reservoir pressures. Typical temperature increases encountered are in the range of 10-30 K for reservoir pressures of 1000 bar. Increasing reservoir pressures gave significantly more heating. Calculated temperature increases in reservoirs with pressures of 2000 bar varied from 55 K to 80 K. Although the reservoir temperature has no influence on occurrence of heating itself, it does influence the amount of heating, i.e. with decreasing reservoir temperatures, maximum possible temperature rises increase. For reservoir conditions of 300 K and 1000 bar a temperature rise of 21 K was calculated and for reservoir conditions of 500 K and 1000 bar this was 9 K. Besides reservoir conditions, heating is strongly affected by the composition of the reservoir fluid.

Increasing molecular weight of the fluid causes more heating, which confirms that besides the reservoir pressure, the amount of heavy constituents in gas-condensates plays a dominant role in the heating of reservoir fluids. Calculated temperature increases varied from 11 K (MW = 19.8 g/mole) up to 30 K (MW = 47.8 g/mole) for reservoir conditions of 400 K and 1000 bar.

To test the reliability of the calculated results and to draw conclusions upon the applicability of the used equations of state and characterization procedures, it is strongly recom-mended to obtain isenthalpic expansion data of gas-condensate systems for comparison. This could be achieved by cooperation with institutions or companies were data is available, but still confidential, or by own experimental work in the near future.

Contents

1. Introduction 1
1.1 Problem description 1
1.2 Objectives of the present work 2

2. An Introduction to the Joule-Thomson Expansion 5
2.1 The Joule-Thomson experiment 5
2.2 The Joule-Thomson coefficient and inversion curve 6

3. Cubic Equations of State 9
3.1 Cubic equations of state 9
3.2 Mixing rules for cubic equations of state 12
3.3 Calculation of molar volume 13
3.4 Calculation of enthalpy and temperature derivative 13
3.5 Calculation of fugacity coefficients 15

4. Calculation of the Joule-Thomson Coefficient 17
4.1 Basic relations 17
4.2 Analytical expressions for the Joule-Thomson coefficient 19
4.3 Calculated results and comparison 19

5. Calculation of the Inversion Curve 27
5.1 Calculation of the inversion curve from equations of state 27
5.2 Calculated results and comparison 31
5.3 Why inversion? 35

6. Isenthalpic Expansion Calculations 37
6.1 Isenthalpic expansion of pure component systems 37
6.1.1 Computation scheme for pure component systems 37
6.1.2 Calculated results and comparison 39
6.2 Isenthalpic expansion of multi-component systems 39
6.2.1 Computation scheme for multi-component systems 39
6.2.2 The Gibbs tangent plane theory 43

6.2.3 The isenthalpic flash problem 45

7. Joule-Thomson Effects in Gas-Condensate Reservoirs 47

7.1 Gas-condensate reservoirs 47

7.2 Characterization of the heavy fraction 50

7.3 Reduction of the number of components 53

7.4 Isenthalpic expansion and inversion curves 54

7.5 Sensitivity study 57

7.6 Recapitulation 63

8. Conclusions and Recommendations 65

List of Symbols and Abbreviations 69

References 73

Appendices 77

1. Introduction

1.1 Problem description

With improving technology, drilling for oil at considerable depth, where high pressures and high temperatures are encountered, became possible. Drilling into these high-pressure high-temperature (HPHT) oil reservoirs can give rise to additional problems which are not present in more conventional reservoirs. Recently is was observed that during production of a North Sea gas-condensate reservoir, heating of the reservoir fluid occurred. This observation seems to be in conflict with the expected behaviour of reservoir fluids, where one would anticipate a cooling effect at expansion.
To find an explanation for this contradictory behaviour, an investigation of the thermodynamics of gas-condensate reservoir fluids at high pressures and high temperatures seems to be necessary. It is suspected that due to the high pressures (over 1000 bar) and high temperatures (100-200 °C) in these gas-condensate reservoirs, temperatures can increase at expansion due to the Joule-Thomson inversion effect.

The Joule-Thomson expansion, or throttling, is a process for which the enthalpy of a fluid remains constant. Consequently the temperature will change when pressure drops and this is generally referred to as the Joule-Thomson effect. Temperatures can either decrease or increase, depending on the initial pressure, initial temperature, and composition of the fluid. The region where cooling takes place is separated from the region where heating takes place by the inversion curve. Heating of a fluid at expansion is therefore called the Joule-Thomson inversion effect.
The Joule-Thomson inversion effect is not an unknown phenomenon. In refrigeration and liquefaction processes, in which the Joule-Thomson expansion is frequently used, it is well-known that fluids can increase in temperature at expansion. For example, expanding hydrogen at ambient temperature will result in a temperature increase. To liquefy hydrogen, it must therefore be pre-cooled until the temperature is below a specific value, before cooling, and finally liquefaction, can take place at expansion.
However, in reservoir engineering the Joule-Thomson inversion effect is not very common. Under normal reservoir conditions the temperature of the reservoir fluid will decrease at expansion, but, as will be shown later, at high temperatures and, more important, at high reservoir pressures, it is possible for fluid temperatures to increase. In the open literature little is reported regarding temperature increase in HPHT oil reservoirs. Baker and Price

(1990) reported temperature increases of some 15 to 20 °F during a series of production tests for a UK Central North Sea HPHT well. Typical reservoir conditions encountered were in the range of 16000 psia (1100 bar) and 400 °F (200 °C). Unfortunately, due to confidentiality requirements, fluid composition and reservoir modelling results could not be published.

It is clear that if for these kind of HPHT reservoirs no precautions are taken, unexpected high surface temperatures may result when reservoir fluids are brought to surface either during production testing or in a well pressure control operation. To ensure a safe operation and to ensure that the equipment is specified according to downhole temperatures, it is required that pressure and temperature profiles can be accurately estimated for future productions.

1.2 Objectives of the present work

The objectives of this work are to examine the Joule-Thomson expansion of gas-condensates and to check the possibility of Joule-Thomson inversion to occur. The isenthalpic expansion of reservoir fluids must therefore be modelled to investigate possible temperature increases and to examine in what way reservoir pressure, temperature, and fluid composition influence temperature increase.

In order to reach these goals, an investigation will take place on the Joule-Thomson inversion effect. Answers are sought for questions like: why does inversion take place, under what circumstances does it take place and is it possible to accurately describe this behaviour. Inversion curves will be calculated and the isenthalpic expansion will be modelled. Modelling requires calculation of fluid properties such as volume, enthalpy and fugacities, and isenthalpic flash calculations in the two-phase region. For calculation purposes three, in engineering applications frequently used, cubic equations of state will be tested: the Redlich-Kwong, the Soave-Redlich Kwong and the Peng-Robinson equation.

Because experimental data are very scarce or, due to confidentiality reasons, not available at all, this work will have an entirely theoretical and predictable basis. It is therefore important to indicate how accurate the calculated results are. A sensitivity study is performed to examine the influence of fluid composition, reservoir conditions, equations of state and characterization procedures for gas-condensate mixtures on the Joule-Thomson effect.

This report is arranged as follows. First of all an introduction into the Joule-Thomson expansion and inversion effects will be given in chapter 2. After describing the used cubic equations of state in chapter 3, chapter 4 and 5 will deal with the calculation of Joule-Thomson coefficients and inversion curves, respectively. In chapter 6 the isenthalpic expansion of pure and multi-component mixtures will be described as well as the necessary computer algorithms needed for calculation purposes. After these preliminary steps, chapter 7 will deal with gas-condensate systems. This includes a description of gas-condensates, several characterization procedures, isenthalpic expansion calculations and a sensitivity study. Finally, conclusions and recommendations are given.

2. An Introduction to the Joule-Thomson Expansion

This chapter deals with the basic principles of the Joule-Thomson expansion. Necessary relations will be given and concepts such as isenthalpic expansion, Joule-Thomson effect, Joule-Thomson coefficient, and inversion curves will be explained.

2.1 The Joule-Thomson experiment

Figure 2.1 gives a schematic representation of the classical porous plug experiment. The original experiments were first performed by Thomson (later known as lord Kelvin) and Joule in 1853.

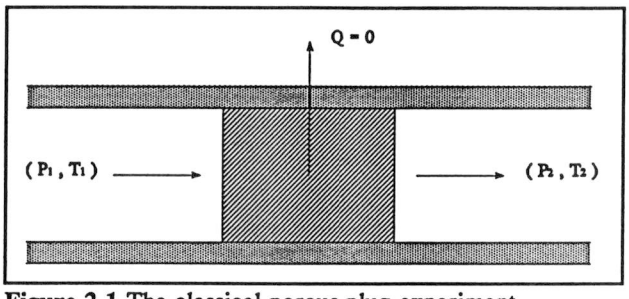


Figure 2.1 The classical porous plug experiment

In the Joule-Thomson experiment a pressurized fluid, at initial conditions P_1 and T_1, flows under steady state circumstances through a thermally insulated porous plug. Due to the porous plug a pressure drop is created across the restriction, causing the fluid to expand. The following energy balance is valid for this system:

$$\Delta H + \frac{1}{2}\Delta v^2 + g\Delta z = Q - W \qquad (2.1)$$

Because flow is steady state, it may be assumed that the changes in kinetic energy $\frac{1}{2}\Delta v^2$ and potential energy $g\Delta z$ are negligible. The plug is fully insulated from the surroundings and therefore the heat effect Q is zero. Finally no work W is performed on or by the system and equation 2.1 is simply reduces to:

$$\Delta H = 0 \qquad (2.2)$$

The inlet and outlet enthalpy are equal and the expansion is thus *isenthalpic*. An isenthalpic expansion is also called a *throttling process*. Based on the original experiments by

Joule and Thomson, expansion of a fluid from a higher to a lower pressure at constant enthalpy, is frequently called the *Joule-Thomson effect.*

When an upstream fluid at pressure P_1 and temperature T_1 crosses the plug the fluid will expand to a downstream pressure P_2 and a temperature T_2. These downstream conditions can be measured from the Joule-Thomson experiment. The initial values for P_1 and T_1 are set and the downstream pressure can be varied. For each setting of P_2 the downstream temperature T_2 can then be measured. Because expansion under these circumstances is isenthalpic, a set of PT-data is obtained for which the equilibrium states have the same enthalpy. The data is plotted in a PT-graph and a constant enthalpy line can be drawn. By choosing different initial values for T_1 or P_1 the experiment can be repeated and a new line of constant enthalpy is derived. A plot can then be made like figure 2.2, which shows a whole range of constant enthalpy lines.

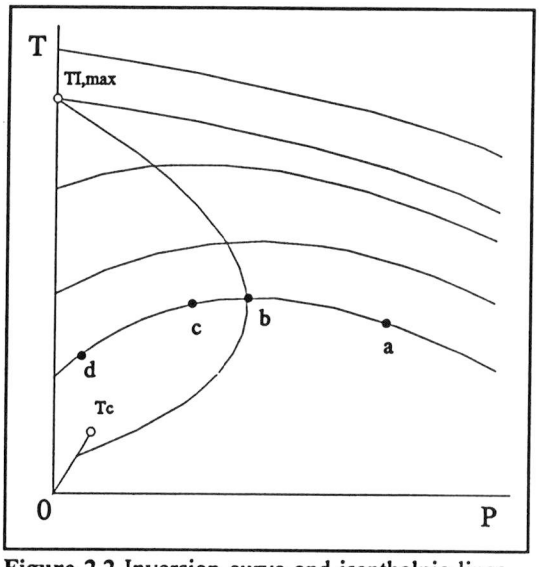


Figure 2.2 Inversion curve and isenthalpic lines

2.2 The Joule-Thomson coefficient and inversion curve

As figure 2.2 shows, throttling of a fluid causes a change in temperature. This change of temperature is given by the Joule-Thomson coefficient μ_{JT}, which is defined as follows:

$$\mu_{JT} = \left(\frac{\partial T}{\partial P}\right)_H \tag{2.3}$$

The Joule-Thomson coefficient is a function of temperature and pressure at constant enthalpy. In a PT-plot the value of μ_{JT} is represented by the slope of a constant enthalpy line. A number of constant enthalpy lines in figure 2.2 have a state of maximum temperature and in these points the slope and thus μ_{JT} have a value of zero. Temperatures for which μ_{JT} is zero are called inversion temperatures. By drawing a line through all these inversion temperatures a curve is obtained, which is called the *inversion curve*. Because a given pressure can correspond with two different states, one speaks of a lower and an upper inversion temperature. The inversion curve is physically very important because it divides the PT-plot into two area's. One for which values of μ_{JT} are positive and one for which values are negative. The region inside the inversion curve, where the value of μ_{JT} is positive, is called the *cooling region* because a pressure drop causes a decrease in temperature. The region outside the inversion curve, for which μ_{JT} is negative, is called the *heating region* because the temperature increases at pressure drop.
When throttling a fluid, the final temperature may be higher than, equal to, or lower than the initial temperature, depending on the initial values of P_1, T_1, and the final value P_2. For example when a fluid is at an initial state corresponding to point **a** in figure 2.2, expansion to the inversion curve (point **b**) will result in heating of the fluid. When further expansion takes place to point **c** some cooling will occur, but this is not sufficient to lower the temperature to that of the initial state. However, if expansion to point **d** is possible, sufficient cooling will occur to bring the final temperature to a lower value than that of the initial state.

At high temperatures, as figure 2.2 shows, a maximum inversion temperature $T_{I,max}$ exists. At temperatures above this maximum inversion temperature $T_{I,max}$, enthalpy lines never cross the inversion curve and heating will always occur at expansion. In table 2.1 values of $T_{I,max}$ are given for some components. At low temperatures the inversion curve intersects the vapour pressure line and stops. By extrapolation a minimum inversion temperature can be obtained, which can be convenient, but is physically incorrect. Koeppe (1959) showed that the vapour pressure curve can be considered as a prolongation of the inversion curve at low temperatures, though this extended definition of the inversion curve must be accepted with some precaution because the relation $\mu_{JT} = 0$ no longer holds.

A practical example of inversion is the contrasting behaviour of hydrogen and nitrogen at

ambient temperature. The maximum inversion temperature of hydrogen, 204 K, is below ambient temperature and, therefore, the negative Joule-Thomson coefficient causes hydrogen to heat at expansion. On the other hand nitrogen has a maximum inversion temperature of 607 K and it is observed that nitrogen cools at expansion at ambient temperature. In order to cool hydrogen at expansion, pre-cooling must take place until the temperature is below 204 K, i.e below its maximum inversion temperature.

Table 2.1 Maximum inversion temperatures †

Component	$T_{I,max}$ (K)
CH_4	939 ‡
CO_2	1275
CO	644
N_2	607
H_2	204
He	43

† Zemansky and Dittman (1981)

‡ Ullmann's (1985)

3. Cubic Equations of State

To model the throttling process for gas-condensate systems, it is necessary to use PVT-relationships which can adequately describe the behaviour of these reservoir fluids. Once a PVT-relationship is available it will be possible to calculate Joule-Thomson coefficients, inversion curves and fluid properties such as molar volume, enthalpy and fugacities, which are needed in the isenthalpic expansion calculations. This chapter will only deal with the cubic equations of state themselves and the derivation of the formula's to calculate fluid properties. Calculation of Joule-Thomson coefficients and inversion curves will be treated in the following chapters.

3.1 Cubic equations of state

In this work calculations will be performed using cubic equations of state. Cubic equations of state (EoS) have gained much popularity over the years for engineering computations. The reason for this is that cubic EoS give relatively simple expression for the thermodynamic properties and phase equilibrium relationships, little computer time is needed to perform calculations, and the description of mixtures is very often quantatively not lesser than with more complicated EoS. The cubic EoS tested in this work are frequently used for engineering purposes: the Redlich-Kwong EoS (1949), the Soave-Redlich-Kwong EoS (1972) and the Peng-Robinson EoS (1976). A well-known disadvantage of cubic EoS is that they cannot accurately predict liquid densities. This disadvantage is taken into account.

Cubic EoS are often used for oil and gas PVT and phase equilibrium computations. Pedersen et al. (1989) use the SRK equation for all their oil/gas computations. They modelled PVT-experiments and also compared experimental and calculated thermodynamic properties and found that no other cubic EoS tested, yielded better results than the SRK equation. Nagy (1991) used a modification of the PR equation for calculation of Joule-Thomson effects in natural gas systems. Calculations were performed for the isenthalpic throttling of a natural gas system and compared with experimental data.

The first equation of state that represented reasonably well both gas and liquid phases was proposed by Van der Waals in 1873. Current cubic EoS are all based on this well-known equation. They share the common characteristic of a separated repulsive and attractive contribution to the pressure:

$$P = P_{rep} - P_{attr} \tag{3.1}$$

The Redlich-Kwong EoS was the first Van der Waals based cubic EoS that was widely used for engineering purposes. The repulsive term is the same as for the van der Waals equation but the attractive term was changed to obtain better results:

$$P = \frac{RT}{V-b} - \frac{a}{\sqrt{T}V(V+b)} \tag{3.2}$$

To improve the reproduction of vapour pressures, Soave (1972) proposed a temperature dependence of the a-parameter:

$$P = \frac{RT}{V-b} - \frac{a(T)}{V(V+b)} \tag{3.3}$$

While an appropriate temperature dependence of the a-parameter is sufficient to accurately calculate vapour pressures, modifications of the P(V) dependence are necessary to improve the prediction of volumetric properties. Peng and Robinson (1976) changed the form of the attractive contribution of the RK EoS:

$$P = \frac{RT}{V-b} - \frac{a(T)}{V(V+b)+b(V-b)} \tag{3.4}$$

The two parameters in the equations, a and b, represent the intermolecular forces and the hard-sphere volume of the molecules, respectively. Both parameters can be determined by satisfying the critical conditions:

$$\left(\frac{\partial P}{\partial V}\right)_T = \left(\frac{\partial^2 P}{\partial V^2}\right)_T = 0 \tag{3.5}$$

For the RK EoS the pure component a-parameter is independent of temperature and is calculated as follows:

$$a_i = \Omega_a \frac{R^2 T_{ci}^{2.5}}{P_{ci}} \tag{3.6}$$

Usually the square root of the temperature in equation 3.2 is put into the denominator of equation 3.6 and the a-parameter changes into:

$$a_i = \Omega_a \frac{R^2 T_{ci}^{2.5}}{\sqrt{T} P_{ci}} \tag{3.7}$$

The advantage of using equation 3.7 instead of 3.6, is that functional form of the RK and the SRK EoS become identical. For the SRK and the PR EoS the a-parameter is a function of temperature:

$$a_i(T) = a_{ci}\,\alpha_i(T) \tag{3.8}$$

The temperature independent contribution a_{ci} is quite similar to that of the RK equation:

$$a_{ci} = \Omega_a \frac{R^2 T_{ci}^2}{P_{ci}} \tag{3.9}$$

and the temperature dependent contribution $\alpha_i(T)$ is obtained from the following expression:

$$\alpha_i(T) = (1 + m_i(1 - T_{ri}^{0.5}))^2 \tag{3.10}$$

where m_i is a function of the acentric factor ω and T_{ri} is the reduced temperature of component i. For the SRK equation m_i is equal to:

$$m_i = 0.480 + 1.574\,\omega_i - 0.176\,\omega_i^2 \tag{3.11}$$

and for the PR equation the expression for m_i is as follows:

$$m_i = 0.37646 + 1.54226\,\omega_i - 0.26992\,\omega_i^2 \tag{3.12}$$

For all three cubic EoS the b-parameter is calculated from the following expression:

$$b_i = \Omega_b \frac{R T_{ci}}{P_{ci}} \tag{3.13}$$

For the RK and the SRK equation the values for the parameters Ω_a and Ω_b are equal to **0.42748** and **0.08664**, respectively. For the PR equation the following values are used: Ω_a = **0.457235** and Ω_b = **0.077796**. Values for the critical properties T_c and P_c, and for the acentric factor ω of some lighter constituents in petroleum mixtures are given in table 3.1. All values are taken from Reid et al. (1987).

Table 3.1 Physical properties of some common petroleum mixtures constituents

Component	M_w (g/mole)	T_c (K)	P_c (bar)	ω
N_2	28.013	126.2	33.9	0.039
CO_2	44.010	304.1	73.8	0.239
C_1	16.043	190.4	46.0	0.011
C_2	30.070	305.4	48.8	0.099
C_3	44.094	369.8	42.5	0.153
n-C_4	58.124	425.2	38.0	0.199
i-C_4	58.124	408.2	36.5	0.183
n-C_5	72.151	469.7	33.7	0.251
i-C_5	72.151	460.4	33.9	0.227
n-C_6	86.178	507.5	30.1	0.299

3.2 Mixing rules for cubic equations of state:

When calculations are performed for a mixture of N components, the a and b-parameter are calculated from the pure component parameters a_i and b_i using the following classical mixing rules:

$$a = \sum_i^N \sum_j^N x_i x_j (a_i a_j)^{0.5} (1 - k_{ij}) \tag{3.14}$$

$$b = \sum_i^N \sum_j^N x_i x_j (b_i b_j)^{0.5} (1 - l_{ij}) \tag{3.15}$$

where x_i is the mole fraction of component i and k_{ij} and l_{ij} are binary interaction parameters between components i and j. Values of k_{ij} are usually considered equal to zero for hydrocarbon-hydrocarbon interactions, and non-equal to zero for interactions between a hydrocarbon and a non-hydrocarbon or pairs of unlike non-hydrocarbons. Values of l_{ij} are all considered zero. Non-zero values for k_{ij} for use with the SRK and PR-equation of some lighter constituents of petroleum mixtures are given in appendix A. Values are taken from Reid et al. (1987). To calculate fluid properties and derivatives of fluid properties, the first and second derivative of a(T) with respect to temperature are needed:

$$\frac{\partial a(T)}{\partial T} = \sum_i^N \sum_j^N x_i\, x_j\, (a_{ci} a_{cj})^{0.5} \Big(- 0.5\, m_i\, (1 + m_j)\, (T_{ci}\, T)^{-0.5} - 0.5\, m_j\, (1 + m_i)\, (T_{cj}\, T)^{-0.5} + m_i m_j (T_{ci}\, T_{cj})^{-0.5} \Big) (1 - k_{ij}) \tag{3.16}$$

$$\frac{\partial^2 a(T)}{\partial T^2} = \sum_i^N \sum_j^N 0.25\, x_i\, x_j\, (a_{ci}\, a_{cj})^{0.5} \Big((m_i + m_i m_j)\, (T_{ci}^{-0.5}\, T^{-1.5}) + (m_j + m_i m_j)\, (T_{cj}^{-0.5}\, T^{-1.5}) \Big) (1 - k_{ij}) \tag{3.17}$$

3.3 Calculation of molar volume

To calculate the molar volume of a fluid for a given temperature and pressure, equations 3.2-3.4 are usually rewritten in terms of the compressibility factor Z. After some rearranging, the following equation is obtained for the RK and SRK equation:

$$Z^3 - Z^2 + (A - B + B^2)Z - AB = 0 \tag{3.18}$$

and for the PR equation:

$$Z^3 - (1 - B)Z^2 + (A - 2B - 3B^2)Z - (AB - B^2 - B^3) = 0 \tag{3.19}$$

where $Z = \frac{PV}{RT}$, $A = \frac{aP}{R^2 T^2}$ and $B = \frac{bP}{RT}$ for all three equations.

These third-order polynomials may have one or three real roots. If for a pure component or a single-phase mixture only one real root exists, this root equals the compressibility of that phase. If for a pure component three real roots exist, the smallest root is the compressibility of the liquid-phase and the largest root that of the gas-phase. For multi-phase mixtures the problem is more complex. For example, a vapour and liquid in equilibrium have different compositions and molar volumes for both phases are calculated separately using either equation 3.18 or 3.19 for SRK or PR, respectively. Compositions are calculated using flash procedures as will be described in chapter 6.

3.4 Calculation of enthalpy and temperature derivative

The enthalpy of a component, or a mixture of given components, may be calculated as the sum of two contributions, namely the ideal gas enthalpy H^{id} and the residual enthalpy H^{res}:

$$H = H^{id} + H^{res} \tag{3.23}$$

The ideal gas enthalpy can be calculated from the following expression:

$$H^{id} = \sum_i x_i H_i^{id} = \sum_i x_i \int_{T_{ref}}^{T} C_p^{id}\, dT \tag{3.24}$$

where T_{ref} is the a reference temperature which can be chosen arbitrarily, in this work 25 °C (298.15 K), and C_p^{id} is the ideal gas heat capacity which can be approximated by a third order polynomial function of temperature:

$$C_p^{id} = C_{p,1} + C_{p,2} T + C_{p,3} T^2 + C_{p,4} T^3 \tag{3.25}$$

Values for the coefficients $C_{p,1}$-$C_{p,4}$ for the lighter constituents are taken from Reid et al. (1987) and are tabulated in appendix A. Because coefficients are only tabulated up to C_{20}, the values of $C_{p,1}$-$C_{p,4}$ for heavier alkanes must be estimated. This is done by using the group contribution method of Joback, given by Reid et al. (1987). This method is shortly described in appendix A along with some examples.

The residual enthalpy ($H-H^0$) can be calculated from an cubic EoS using the following equation:

$$H - H^0 = (A - A^0) + T(S - S^0) + RT(Z - 1) \tag{3.26}$$

The residual Helmholtz energy ($A-A^0$) is calculated as follows:

$$A - A^0 = -\int_{\infty}^{V} \left(P - \frac{RT}{V} \right) dV - RT \ln \frac{V}{V^0} \tag{3.27}$$

and the residual entropy ($S-S^0$) is obtained using the equation:

$$S - S^0 = \int_{\infty}^{V} \left[\left(\frac{\partial P}{\partial T} \right)_V - \frac{R}{V} \right] dV + R \ln \frac{V}{V^0} \tag{3.28}$$

The necessary differentiations and integrations were performed using the program *Maple V Release 2.0*, and after some rearranging the following expression for the residual enthalpy are obtained, for the RK:

$$\frac{H - H^0}{RT} = Z - 1 - \frac{3a}{2b} \ln \left(1 + \frac{B}{Z} \right) \tag{3.29}$$

for the SRK equation:

$$\frac{H-H^0}{RT} = Z - 1 - \frac{A}{B}\left[1 - \frac{T}{a(T)}\frac{\partial a(T)}{\partial T}\right]\ln\left(1 + \frac{B}{Z}\right) \tag{3.30}$$

and for the PR equation:

$$\frac{H-H^0}{RT} = Z - 1 - \frac{A}{2^{1.5}B}\left[1 - \frac{T}{a(T)}\frac{\partial a(T)}{\partial T}\right]\ln\left[\frac{Z+(\sqrt{2}+1)B}{Z-(\sqrt{2}-1)B}\right] \tag{3.31}$$

In order to iteratively calculate the final temperature after an isenthalpic expansion, the temperature derivative of the residual enthalpy is required. Therefore, equations 3.29 through 3.31 are differentiated with respect to temperature at constant pressure and composition. After rearranging, the following expressions can be obtained, for the RK equation:

$$\frac{\partial(H-H^0)}{\partial T} = -R + \frac{3}{4}\frac{a}{bT}\ln\left[1+\frac{b}{V}\right] + \left(\frac{\partial V}{\partial T}\right)_P\left[P + \frac{3}{2}\frac{a}{V(V+b)}\right] \tag{3.32}$$

for the SRK equation:

$$\frac{\partial(H-H^0)}{\partial T} = -R + \frac{T}{b}\frac{\partial^2 a(T)}{\partial T^2}\ln\left[1+\frac{b}{V}\right] + \left(\frac{\partial V}{\partial T}\right)_P\left[P + \frac{a(T) - T\dfrac{\partial a(T)}{\partial T}}{V(V+b)}\right] \tag{3.33}$$

and for the PR equation:

$$\begin{aligned}\frac{\partial(H-H^0)}{\partial T} = -R &+ \frac{T}{2^{1.5}b}\frac{\partial^2 a(T)}{\partial T^2}\ln\left[\frac{V+(\sqrt{2}+1)b}{V-(\sqrt{2}-1)b}\right] \\ &+ \left(\frac{\partial V}{\partial T}\right)_p\left[p + \frac{a(T) - T\dfrac{\partial a(T)}{\partial T}}{V(V+b)+b(V-b)}\right]\end{aligned} \tag{3.34}$$

For the first and second temperature derivative of a(T), the equations 3.16 and 3.17 are substituted in equations 3.32 through 3.34. The temperature derivative of the molar volume V at constant pressure cannot be calculated directly, because a cubic equation of state cannot be expressed by V in an explitted form. In appendix B is shown how this derivative was calculated, along with the analytical expressions for all three equations.

3.5 Calculation of fugacity coefficients

In the case of two (or more) phases present, each component will have equal fugacities, f_i,

in both phases:

$$f_i^V = f_i^L \tag{3.35}$$

where V and L refer to the vapour and liquid phases, respectively. The criterion of equal fugacities indicates equality of the driving forces in the transfer of components from one phase to the other. Equation 3.35 can be rewritten in terms of the fugacity coefficient ϕ_i:

$$\phi_i^V y_i = \phi_i^L x_i \tag{3.36}$$

where x_i and y_i are the mole fractions of component i in the liquid and in the vapour phase, respectively. To calculate the fugacity coefficient ϕ_i from an cubic EoS the following general relation is used, given by Reid et al. (1987):

$$RT \ln \phi_i = - \int_{\infty}^{V} \left[\left(\frac{\partial P}{\partial n_i} \right)_{V,T,n_j} - \frac{RT}{V} \right] dV + RT \ln \frac{nRT}{PV} \tag{3.37}$$

For a mixture, with the total overall composition of one mole, the following expressions are obtained for the RK and SRK equation:

$$\ln \phi_i = - \ln (Z - B) + (Z - 1) B_i - \frac{A}{B} [A_i - B_i] \ln \left(1 + \frac{B}{Z} \right) \tag{3.38}$$

and for the PR equation:

$$\ln \phi_i = - \ln (Z - B) + (Z - B) B_i - \frac{A}{2^{1.5} B} (A_i - B_i) \ln \left[\frac{Z + (\sqrt{2} + 1) B}{Z - (\sqrt{2} - 1) B} \right] \tag{3.39}$$

where: $$A_i = \frac{1}{a} \left[2 a_i^{0.5} \sum_j^N x_j a_j^{0.5} (1 - k_{ij}) \right] \quad \text{and} \quad B_i = \frac{b_i}{b}$$

To calculate saturation temperatures for pure component systems (see chapter 6) the temperature derivative of fugacity coefficient at constant pressure is needed. The derivative can be easily calculated from the residual enthalpy (equations 3.29 through 3.31) using the following general relation:

$$\left(\frac{\partial \ln \phi}{\partial T} \right)_{P,x} = - \frac{H^R}{RT^2} \tag{3.42}$$

4. Calculation of the Joule-Thomson Coefficient

Joule-Thomson coefficients are sensitive to small deviations in temperature and pressure. Therefore, they are quite useful to measure the applicability of equations of state to describe a certain system. This chapter deals with the calculation of Joule-Thomson coefficients from equations of state. Analytical expressions are derived and calculated results are compared with experimental data to check for accuracy. Calculations are performed for systems containing nitrogen, hydrocarbons, and mixtures of hydrocarbons.

4.1 Basic relations

As was shown in chapter 2, the Joule-Thomson coefficient μ_{JT} is defined as the pressure derivative of the temperature at constant enthalpy. To calculate the Joule-Thomson coefficient from a cubic EoS, equation 2.3 must be expressed in terms of measurable properties of a fluid: P, V and T. Using the cyclic relation:

$$\left(\frac{\partial T}{\partial P}\right)_H \left(\frac{\partial P}{\partial H}\right)_T \left(\frac{\partial H}{\partial T}\right)_P = -1 \qquad \textbf{(4.1)}$$

and the definition of the heat capacity coefficient C_p (temperature derivative of the enthalpy at constant pressure), equation 2.3 changes as follows:

$$\mu_{JT} = -\frac{1}{C_p}\left(\frac{\partial H}{\partial P}\right)_T \qquad \textbf{(4.2)}$$

To express the enthalpy in terms of P, V and T, the following generalized equation, relating the enthalpy of a homogeneous fluid of constant composition to temperature and pressure, is used (Smith and Van Ness, 1987):

$$dH = C_p dT + \left[V - T\left(\frac{\partial V}{\partial T}\right)_P\right] dP \qquad \textbf{(4.3)}$$

By differentiating equation 4.3 with respect to pressure at constant temperature and then substituting into equation 4.2, the following result is obtained:

$$\mu_{JT} = \frac{1}{C_p}\left[T\left(\frac{\partial V}{\partial T}\right)_P - V\right] \qquad \textbf{(4.4)}$$

The Joule-Thomson coefficient can thus be calculated from the knowledge of the PVT

Table 4.1 Joule-Thomson Coefficient for the RK EoS

$$V1 = v - b \qquad V6 = V3 + \frac{1}{2} V4$$

$$V2 = v + b \qquad V7 = V5 \Big/ \left(- \frac{V3}{V1} + \frac{V4}{V2} + \frac{V4}{v} \right)$$

$$V3 = \frac{R}{V1} \qquad C = \frac{3}{4} \frac{a}{b}$$

$$V4 = \frac{a}{T v V2} \qquad C_p = C_p^0 - V5\, V6 + \frac{C}{T} \ln(V2 / v) - R$$

$$\mu_{JT} = - \frac{V6 + v}{C_p}$$

Table 4.2 Joule-Thomson Coefficient for the SRK EoS

$$V1 = v - b \qquad V6 = V4 - V3 / V2$$

$$V2 = v + b \qquad V7 = T\, V6 \Big/ \left(- \frac{V4\, T}{V1} + \frac{V5}{v\, V2} + \frac{V5}{V2^2} \right)$$

$$V3 = \frac{\partial a(T)}{\partial T} / v \qquad C = \frac{1}{b} \frac{\partial^2 a(T)}{\partial T^2}$$

$$V4 = \frac{R}{V1} \qquad C_p = C_p^0 - V6\, V7 + T\, C \ln(V2 / v) - R$$

$$V5 = \frac{a(T)}{v} \qquad \mu_{JT} = - \frac{V7 + v}{C_p}$$

Table 4.3 Joule-Thomson Coefficient for the PR EoS

$$V1 = v - b \qquad V7 = V4 - V5$$

$$V2 = v + b \qquad V8 = TV7 \Big/ \left(- \frac{V4T}{V1} + 2V6V2 \right)$$

$$V3 = \frac{\partial a(T)}{\partial T} \qquad V9 = \frac{v + (\sqrt{2} + 1) b}{v - (\sqrt{2} - 1) b}$$

$$V4 = \frac{R}{V1} \qquad C = \frac{1}{2^{1.5} b} \frac{\partial^2 a(T)}{\partial T^2}$$

$$V5 = \frac{V3}{(v\, V2 + b\, V1)} \qquad C_p = C_p^0 - V6\, V7 + T\, C \ln(V2 / v) - R$$

$$V6 = \frac{a(T)}{(v V2 + b V2)^2} \qquad \mu_{JT} = - \frac{V7 + v}{C_p}$$

behaviour of a fluid and the specific heat at constant pressure for that state. For the PVT behaviour the three cubic equations of state, described in the previous chapter, are used. These cubic EoS cannot be expressed by V in an explicit form and, therefore, to evaluate μ_{JT} analytically, the rearranged cyclic relation

$$\left(\frac{\partial V}{\partial T}\right)_P = -\left(\frac{\partial P}{\partial T}\right)_V \left(\frac{\partial P}{\partial V}\right)_T^{-1} \tag{4.5}$$

is substituted into equation 4.4. The heat capacity of a real gas is related to the value in the ideal-gas state, at the same temperature and composition (Reid et al., 1987):

$$C_p = C_p^{id} + \Delta C_p \tag{4.6}$$

where the ideal-gas heat capacity C_p^{id} is calculated using equation 3.25. According to Reid et al. (1987), the residual heat capacity ΔC_p for a pressure-explicit EoS, is most conveniently determined by the following expression:

$$\Delta C_p = T \int_{\infty}^{V} \left(\frac{\partial^2 P}{\partial T^2}\right) dV - \frac{T(\partial P/\partial T)_V^2}{(\partial P/\partial V)_T} - R \tag{4.7}$$

Thus, the necessary basic relationships to calculate the Joule-Thomson coefficient using cubic EoS have now been derived.

4.2 Analytical expressions for the Joule-Thomson coefficient

Using the relationships given in the preceeding paragraph the Joule-Thomson coefficients were calculated for all three cubic EoS. Differential and integral operations were performed using the program *Maple V Release 2.0.* To obtain practical formula's for the Joule-Thomson coefficient, the formula's were manipulated in the same way as was done by Hirose and Yoshida (1990). The practical formula's obtained are given in tables 4.1 through 4.3.

4.3 Calculated results and comparison

To check the derived expressions for accuracy, calculated Joule-Thomson coefficients are compared to experimental data. For hydrocarbon systems, Joule-Thomson experiments have been performed by Budenholzer (1939, 1940 & 1942), and Sage and Lacey (1936, 1937 & 1942). Systems investigated were methane, propane, butane, pentane, mixtures of

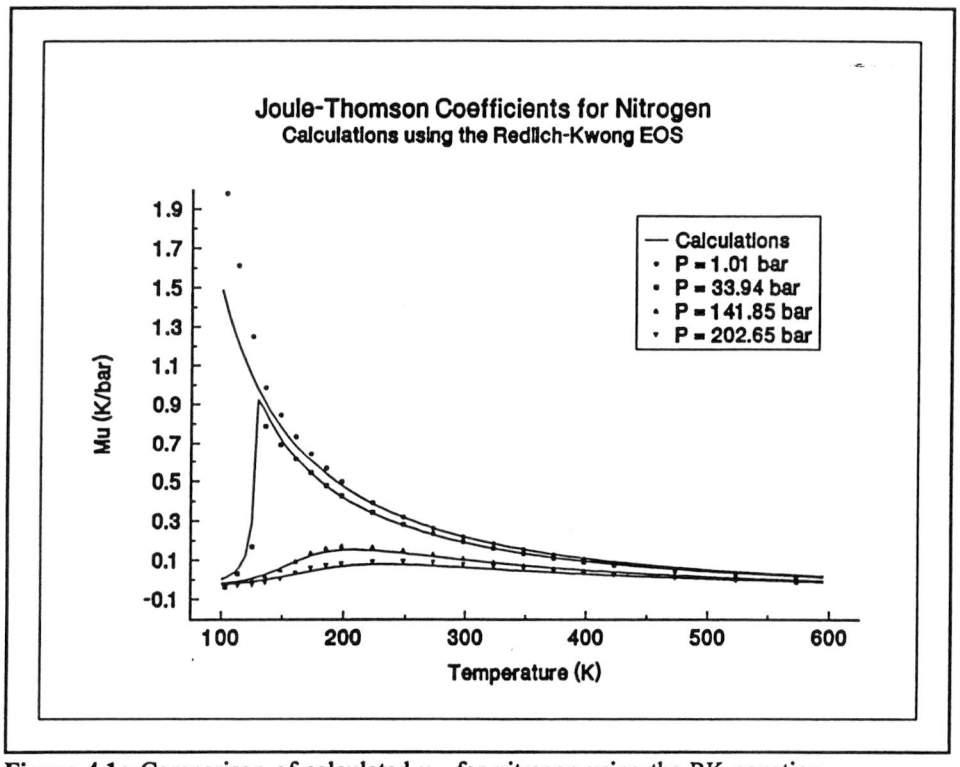


Figure 4.1a Comparison of calculated μ_{JT} for nitrogen using the RK equation

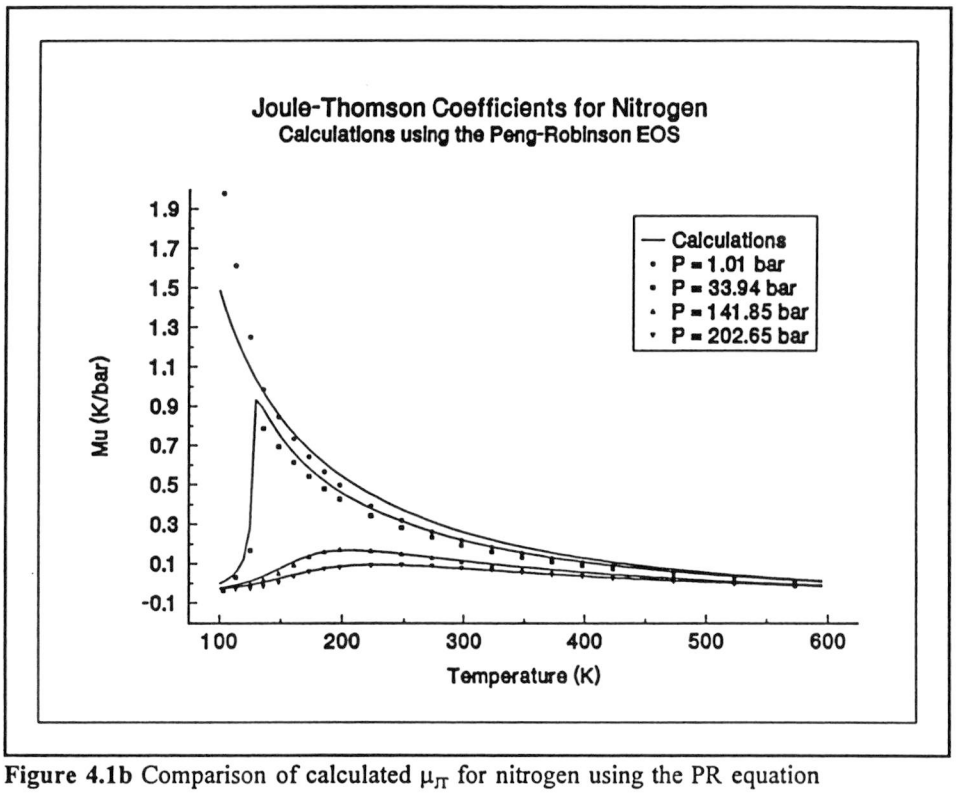


Figure 4.1b Comparison of calculated μ_{JT} for nitrogen using the PR equation

methane-ethane, methane-propane, methane-butane, and two natural gases. The usefulness of the data is somewhat limited because the temperature and pressure range, for which data is available, is rather small. To check the accuracy of the EoS predictions over a wider range, a comparison is first made for nitrogen, because for this component reliable data is available over a wide range of temperatures and pressures.

Hirose and Yoshida (1990) calculated Joule-Thomson coefficients for nitrogen and ethane using several equations of state, including the RK and SRK equation. They found that the RK equation gave the best overall predictions despite its simplicity. The results they obtained for nitrogen using the RK equation are shown in figure 4.1a. The experimental data are taken from the original work of Deming and Deming (1935). The SRK calculations are not shown here, but are quite similar to the RK calculations. The results obtained with the PR equation are given in figure 4.1b. These figures show that for high pressures and high temperatures both equations are quite capable to predict accurately the Joule-Thomson coefficient. At lower pressures and between 150 and 400 K, the RK equations gives the better predictions. In this temperature region the PR equation slightly over-predicts and, therefore, temperature decrease - coefficients are positive - predicted by the PR equation will be too high. In the low temperature and pressure region, predictions by both equations are too low, and both equations will predict temperatures decreases lower than observed.

Calculations were then performed for methane. Budenholzer et al. (1939) reported experimental data for methane, but unfortunately, data is only available for a very small region. Table C1 in appendix C shows calculated values for methane using all three equations and the experimental data for comparison. These results show the same trend as for nitrogen in the temperature region 300-400 K: at low pressures and low temperatures, calculated values are too high, especially for the PR equation. With increasing pressures and temperatures, the RK and the SRK equations give the better results.
To check the results for methane over a wider range, calculations are also compared with data from the IUPAC tables (1978). These data were calculated using an equation with 33 parameters, optimized for methane. Figures 4.2a through 4.2c show the results for the three equations. The results are very similar to those of nitrogen. Large deviations occur in the low pressure and temperature region for all three equations. However, in contradiction to nitrogen, predictions in this region are higher than the actual values and, therefore, predicted temperature drop will be too high. At low pressures and higher temperatures ($T > 250$ K), the PR equations still over-predicts. The RK and SRK equation both show good agreement in this region, especially the RK equation. For pressures higher than 100 bar, all

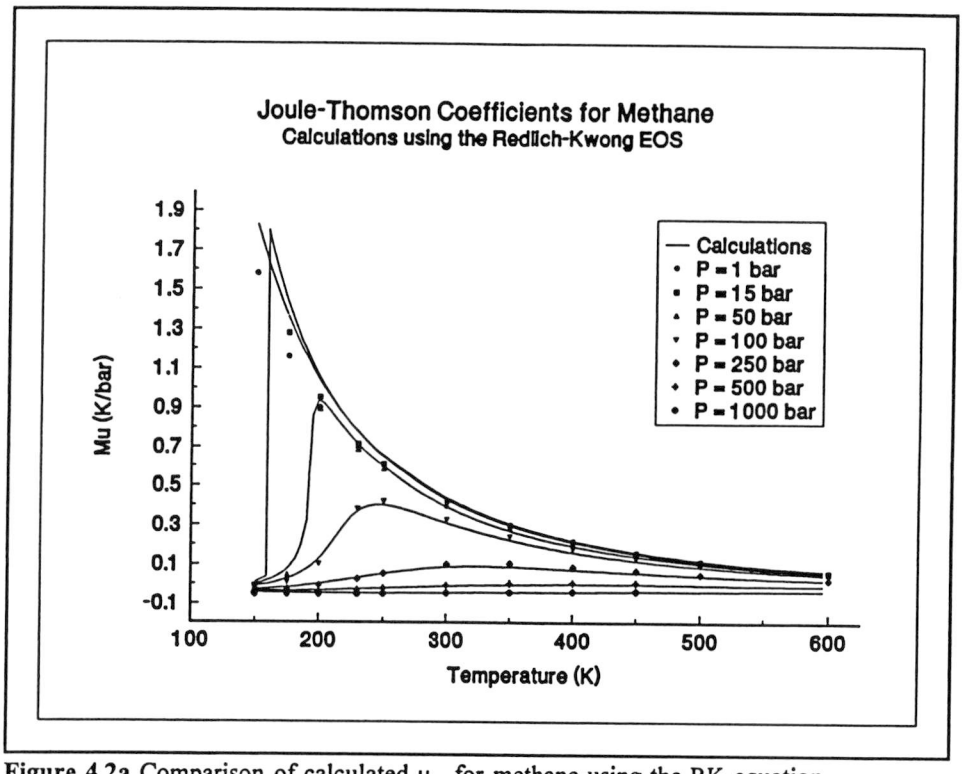


Figure 4.2a Comparison of calculated μ_{JT} for methane using the RK equation

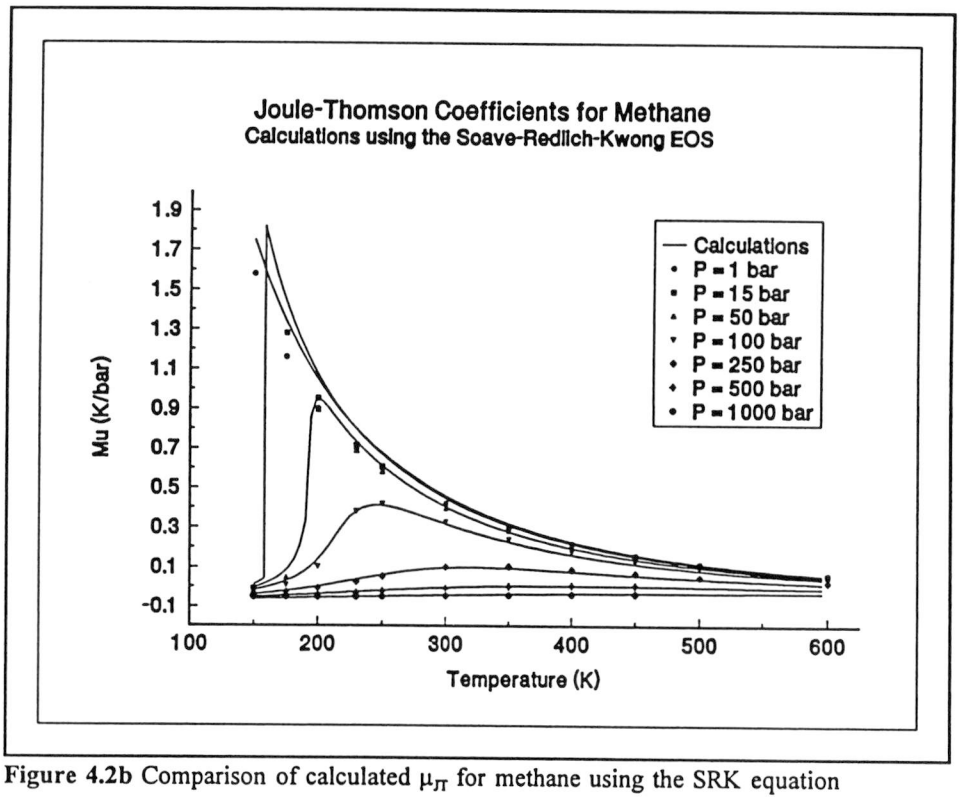


Figure 4.2b Comparison of calculated μ_{JT} for methane using the SRK equation

three equations give accurate predictions.

Both nitrogen and methane are close to spherical, i.e. the acentric factor ω is close to zero. Since the RK equation doesn't use the acentric factor as an input-parameter, it is expected that for larger molecules predictions will get poorer. Therefore, Joule-Thomson coefficients were calculated for mixtures of hydrocarbons containing components with increasing values of ω.
Table 4.4 shows some of the calculated and experimental values for a mixture of methane (78.37 mole%) and n-butane (21.63 mole%). Data are taken from Budenholzer et al. (1940). Table 4.5 shows some of the calculated and experimental values for one of the natural gas, taken from Sage et al. (1942). Results for the other natural gas are given in appendix C (table C2) and are quite similar. Compositions of both natural gases are given in appendix F. For the RK equation, the same values for the binary interaction parameters k_{ij} were used as for the SRK equation.
Rather surprisingly, both tables 4.4 and 4.5 show that the RK equation still gives the best results. All three equations over-predict the value of the Joule-Thomson coefficient, but deviations for the RK equations are the smallest. Although this result is surprising, it is very much in agreement with the results for nitrogen and methane in the same pressure and temperature region. Unfortunately, no data for mixtures is available at higher temperatures and pressures to check for accuracy in that region.

Summarizing, the equations of state have been tested in their ability to accurately calculate Joule-Thomson coefficients over a wide range of pressures and temperatures. Because no experimental data for gas-condensate-like systems is available, calculations have been performed for nitrogen, methane, and mixtures of hydrocarbons. It was found, that despite the fact that the RK equation does not use the acentric factor as an input-parameter, this equation gives the best results over a wide range. Except for the low pressure and temperature region, the RK calculations are quite accurate. The SRK equation gives similar predictions as the RK equation, but in the low pressure and temperature region, deviations are somewhat larger. The PR equation tends to over-predict in low pressure region. Using the PR equation, accurate results are only obtained for high pressures (>100 bar).

In the following chapter the equations will be tested in their ability to predict the inversion curve.

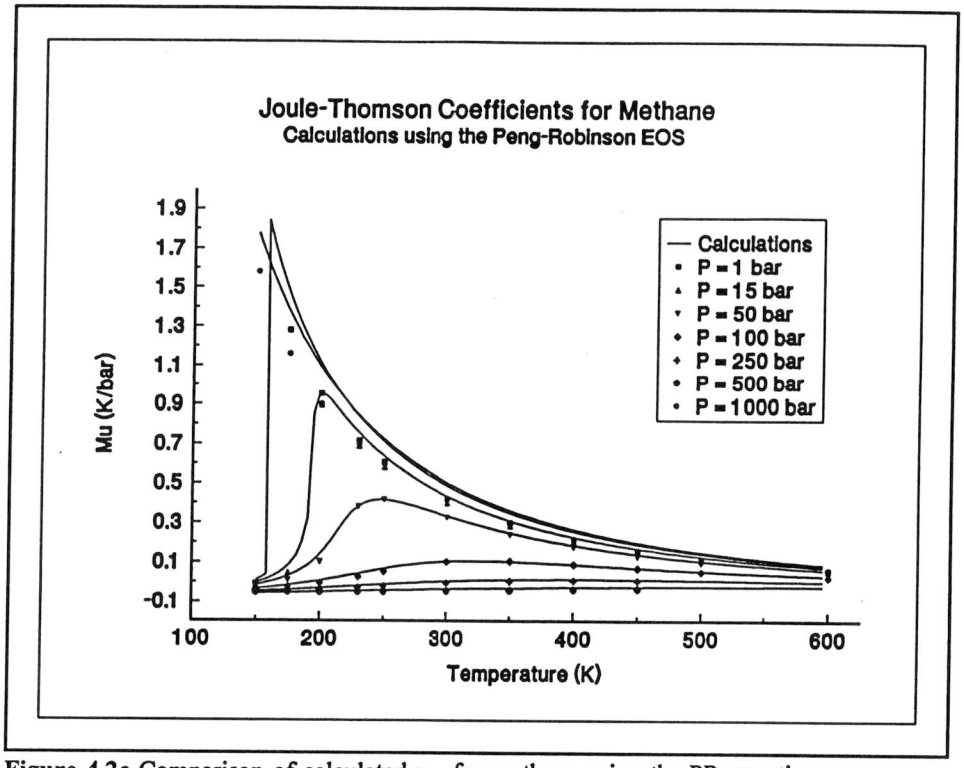


Figure 4.2c Comparison of calculated μ_{JT} for methane using the PR equation

Table 4.4 Calculated and experimental JT coefficients for a methane-butane mixture

T (K)	P (bar)	μ_{JT} (K/bar)			
		experiment	RK	SRK	PR
360.93	17.24	0.4306	0.4482	0.4818	0.5157
	51.71	0.4233	0.4285	0.4535	0.4765
	103.42	0.3564	0.3312	0.3466	0.3543
427.59	17.24	0.2741	0.2910	0.3095	0.3414
	51.71	0.2693	0.2714	0.2842	0.3088
	103.42	0.2274	0.2236	0.2315	0.2460

Table 4.5 Calculated and experimental JT coefficients for natural gas B †

T (K)	P (bar)	μ_{JT} (K/bar)			
		experiment	RK	SRK	PR
294.26	13.79	0.5314	0.5599	0.5910	0.6303
	27.58	0.5314	0.5518	0.5800	0.6149
	41.37	0.5281	0.5382	0.5629	0.5926
360.93	13.79	0.3411	0.3432	0.3616	0.3990
	27.58	0.3378	0.3344	0.3503	0.3846
	41.37	0.3282	0.3240	0.3375	0.3685
427.59	13.79	0.2121	0.2243	0.2321	0.2667
	27.58	0.2209	0.2174	0.2237	0.2559
	41.37	0.2104	0.2100	0.2147	0.2447

† Composition given in appendix F

5. Calculation of the Inversion Curve

It has been recognized for a long time that the prediction of the Joule-Thomson inversion curve is an extremely severe test of an equation of state. Miller (1970) suggested that for newly developed equations of state, it would be desirable to compare the predicted inversion curve with an experimental one.
Several studies have been published dealing with the prediction of the Joule-Thomson inversion curves. Corner (1939) was one of the first to calculate inversion curves using equations of state. After many years, calculations were continued by Miller (1970), Juris and Wenzel (1972), Dilay and Heidemann (1986), Geană and Feroiu (1992) and Najjar et al. (1993). Gunn et al. (1966) derived a general correlation by a curve fit from experimental data. Because only components with acentric factors close to zero were used, the correlation is limited to simple fluids. Heyes and Llaguno (1992) proposed a new molecular dynamics procedure for determining the inversion curve of model simulation fluids.

All these studies have in common that calculations were only performed for pure component systems. The only exception is Najjar et al. (1993), who calculated inversion curves for air, but did not mention anything about mixing rules. To calculate inversion curves for multi-component systems, e.g. gas-condensates, expressions for the inversion curves will have to be derived which include mixing rules. Therefore, the following paragraph will deal with the calculation of inversion curves for both pure and multi-component systems. Calculated results for a number of constituents of oil mixtures will then be compared with experimental data to check for accuracy. Finally the question of why inversion takes place, will be considered.

5.1 Calculation of the inversion curve from equations of state

To calculate the inversion curve from an equation of state, equation 4.5 is substituted in equation 4.4. The following result is then obtained:

$$\mu_{JT} = \frac{1}{C_p}\left(-T\left(\frac{\partial P}{\partial T}\right)_V\left(\frac{\partial P}{\partial V}\right)_T^{-1} - V\right) \qquad (5.1)$$

At inversion conditions the value of μ_{JT} is equal to zero and since C_p is never equal to zero, the locus of the inversion points for any equation of state can be calculated by

solving the following equation:

$$T\left(\frac{\partial P}{\partial T}\right)_V + V\left(\frac{\partial P}{\partial V}\right)_T = 0 \tag{5.2}$$

or by transforming into reduced variables:

$$T_r\left(\frac{\partial P_r}{\partial T_r}\right)_{V_r} + V_r\left(\frac{\partial P_r}{\partial V_r}\right)_{T_r} = 0 \tag{5.3}$$

According to the two-parameter theorem of corresponding states, the compressibility factor is a function of reduced temperature and reduced pressure only. Therefore, a plot of the inversion curve in terms of reduced variables should yield a curve which is valid for all gases. However Pitzer and co-workers (1955) showed that by introducing a third parameter, the acentric factor ω, into the theorem of corresponding states results are significantly improved. That the acentric factor does indeed have a large affect on inversion curve predictions will be shown later.

For pure component systems analytical expressions for the inversion curve were obtained in terms of reduced variables. The necessary differentiations in equation 5.3 were performed and multiplied with T_r and V_r respectively. The expression is then collected in terms of reduced temperature T_r and simplified by factorization. All operations and manipulations were performed using the program *Maple V Release 2.0*. Finally, the following expressions for the inversion curve were obtained for the RK, the SRK, and the PR equations, respectively:

RK EoS

$$0 = T_r\,[-\,\Omega_b x^2 (x+1)^2] + T_r^{-0.5}\,[0.5\;\Omega_a x(5x+3)\;(x-1)^2] \tag{5.4}$$

$$P_r = \frac{T_r}{\Omega_b (x-1)} - \frac{\Omega_a}{T_r^{0.5}\,\Omega_b^2\, x(x+1)} \tag{5.5}$$

SRK EoS

$$0 = T_r[\,\Omega_a m^2 x^2 (x-1)^2 - \Omega_b x^2 (x+1)^2] + T_r^{0.5}[\,\Omega_a m(m+1)x(3x+1)(x-1)^2] + [\,\Omega_a (m+1)^2 x(2x+1)(x-1)^2] \tag{5.6}$$

$$P_r = \frac{T_r}{\Omega_b (x-1)} - \frac{\Omega_a\,\alpha(T)}{\Omega_b^2\, x(x+1)} \tag{5.7}$$

PR EoS

$$0 = T_r[\Omega_a m^2 (x^2+1)(x-1)^2 - \Omega_b (x^2+2x-1)^2] + T_r^{0.5}[\Omega_a m (m+1)(x-1)^2 (3x^2+2x+1)] + [2\Omega_a (m+1)^2 x (x+1)(x-1)^2] \quad (5.8)$$

$$P_r = \frac{T_r}{\Omega_b (x-1)} - \frac{\Omega_a \alpha(T)}{\Omega_b^2 [x(x+1)+(x-1)]} \quad (5.9)$$

The inversion curve can now be calculated by choosing a small value for the dimensionless volume x (x=V/b) and solving the second order equation for T_r. The corresponding value of the reduced inversion pressure P_r is then calculated by filling in the values for x and T_r. By repeatedly increasing x and calculating T_r and P_r, the complete inversion curve is obtained. If it is not desired to have the inversion curve in terms of reduced variables, the reduced variables are simply multiplied by the critical properties.
For multi-component mixtures, the inversion curve is not calculated in terms of reduced variables. For two reasons inversion curve calculations are different for multi-component systems compared with pure component systems. First of all, for a multi-component mixture it is difficult to calculate true critical properties and secondly, for the SRK and the PR equations, the temperature must be extracted from the mixing rule for the a-parameter (equation 3.13) in order to solve equation 5.2 analytically. In Appendix D it is shown how this was done. Equation 3.13 can then be rewritten as follows:

$$a(T) = \Omega_a R^2 [Sum1 - T^{0.5} Sum2 + T Sum3] \quad (5.10)$$

where:

$$Sum1 = \sum_i^N \sum_j^N x_i x_j \frac{T_{ci} T_{cj}}{(P_{ci} P_{cj})^{0.5}} (1+m_i)(1+m_j)(1-k_{ij}) \quad (5.11)$$

$$Sum2 = \sum_i^N \sum_j^N x_i x_j \left(T_{cj} T_{ci}^{0.5} m_i (1+m_j) + T_{ci} T_{cj}^{0.5} m_j (1+m_i)\right) (P_{ci} P_{cj})^{-0.5} (1-k_{ij}) \quad (5.12)$$

$$Sum3 = \sum_i^N \sum_j^N x_i x_j m_i m_j \left(\frac{T_{ci} T_{cj}}{P_{ci} P_{cj}}\right)^{0.5} (1-k_{ij}) \quad (5.13)$$

The same kind of formula manipulation, as was done for pure components, was performed to obtain the inversion curve for multi-component systems. For the SRK and PR equation the following expressions are obtained, which are quite similar to the equations for pure

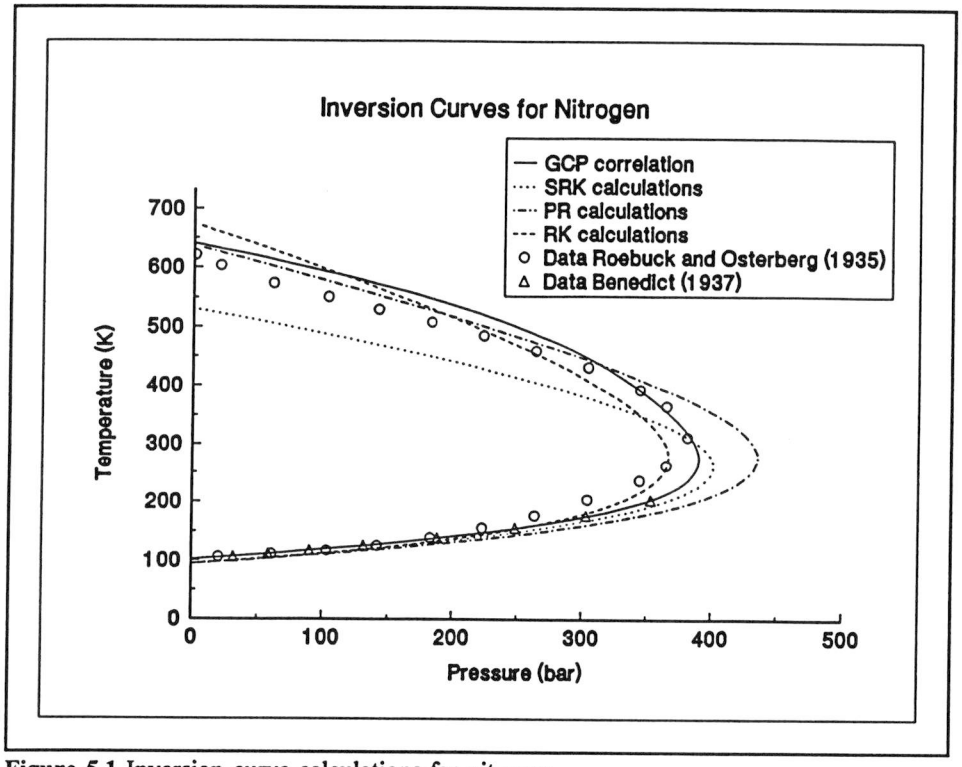


Figure 5.1 Inversion curve calculations for nitrogen

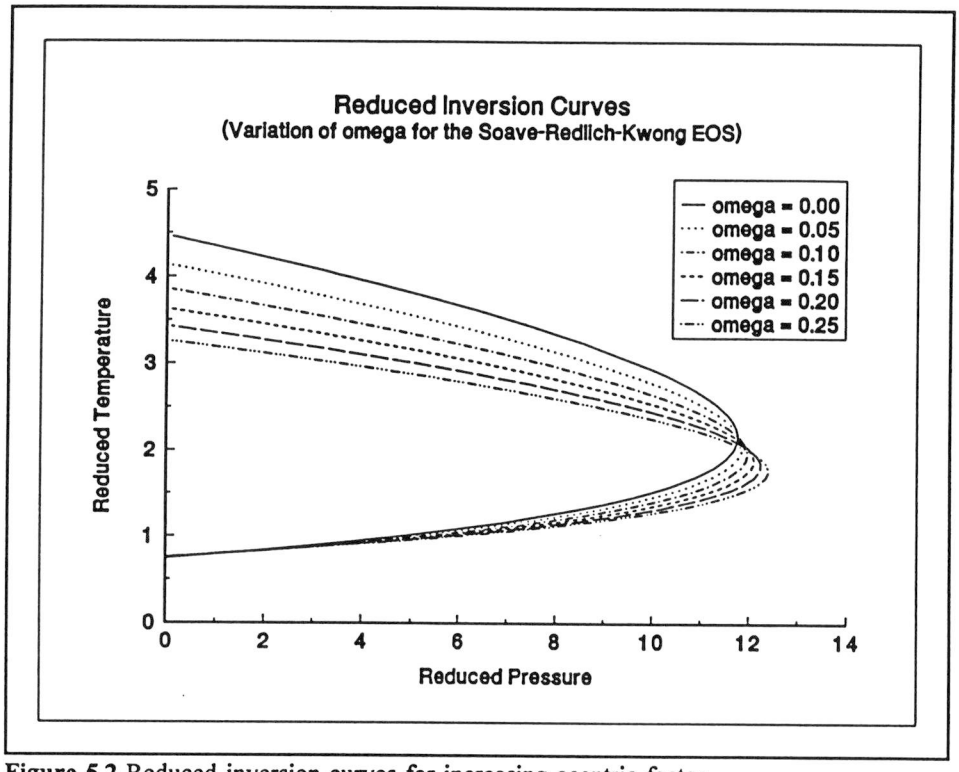


Figure 5.2 Reduced inversion curves for increasing acentric factor

components:

SRK EoS

$$\begin{aligned} 0 &= T[\,\Omega_a R^2 Sum3\, x^2 (x-1)^2 - R b x^2 (x+1)^2\,] \\ &+ T^{0.5}[\,-0.5\,\Omega_a R^2 Sum2\, x (3x+1)(x-1)^2\,] \\ &+ [\,\Omega_a R^2 Sum1\, x (2x+1)(x-1)^2\,] \end{aligned} \qquad (5.14)$$

PR EoS

$$\begin{aligned} 0 &= T[\,\Omega_a R^2 Sum3\,(x^2+1)(x-1)^2 - R b (x^2+2x-1)^2\,] \\ &+ T^{0.5}[\,-0.5\,\Omega_a R^2 Sum2\,(x-1)^2 (3x^2+2x+1)\,] \\ &+ [\,2\Omega_a R^2 Sum1\, x (x+1)(x-1)^2\,] \end{aligned} \qquad (5.15)$$

Inversion curve calculations are performed in the same way as for pure components, i.e. a value for x is repeatedly chosen, the second order equation for T_r is solved, and the inversion pressure is simply calculated using the appropriate PVT relationship, as given in chapter 3 (After converting x back to V).

5.2 Calculated results and comparison

When the calculated inversion curves are compared with the experimental data, two problems are encountered. First of all, inversion data is only available for a few well-studied materials, including He, H_2, N_2, CH_4, CO, C_2H_4, CO_2, and NH_3. For hydrocarbons, inversion data is given for ethane and propane by Din (1956), but for higher carbon-numbers no data has been found. For mixtures of hydrocarbons, the situation is even worse. The only data found, is given by Gunn et al. (1966). They include some experimental inversion points for two methane-butane mixtures. Secondly, the data that is available, is mostly available at low reduced temperatures, i.e. for the low-temperature part of the inversion curve. Therefore, comparison of calculated inversion curves and experimental data over a wide range of pressures and temperatures will be difficult.
It is to be noted that for convenience all calculated inversion curves have been extrapolated to the minimum inversion temperature. As mentioned in chapter 2, this is physically incorrect because the inversion curve stops when it cuts the vapour-pressure line.

Nitrogen is one of the few components for which inversion data is available over a wide range of temperatures and pressures and, therefore, a comparison between calculated and experimental curves is made for this component.

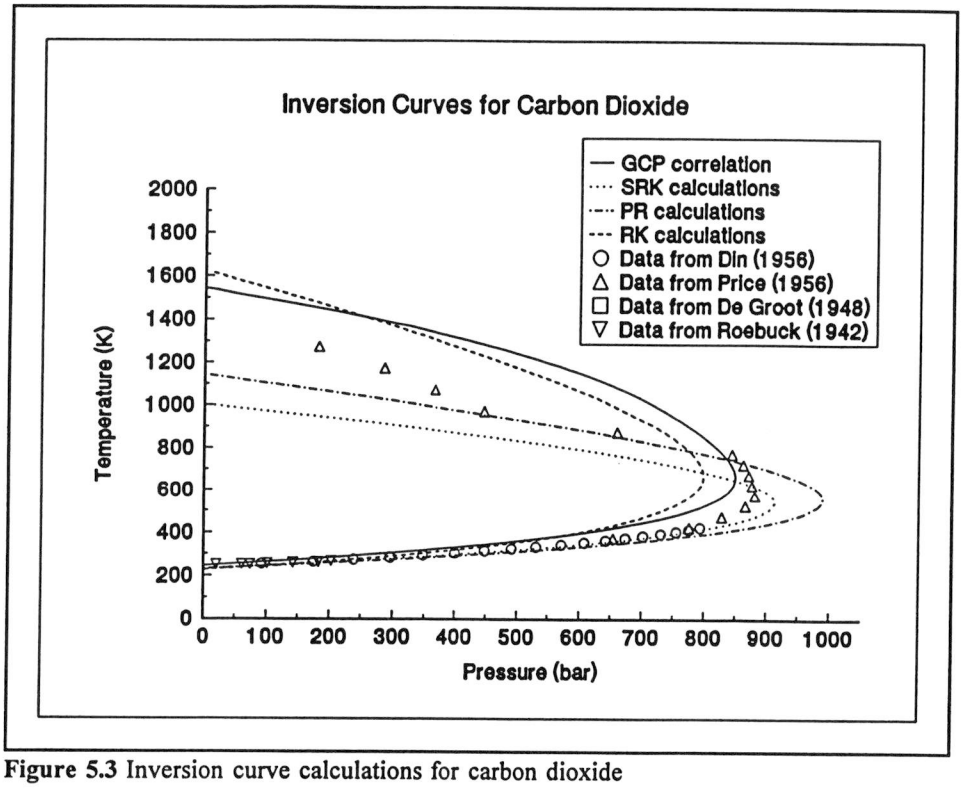


Figure 5.3 Inversion curve calculations for carbon dioxide

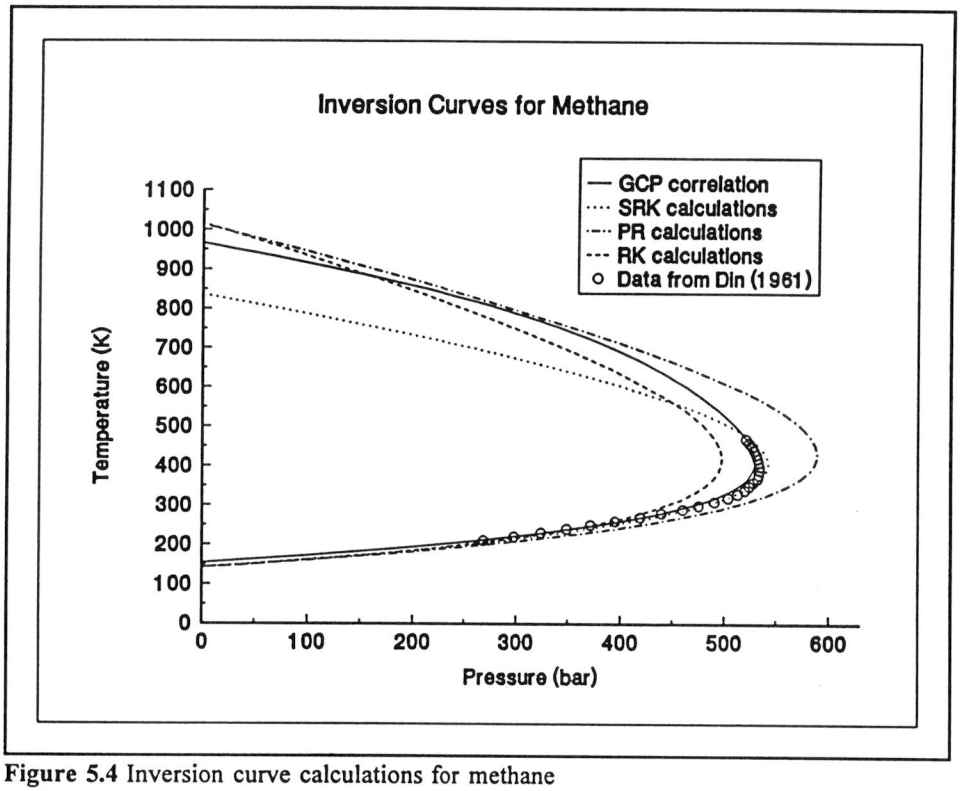


Figure 5.4 Inversion curve calculations for methane

Figure 5.1 gives the calculated inversion curves for the three cubic EoS. The Gunn-Chueh-Prausnitz (GCP) correlation (1966) for simple fluids is also included for comparison. Experimental data is taken from Roebruck and Osterberg (1935), and from Benedict (1937). Figure 5.1 shows that all three equations are capable of adequately describing the lower part of the curve. The upper part of the curve and the maximum inversion temperature are best predicted by the PR equation. The SRK equation gives a close match for the peak of the curve and the maximum inversion pressure. The RK gives pressures that are to low in this region and the PR equation predicts a maximum inversion pressure that is too high.

Another component with a wide range of available data is carbon dioxide. This example is chosen to show the importance of including the acentric factor in inversion curve calculations. Figure 5.2 shows the influence of the acentric factor ω on the location of the inversion curve for the SRK equation. It is seen that increasing values of ω have only a small effect on the lower part of the curve, but a rather large effect on the upper part. The maximum reduced inversion temperature decreases with increasing ω and it is also seen that the peak of the curve is effected by ω. The maximum reduced inversion pressures increases for increasing ω, and the temperature for which this pressure is found decreases. The PR equation shows exactly the same trend as the SRK equation. Because the acentric factor is not a input-parameter for the RK equation, calculated inversion curves for this equation will show decreasing accuracy for increasing ω. This shown by figure 5.3 which gives the inversion curves for carbon dioxide, which has a value of 0.239 for ω. All equations give good predictions for the lower branch, with increasing deviations for the RK equation and the GCP correlation with increasing temperature. These two equations also seam to over-predict the location of the upper part. The data of Price (1956) appear somewhat scattered, especially for the upper part and, therefore, exact comparison is rather difficult in this area. Once again the SRK equation most closely matches the peak of the curve and again the PR equation predicts a maximum inversion pressure that is too high.

Figure 5.4 shows calculations for methane. Methane has a small acentric factor that is comparable with that of nitrogen and results for methane follow the same trend as for nitrogen. To examine the effect of mixtures, inversion curves were calculated for a mixture of methane (79.97 mole%) and butane (20.03 mole%). This mixture is slightly different than the one used in the Joule-Thomson coefficient calculations, but still very comparable. Calculated curves, using a binary interaction parameter k_{ij} of zero, are shown in figure 5.5. The region for which data is available is rather small, but results show the same trend as for carbon dioxide.

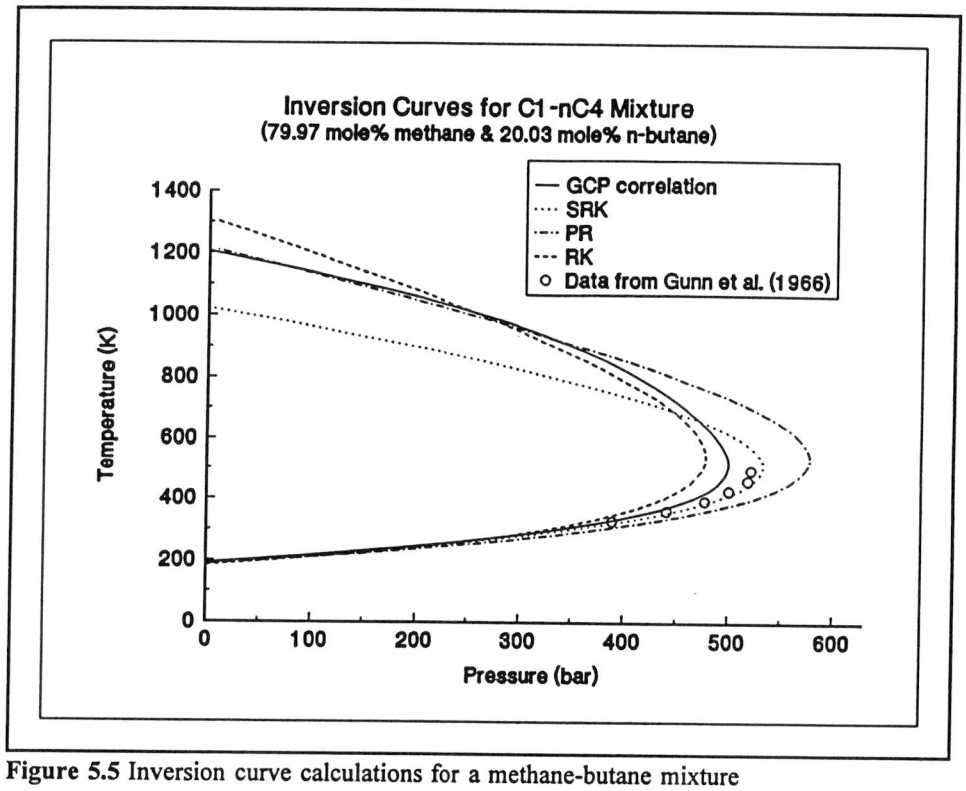


Figure 5.5 Inversion curve calculations for a methane-butane mixture

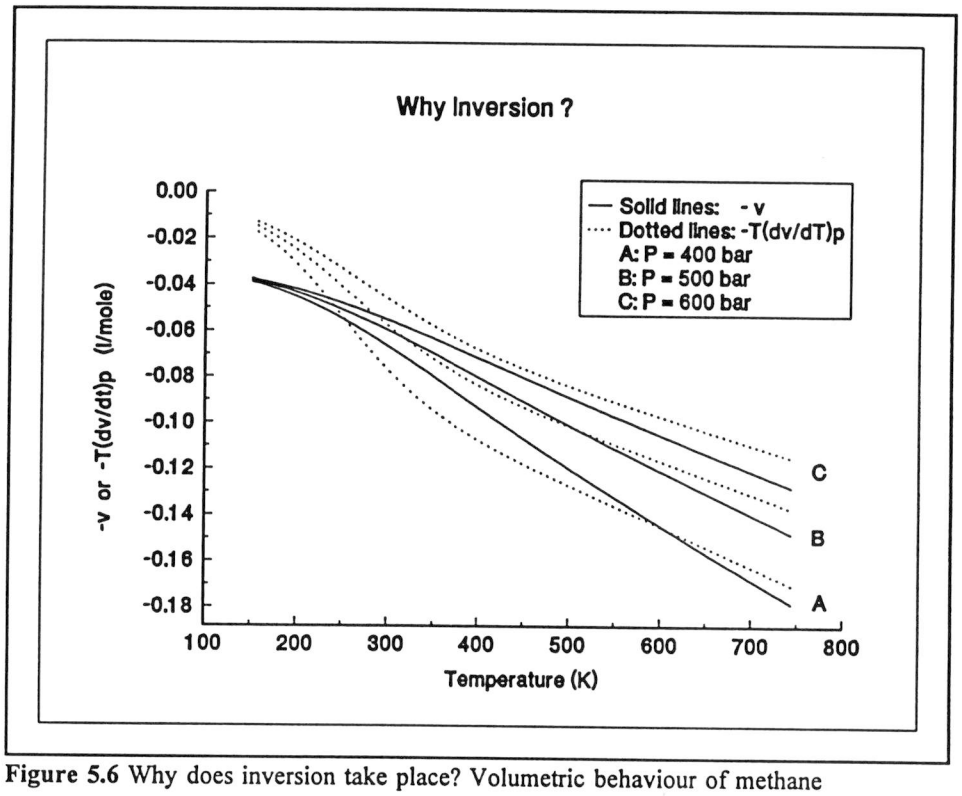


Figure 5.6 Why does inversion take place? Volumetric behaviour of methane

Although no experimental data is available for complex hydrocarbon mixtures that resemble gas-condensate mixtures (see chapter 7), the shown examples illustrate the ability of the equations of state to predict inversion curves. They show that with increasing accentric factor, predictions by the RK equation become poorer. Therefore, the RK equation is probable not capable of accurately prediction inversion curves for gas-condensate systems. The SRK and PR equation both give good predictions for the low-temperature part of the curve. The SRK matches the peak of the curve fairly well, but predictions for the upper part are too low. The PR equation over-predicts the maximum inversion pressures, but gives a better description of the upper curve.

5.3 Why inversion?

Now that it has been shown that Joule-Thomson inversion can take place and that the inversion curve can be calculated, an interesting question arises: why does inversion takes place? Or in other words, what causes the Joule-Thomson coefficient to change sign.

To answer this question, we must turn back to equation 2.2, which states that no heat is transferred and that no work is performed, and that the Joule-Thomson expansion is therefore isenthalpic. Although no heat is transferred externally, heat transfer does take place internally, as will be shown. Therefore, we will start with the definition of enthalpy:

$$H = U + PV \tag{5.16}$$

The enthalpy changes by an amount

$$dH = dU + PdV + VdP \tag{5.17}$$

By substituting the following expression for the change in internal energy

$$dU = TdS - PdV \tag{5.18}$$

into equation 5.17; the following result is obtained for the change in enthalpy:

$$dH = TdS + VdP \tag{5.19}$$

which shows that changes in enthalpy involve changes in entropy and compression energy. Since for Joule-Thomson expansion changes in enthalpy are zero, the work energy lost during pressure drop will appear as an increase in entropy:

$$-\int VdP = \int TdS \tag{5.20}$$

Therefore, during expansion, compression energy is converted into heat energy of which entropy is the extensive factor. It can be easily shown that the change in entropy is related to the change in volumetric behaviour of a fluid (Zemansky and Dittman, 1981):

$$TdS = C_p dT - T\left(\frac{\partial V}{\partial T}\right)_P dP \tag{5.21}$$

Substituting equation 5.21 into 5.19 will give the relation for the change in enthalpy that was given in chapter 4. (equation 4.3).
With the use of equation 5.21 it can now be made clear when temperatures will increase or decrease at expansion. If the right-hand side of equation 5.21 - the isothermal entropy change - equals the change in compression energy -VdP, the change in temperature will be equal to zero. This situation is satisfied in the inversion points. However, if the compression energy exceeds this value, the term $C_P dT$ must then be greater than zero and because C_P is always positive, dT must be greater than zero and, therefore, temperature will increase. Reversely, the temperature will be decrease, if the change in compression energy does not exceed the right-hand side. Therefore, the occurrence of inversion is related to the change in volumetric behaviour of fluids.

Figure 5.6 elucidates the change in volumetric behaviour. The figure shows the value of -V and $-T(dV/dT)_P$ as a function of temperature for different pressures. Calculations were performed for methane using the SRK equation, but it is merely an example to makes things clear. The figure shows some interesting features: at 600 bar, which is higher than the maximum inversion pressure for methane (calculated by the SRK equation), the value of -V is always greater than $-T(dV/dT)_P$ and, therefore, temperatures will increase at expansion. At 400 and 500 bar two temperatures exist for which both curves intersect one another. These temperatures correspond to the inversion temperature. Another interesting aspect is the difference in excess for low and high temperatures. At low temperatures the excess of -V over $-T(dV/dT)_P$ is greater than at high temperature and, therefore, if values of C_P don not differ much, temperature increase will be higher at low temperatures.

6. Isenthalpic Expansion Calculations

Now that it has been made clear that during isenthalpic expansion temperature changes can either be positive or negative, depending on the initial and final state, it is time to calculate these temperature changes. Therefore, a computer scheme is developed that calculates the enthalpy H_0 in the initial state (P_0,T_0) and after pressure has dropped to P_1, iteratively calculates the temperature T_1 until the enthalpy in the final state H_1 equals H_0. Enthalpies are calculated using the relations given in chapter 3. For pure and multi-component systems the problem of calculating isenthalpic expansion is basically the same, that is the final temperature must be iteratively corrected to keep the enthalpy constant. But because the algorithms are essentially different, both systems will be dealt with separately. This chapter describes these computer algorithms. Calculations for pure component systems are performed for methane and a comparison is made with tabulated data. Calculations for multi-component systems will be dealt with in the next chapter.

6.1 Isenthalpic expansion of pure component systems

6.1.1 Computation scheme for pure component systems

To calculate the isenthalpic expansion of pure components a computer algorithm was developed which is schematically shown in figure 6.1.
The enthalpy H_0 is calculated in the initial state T_0 and P_0 and after the pressure has dropped to its final value P_1, possible calculation routes are two-fold:

- The pressure is less than the critical pressure P_c. The saturation temperature T_{sat} is then calculated by an iterative correction of the temperature to equalize the fugacity coefficient of the liquid and the vapour phase. Iterations are performed using a one-dimensional Newton-Raphson procedure. Enthalpies of the saturated liquid and vapour are then calculated. If the initial enthalpy H_0 lies between the two saturation enthalpies, the system is in the two-phase region. The temperature is then equal to T_{sat} and the vapour fraction can easily be calculated.

- The pressure exceeds the critical pressure, or the enthalpy lies outside the saturated region. The system is then in the single-phase region (either vapour or liquid) and the temperature is iteratively corrected until the initial and final enthalpy are equal.

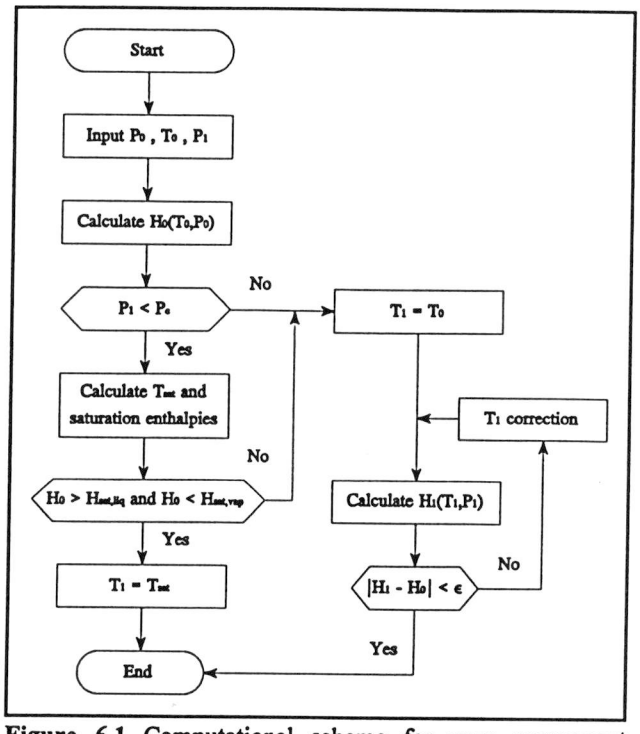


Figure 6.1 Computational scheme for pure component isenthalpic expansion

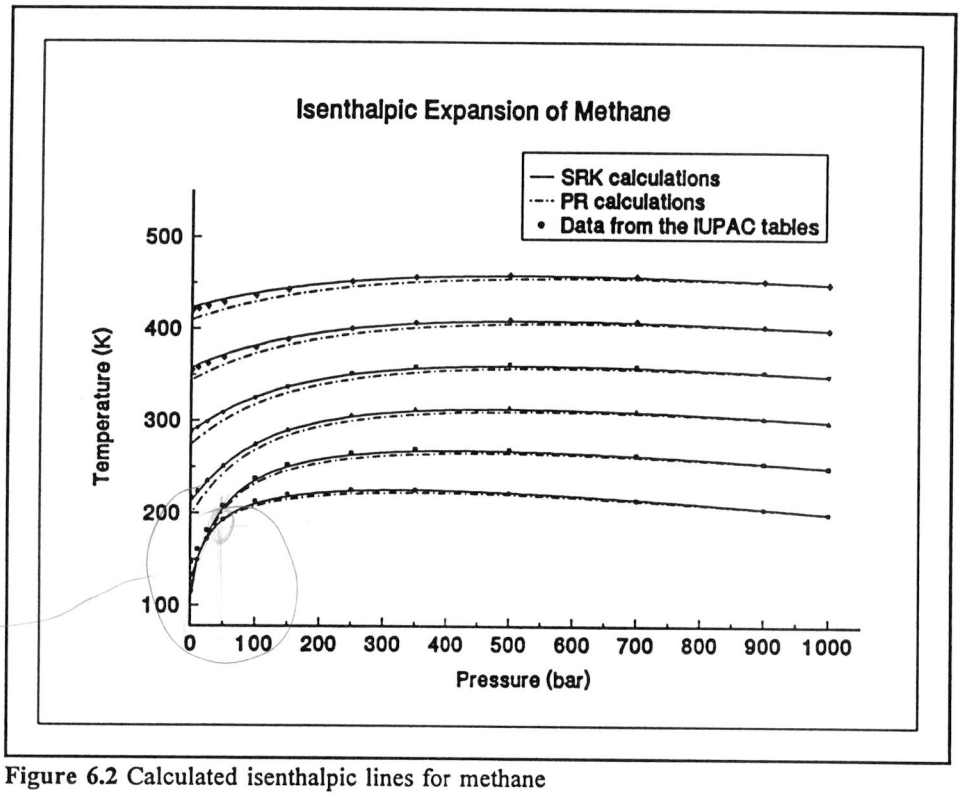


Figure 6.2 Calculated isenthalpic lines for methane

The problem is also solved using a one-dimensional Newton-Raphson iteration procedure.

To calculate an isenthalpic line the pressure is reduced step-wise to the final pressure. For each step the final temperature is calculated using the pressure and temperature of preceding step as initial values.

6.1.2 Calculated results and comparison

Isenthalpic expansion calculations for pure components have been performed for methane. Both the SRK and PR equation were used to calculate isenthalpic lines. Several starting points were used; the initial pressure was 1000 bar for all cases, but initial temperatures varied from 200 to 450 K in steps of 50 K. The final pressure was 1 bar for all cases. Results are shown in figure 6.2 for both equations. For comparison data was obtained through linear intrapolation of the IUPAC tables for methane (1978) and these are also shown in figure 6.2. Figure 6.2 shows that especially the SRK equation is capable of accurately describing the isenthalpic expansion of methane. At high temperatures and low pressures predictions are slightly higher than the data and only at low starting temperatures (e.g. 250 K), significant deviations occur in the low pressure region. Although the saturation temperatures are accurately calculated, calculated saturation enthalpies en enthalpies near the two phase region deviate somewhat from the data, and therefore predictions in this area are less accurate. The PR predictions show larger deviations than the SRK predictions, especially at lower pressures in the temperature region 250-400 K. This is in agreement with the results in §4.3, where it was shown that in this region Joule-Thomson coefficients calculated by PR are too high.

6.2 Isenthalpic expansion of multi-component systems

6.2.1 Computation scheme for multi-component systems

For multi-component systems staying in the single phase, the calculations are performed in the same way as for pure component systems: i.e. an iterative correction of the temperature to solve the enthalpy balance. However, when staying in the two-phase system the problem is different. The main difference between computation of the Joule-Thompson expansion in the single and the two phase region consists in:

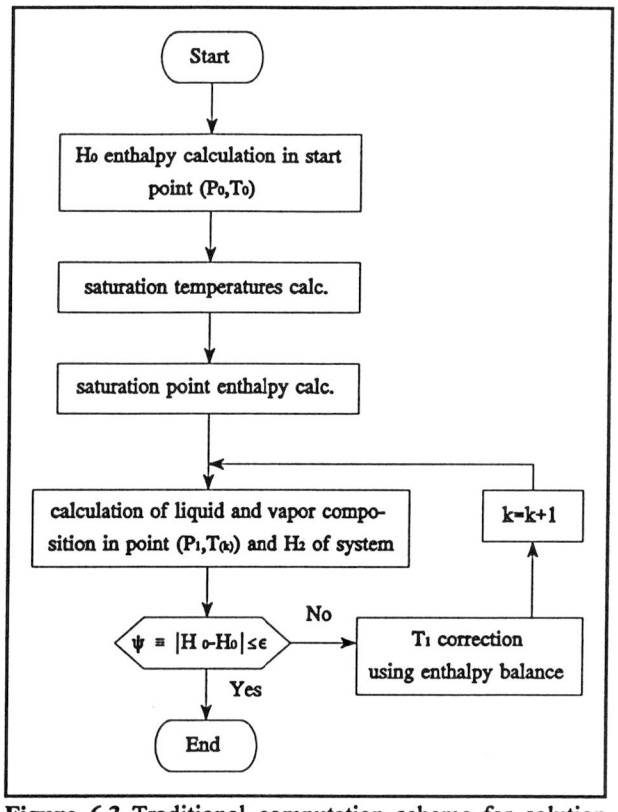


Figure 6.3 Traditional computation scheme for solution of isenthalpic problem in two-phase region

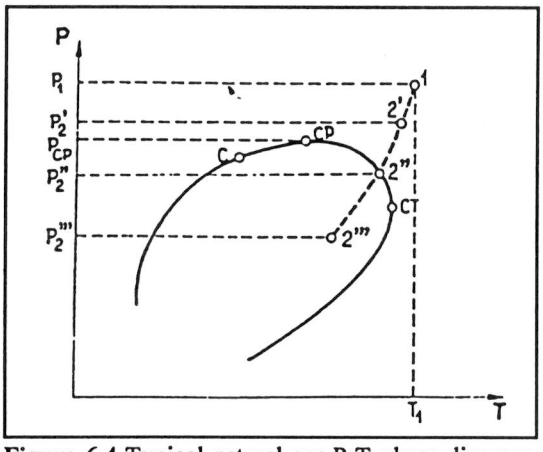


Figure 6.4 Typical natural gas P-T phase diagram

- calculation of liquid and vapour compositions during throttling
- necessity of computation of enthalpy for both phases
- necessity of computation of the phase fractions in the system

Figure 6.3 shows a standard algorithm for solving a two phase isenthalpic problem, taken from Nagy (1991). The key point of this scheme is the calculation of the system enthalpy in the saturation conditions, i.e. bubble and dew-point temperatures, at final pressure P_1. This algorithm however has no possibility to verify the existence of two phase region at P_1. In other words, it may not even be possible to calculate bubble and dewpoint temperatures, because they do not exist. This is shown in figure 6.4 taken from Nagy (1991), which shows the phase envelope for typical natural gas and a throttling line. Note that the initial pressure is P_1 and the final pressure is P_2', P_2'' and P_2''', respectively. The figure shows that if final pressure is equal to:

- P_2' : it is not possible to calculate saturation temperatures, because the system is in the single-phase region
- P_2'' : it is possible to get a single and a two-phase solution
- P_2''': the system is in the two-phase region

This example shows the importance of knowing when the two-phase region is entered. The key point in the two-phase Joule-Thompson effect is, therefore, the verification of the two-phase region. To verify the presence of two (or more) phases, the Gibbs tangent plane criterion to check phase stability is used, as described by Michelsen (1982). If two phases are present a isenthalpic flash is performed to calculate the final temperatures as well as the phase amounts and phase compositions.
Figure 6.5 presents the used computational scheme for a multi-component system in the single or two-phase region. After checking stability in the initial phase, the problem is either a single or a two-phase solution.

- **Single phase solution;** the initial enthalpy is calculated and the pressure is dropped to its final value. the Joule-Thomson coefficient is calculated in the initial state and from the coefficient and the pressure drop (P_0-P_1), an initial value of T_1, called T_1^0, is estimated. The phase stability is checked in this hypothetical point (P_1,T_1^0) and if a single phase is present, T_1 is simply calculated by solving the enthalpy balance as was done for pure component systems. The system is checked for stability in the final state and if this is true, the problem is solved. If the single phase is not stable, a two-phase solution is necessary.

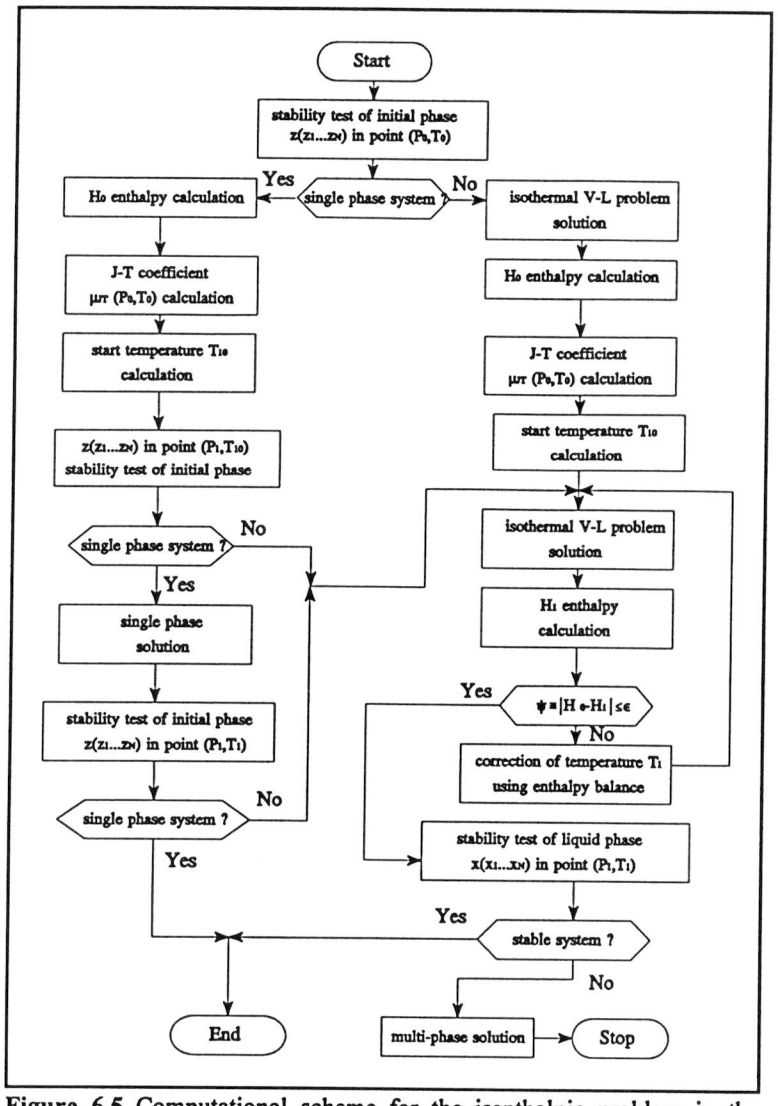


Figure 6.5 Computational scheme for the isenthalpic problem in the single and two-phase region

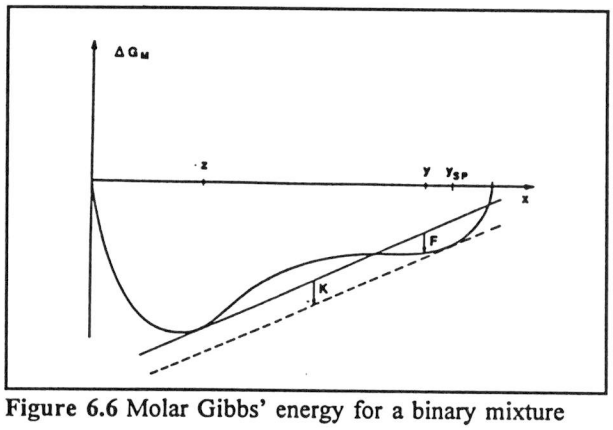


Figure 6.6 Molar Gibbs' energy for a binary mixture

- **Two-phase solution;** the initial enthalpy is calculated as a sum of the liquid and the vapour phase contributions. The Joule-Thomson coefficient is calculated for the vapour phase to get a first estimation of T_1. The problem is then solved using a isenthalpic flash procedure, with the resulting phase compositions of the stability test as initial estimates. The system is checked for stability in the final state (i.e. if two phases are still present), and if this is true, the problem is also solved. If more than two phases exist the problem remains unsolved and the program is terminated.

Calculation of isenthalpic lines is done in the same way as was done for pure component systems, i.e. reducing pressure step-wise and repeatedly calculating the final temperature for each step.
The following two paragraphs will describe the Gibbs tangent plane theory and the isenthalpic flash procedure.

6.2.2 The Gibbs tangent plane theory

To calculate whether a given mixture at specified P and T splits into two or more phases is called a stability analysis. This paragraph will give a summary of the work done by Michelsen (1982). For further details reference is made to the original article.

Consider a M-component system at given temperature and pressure (T_0, P_0) with mole fractions z_1, z_2. ..., z_N. The Gibbs energy of the mixture, if only one phase present, is:

$$G_0 = \sum_i^M n_i \mu_i^0 \qquad (6.1)$$

where n_i is the number of moles of component i in the mixture, and μ_i^0 the chemical potential. Assume that this mixture is divided into two phases (I and II) with mole numbers N-ε and ε, respectively, and that the amount ε of the second phase II is infinitesimal. The change in the Gibbs free energy due to the phase split is then:

$$\Delta G = G_I + G_{II} - G_0 = G(N - \epsilon) + G(\epsilon) - G_0 \qquad (6.2)$$

The Gibbs energy of phase I may be approximated by a Taylor series expansion, truncated after the first order term:

$$G_I = G(N - \epsilon) = G_0 - \sum_{i=1}^{M} \epsilon_i \left(\frac{\partial G}{\partial n_i} \right)_n \qquad (6.3)$$

When substituting this equation into equation 6.2 the following result is obtained:

$$\Delta G = G_0 - \sum_{i=1}^{M} \epsilon_i \left(\frac{\partial G}{\partial n_i} \right)_n + G(\epsilon) - G_0 \qquad (6.4)$$

or by introducing the chemical potential μ_i^0:

$$\Delta G = G(\epsilon) - \epsilon \sum_{i}^{M} y_i \mu_i^0 = \epsilon \sum_{i}^{M} y_i (\mu_i(y) - \mu_i^0) \qquad (6.5)$$

Stability of the original mixture requires that its Gibbs energy is at the global minimum. Hence a necessary criterion for stability is that

$$F(y) = \sum_{i}^{M} y_i (\mu_i(y) - \mu_i^0) \geq 0 \qquad (6.6)$$

for all trial compositions y of phase II. Geometrically F(y) is the vertical distance from the tangent hyperplane to the molar Gibbs energy surface at composition z to the energy surface at composition y, as shown schematically in figure 6.6 for a binary mixture. For stability at point z, the tangent hyper plane must be below the Gibbs energy surface for each value of y. The figure includes a parallel tangent at the stationary point y_{sp} and the vertical distance K between both tangents, as will be explained.

For cubic equations of state it is more convenient tot work in terms of fugacity coefficients than in terms of chemical potentials. By expressing the chemical potential in terms of the fugacity coefficient:

$$\mu_i = \mu_i^0 + RT(\ln z_i + \ln \phi_i + \ln P) \qquad (6.7)$$

equation 6.5 can be rewritten as:

$$\frac{\Delta G}{\epsilon RT} = \sum_{i=1}^{M} y_i (\ln y_i + \ln (\phi_i)_{II} - \ln z_i - \ln (\phi_i)_0) \qquad (6.8)$$

The stability criterion F(y) can now be expressed in terms of mole fractions and fugacity coefficients. Only one phase exists if:

$$F(y) = \sum_{i=1}^{M} y_i (\ln y_i + \ln (\phi_i)_{II} - \ln z_i - \ln (\phi_i)_0) \geq 0 \qquad (6.9)$$

for all trial compositions y of phase II. Since all minima of F(y) are located in the interior

of the permissible region ($y_i \geq 0$, $\Sigma y_i = 1$), F(y) will be non-negative if it is non-negative at all stationary points, that is points where the derivatives with respect to all independent variables equal zero. A stationary point must satisfy the following equation:

$$\ln y_i + \ln(\phi_i)_{II} - \ln z_i - \ln(\phi_i)_0 = k \qquad \textbf{(6.10)}$$

where k is the dimensional chemical potential difference, and is independent of the component index. By introducing new variables, Y_i, given by:

$$\ln Y_i = \ln y_i - k \qquad \textbf{(6.11)}$$

the stationary criterion can be rewritten as:

$$\ln Y_i + \ln(\phi_i)_{II} - \ln z_i - \ln(\phi_i)_0 = 0 \qquad \textbf{(6.12)}$$

The new independent variables Y_i can formally be interpreted as mole numbers, the corresponding mole fractions being $y_i = Y_i / \Sigma_j Y_j$. Stationary points are located as solutions to equation 6.12, and stability is provided if at all stationary points $k \geq 0$, corresponding to $\Sigma_i Y_i \leq 0$. Conversely, a phase is unstable if a stationary point can be located were $k < 0$, corresponding to $\Sigma_i Y_i > 0$, and a split of the original phase in two phases with mole fractions y_i in the second phase, will decrease the total Gibbs energy if the amount of the second phase is chosen small enough.

6.2.3 The isenthalpic flash problem

After establishing that the system is unstable in the hypothetical point P_1, T_1^0, isenthalpic flash calculations are performed. For two phases in equilibrium, the following set of equations must be satisfied:

Material Balance Equations:
$$\beta y_i + (1 - \beta) x_i = z_i \qquad \{i = 1,\ldots,N\} \qquad \textbf{(6.13)}$$

Equilibrium equations:
$$y_i \phi_i^V = x_i \phi_i^L \qquad \{i = 1,\ldots,N\} \qquad \textbf{(6.14)}$$

Summation of Mole Fractions:
$$\sum_{i=1}^{N} (y_i - x_i) = 0 \qquad \textbf{(6.15)}$$

Enthalpy Balance Equation: $H - H_{spec} = 0$ **(6.16)**

Equations 6.13 through 6.16 contain (2N+2) equations and (2N+4) variables, namely (x_1, x_2, ..., x_N), (y_1, y_2, ...,y_N), vapour fraction β, T, P and H. With H and P specified for an isenthalpic flash, the number of variable equals the number of equations and the system can be solved for x_i, y_i, β and T.

To solve the isenthalpic flash problem, the general flash procedure given by Prausnitz et al. (1980) was used. In this procedure the equations 6.13 through 6.16 are solved using a simplified procedure. The 2N+2 equations are not solved simultaneously, but two object functions are defined, namely the Rachford-Rice equation:

$$g_1 = \sum_{i=1}^{N} \frac{z_i (K_i - 1)}{\beta (K_i - 1) + 1} \qquad \textbf{(6.17)}$$

and the enthalpy balance equation:

$$g_2 = \beta H^V + (1 - \beta) H^L - H_{spec} \qquad \textbf{(6.18)}$$

where K_i is the equilibrium constant ($K_i=y_i/x_i$), H^V and H^L the molar vapour and liquid enthalpy and H_{spec} the specified (i.e. initial) enthalpy. At convergence both g_1 and g_2 are equal zero.

Solutions are found using a two-dimensional Newton-Raphson iteration procedure to determine the size of the next β and temperature step (Δβ and ΔT). New estimates for β and T are calculated, then new compositions of the vapour and the liquid phase are calculated and iteration is proceeded until convergence is reached. For further details, reference is made to Prausnitz et al. (1980).

The procedure of Prausnitz calculates derivatives with respect temperature using a finite difference approximation (calculation of function at T+ΔT). To increase computer speed and ensure accuracy, these numerically obtained derivatives should be replaced by analytical expressions for the two temperature derivatives, which are given by Michelsen (1986). An outline is given in appendix E. Replacements have not been performed yet, but it is recommended to do this in the near future.

7. Joule-Thomson Effects in Gas-Condensate Reservoirs

Before starting Joule-Thomson calculations for gas-condensate systems, it is important to get an insight into these systems first. Therefore, a short introduction into the matter of gas-condensates will be given. This includes typical compositions and composition ranges, reservoir temperatures and pressures, and phase behaviour. Then attention will be given to the characterization and lumping procedures of the heavy fractions, which are needed to perform calculations for gas-condensates with cubic equations of state. When this is done, isenthalpic expansion calculations will be performed using the algorithms as described in chapter 6. Because no experimental data are available at this time, a sensitivity study is included to examine the effect of factors such as fluid composition, reservoir pressure and temperatures, pressure drop, the choice of an equation of state, and the influence of characterization and lumping procedures on calculation results.

7.1 Gas-condensate reservoirs

Gas-Condensate systems, like all petroleum reservoir fluids, are multi-component mixtures consisting primarily of hydrocarbons, which are separated into three classes:

- Paraffins (or alkanes)
- Naphthenes (or cycloalkanes)
- Aromatics

In addition to the hydrocarbons, also nitrogen, carbon dioxide, and hydrogen sulphide are often found in petroleum mixtures. Finally, petroleum mixtures may contain helium, mercury, and metal-organic compounds. Petroleum reservoir fluids may be classified as gases, gas-condensates, volatile oils, or black oils. As Bradley (1989) stated: "The single most striking factor about gas-condensate systems is that they exist either wholly or preponderantly as vapour phase in the reservoir at the time of discovery (the critical temperature of the system is lower than the reservoir temperature)". This is shown in figure 7.1, taken from Pedersen et al. (1989), which shows the phase envelopes for each class along with typical reservoir conditions for petroleum fluids. Typical compositions of these petroleum reservoir fluids are shown in table 7.1. Compared with natural gas the methane content of gas-condensates is lower and the fraction heavier ends (C_2^+) is larger.

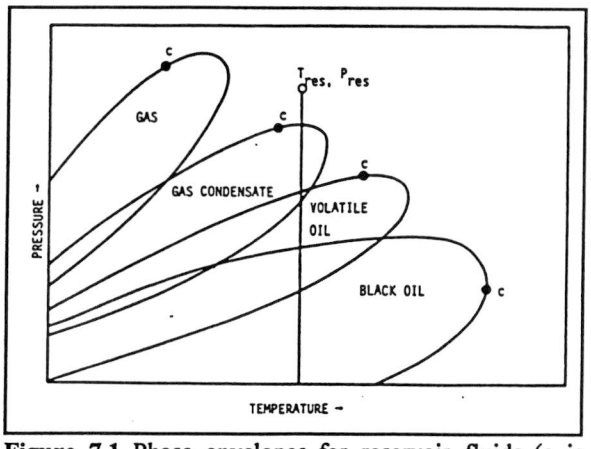


Figure 7.1 Phase envelopes for reservoir fluids (c is the critical point)

Table 7.1 Typical molar compositions of petroleum reservoir fluids†

Component	Gas	Gas-Condensate	Volatile Oil	Black Oil
N_2	0.3	0.71	1.67	0.67
CO_2	1.1	8.65	2.18	2.11
C_1	90.0	70.86	60.51	34.93
C_2	4.9	8.53	7.52	7.00
C_3	1.9	4.95	4.74	7.82
C_4 (i+n)	1.1	2.00	4.12	5.48
C_5 (i+n)	0.4	0.81	2.97	3.80
C_6 (i+n)	6^+: 0.3	0.46	1.99	3.04
C_7		0.61	2.45	4.39
C_8		0.71	2.41	4.71
C_9		0.39	1.69	3.21
C_{10}		0.28	1.42	1.79
C_{11}		0.20	1.02	1.72
C_{12}		0.15	12^+: 5.31	1.74
C_{13}		0.11		1.74
C_{14}		0.10		1.35
C_{15}		0.07		1.34
C_{16}		0.05		1.06
C_{17}		17^+: 0.37		1.02
C_{18}				1.00
C_{19}				0.9
C_{20}				20^+: 9.18

† Data taken from Pedersen (1989)

Compared with crude oils, gas-condensates contain much smaller amounts of heavy (C_7^+) fractions. The compositions of gas-condensate system can vary widely and supposedly have a large influence on the Joule-Thomson effect. In appendix F compositions of more gas-condensates, taken from Pedersen et al. (1988 & 1989) and Montel (1994), are listed to illustrate how wide these variations in composition can be. These gas-condensates systems will be used for isenthalpic calculations later on.

Reservoir conditions of gas-condensate systems may also vary considerably. Bradley (1989) states that gas-condensate reservoirs may occur at pressures below 2000 psia (138 bar) and temperatures below 100 °F (38 °C), and probably can occur at any higher fluid pressure and temperatures within reach of drill. Most known gas-condensate reservoirs are in the range of 3000 to 8000 psia (207 to 552 bar) and 200 to 400 °F (93 to 204 °C). In the introduction, reservoir pressures of approximately 1000 bar were mentioned, which lie well outside this pressure range. This confirms that drilling into high-pressure gas-condensate reservoirs is not yet very common. However, high-pressure high-temperature reservoirs are not an unknown phenomenon and work has been published regarding this matter. Baker and Price (1990) reported typical reservoir conditions in the range of 16000 psia (1100 bar) and 400 °F (200 °C). However, due to confidentiality requirements, actual composition of this light gas-condensate could not be published. The only fluid property reported was a molecular weight of 31.0 g/mole, which, as will be shown later, is still useful. Montel (1994) has also reported high-pressure high-temperature reservoirs. Typical conditions were 1100 bar and 185-195 °C. Compositions for these systems are given in appendix F.

When describing the phase behaviour of multi-component mixtures, such as gas-condensates systems, phase envelopes as shown in figure 7.1 are used. The figure gives the region for which two-phases are formed, separated by the bubble and dewcurve and both curves ending in the critical point. In multi-component systems, the critical temperature is no longer the maximum temperature for which two phases can exist. Instead the phase envelope is bounded by the cricondentherm, the maximum temperature for formation of two phases, and the cricondenbar, the maximum pressure. As mentioned earlier, typical reservoir conditions for gas-condensate reservoirs lie outside the phase envelope and hence the fluid is a single phase.

Describing the phase behaviour of a gas-condensate is generally very difficult, because phase behaviour of a gas-condensate is very much dependent on the heaviest constituents of the mixture. For example, a dew point pressure is highly sensitive to the molecular

weights of the heaviest components. It is also expected that a cubic equation of state is less accurate when dealing with larger molecules. While the lighter constituents of reservoir fluids are well-defined individual components, the heavier (C_7^+) fractions however are mixtures of different components. In order to simulate the phase behaviour of a gas-condensate, it is necessary that the C_7^+ fractions are accurately characterized. Therefore, several characterization procedures have been developed over the years, which are described in the following paragraph.

7.2 Characterization of the heavy fraction

As described in the previous paragraph, petroleum reservoir fluids are mixtures of primarily hydrocarbons. These mixtures are often too complex for a complete identification of the components. The components of a reservoir fluid may instead be classified as follows:

- **Defined components** are those for which there exist a quantitative gas chromatographic analysis and for which T_c, P_c and ω are known. This incudes non-hydrocarbons and hydrocarbons up to at least C_6.

- **TBP fractions** cover the true boiling point (TBP) fractions from C_7 and up to maximum C_{30}, each containing many different components. The density and the average molecular weight may be measured on each fraction.

- The **TBP residue** consists of components that are too heavy to be separated using a TBP distillation procedure. The density and average molecular weight may be measured for the total sample, but neither boiling point nor molecular weight distributions are determined.

Because pure component properties are not available, calculation of mixtures containing heavy hydrocarbon fractions require characterization procedures to determine the cubic EoS parameters T_c, P_c and ω. These parameters can be estimated from measured physical properties, i.e. the boiling point T_B, the molecular weight MW, and the density ρ, using empirical correlations. The density is frequently replaced by the specific gravity SG, i.e the density of the fluid divided by the density of water at the same conditions. In this work three different procedures are used to characterize the heavy fractions.

After analyzing several characterization methods, Pedersen et al. (1982) concluded that

for the North Sea oils, the best results were obtained using the Cavett (1964) relations for T_c an P_c:

$$T_c = 768.071 + 1.7134\,T_B - 0.10834\,10^{2}\,T_B^2 + 0.3889\,10^{6}\,T_B^3 - 0.89213\,10^{-2}\,T_B API + 0.53095\,10^{-6}\,T_B^2 API + 0.32712\,10^{-7}\,T_B^2 API^2 \quad (7.1)$$

$$\log P_c = 2.829 + 0.9412\,10^{-3}\,T_B - 0.30475\,10^{-5}\,T_B^2 + 0.15141\,10^{-8}\,T_B^3 - 0.20876\,10^{-4}\,T_B API + 0.11048\,10^{-7}\,T_B^2 API + 0.1395\,10^{-9}\,T_B^2 API^2 - 0.4827\,10^{-7}\,T_B API^2 \quad (7.2)$$

and the Lee-Kesler relations (1976) for the acentric factor ω:

$$\omega = \frac{\ln P_{Br} - 5.92714 + 6.09648\,T_{Br}^{-1} + 1.28862 \ln T_{Br} - 0.169347\,T_{Br}^{6}}{15.2518 - 15.6875\,T_{Br} - 134721 \ln T_{Br} + 0.43577\,T_{Tr}^{6}} \quad (7.3)$$

(for $T_{Br} < 0.8$*)*

$$\omega = -7.904 + 0.1352K - 0.007465K^2 + 8.359\,T_{Br} + (1.408 - 0.001063K)\,T_{Br}^{-1} \quad (7.4)$$

(for $T_{Br} > 0.8$*)*

In these equations, API = 141.5/SG - 131.5, where SG is the 60 °F/60 °F specific gravity. The critical temperature T_c and the boiling point of the fraction T_B are given in °F, and the critical pressure P_c in psia. T_{Br} is the reduced boiling point (T_B/T_c) and P_{Br} the reduced pressure (P_B/P_c). K is the Watson characterization factor and is related to the molecular weight of the fraction and the specific gravity:

$$K = 4.5579\,MW^{0.15178}\,SG^{-0.84573} \quad (7.5)$$

Because measured boiling points are seldom available above C_{20}, they have to be estimated for the heavier fractions. To use the Cavett and Lee-Kesler relations, T_B can be estimated using the relation from Katz and Firoozabadi (1978):

$$T_B(K) = 97.58\,MW^{0.3323}\,SG^{0.04609} \quad (7.6)$$

T_B is thus estimated from the available physical data, MW and ρ. This is in fact an unnecessary intermediate step and therefore Pedersen and Thomassen (1988) suggested to use correlations, which directly correlate T_c, P_c and ω with MW and ρ using the following expressions:

$$T_c = c_1 \rho + c_2 \ln MW + c_3 MW + c_4 MW^{-1} \quad (7.7)$$

$$\ln P_c = d_1 + d_2 \rho + d_3 MW^{-1} + d_4 MW^{-2} \quad (7.8)$$

$$m = e_1 + e_2 MW + e_3 \rho + e_4 MW^2 \quad (7.9)$$

where m is equal to the relations 3.10 for SRK or 3.11 for PR, from which ω can be calculated. Table 7.2 lists optimum values for the parameters c_1-c_4, d_1-d_4, e_1-e_4 for use with the SRK equation.

Table 7.2 Optimum values of the coefficients in equations 7.7-7.10

	Pedersen (1988)	Aasberg-Petersen (1991)
c_1	$1.6312\ 10^{2}$	190.8 ± 37
c_2	$8.6052\ 10^{0}$	87.70 ± 3.1
c_3	$4.3475\ 10^{-1}$	0.6387 ± 0.087
c_4	$-1.8774\ 10^{3}$	-5247.7 ± 2830
d_1	$-1.3408\ 10^{-1}$	-0.5836 ± 0.097
d_2	2.5019	2.3211 ± 0.23
d_3	$2.0846\ 10^{2}$	282.2 ± 24
d_4	$-3.9872\ 10^{3}$	-4484.9 ± 2100
e_1	$7.4310\ 10^{-1}$	$2.829\ 10^{-2} \pm 6.4\ 10^{-2}$
e_2	$4.8122\ 10^{-3}$	$4.248\ 10^{-4} \pm 4.7\ 10^{-4}$
e_3	$9.6707\ 10^{-3}$	$1.156\ 10^{-2} \pm 3.0\ 10^{-3}$
e_4	$-3.7184\ 10^{-6}$	$-4.069\ 10^{-6} \pm 7.5\ 10^{-7}$

Aasberg-Petersen and Stenby (1991) used equations to calculated T_c and P_c, which have the same functional form as those from Pedersen et al. (however, ρ is replaced by SG). The value of the parameters are slightly different (table 7.2). The value of omega is calculated directly from:

$$\omega = e_1 + e_2 MW + e_3 SG + e_4 MW^2 \quad (7.10)$$

To calculate the ideal gas heat capacity of oil-mixtures the third-order polynomial is used as described in chapter 3. To calculate the ideal-gas heat capacity coefficients $C_{p,1}$-$C_{p,4}$ the following procedure suggested by Kesler and Lee (1976) is used (the calculated heat

capacity is in Btu/lb):

$$C_{p,1} = -0.33886 + 0.02827\,K - 0.26105\,CF + 0.59332\,\omega\,CF$$

$$C_{p,2} = -(0.9291 - 1.1543\,K + 0.0368\,K^2)\,10^{-4} + CF\,(4.56 - 9.48\,\omega)\,10^{-4}$$

$$C_{p,3} = -1.6658\,10^{-7} + CF\,(0.536 - 0.6828\,\omega)\,10^{-7}$$

$$C_{p,4} = 0$$

where CF = ((12.8 - K)(10 - K)/(10ω))², and K the Watson characterization factor as defined by equation 7.5.

7.3 Reduction of the number of components

Gas-condensates mixtures may consist of a many components or fractions of pseudo-components. This amount can easily exceed values of thirty or more, and it is obvious that for extensive calculation purposes this number must be reduced. In this work two reasons exist for reducing this number:

1) reducing the number of components saves computer time
2) the Michelsen flash routines that are used in this work to perform the stability test, are limited to a maximum of 15 components.

Therefore, a lumping procedure is used to group components and hence reducing the amount of fractions. Pedersen et al. (1985) have developed a procedure whereby all hydrocarbon segments of the C_{7+}-fractions are given approximately equal importance. This is accomplished via a grouping of carbon number fractions on weight basis. N_2, CO_2, and CH_4 (C_1) are given separate status, ethane and propane are grouped together (C_2-C_3), and so are C_4's, C_5's, and C_6's. The C_{7+}-fraction is divided into three (or more) groups that, by weight, are of equal size. The group critical properties are computed as weight-mean values:

$$T_{ci} = \sum_j x_j MW_j T_{cj} \,/\, \sum_j x_j MW_i \qquad (7.12)$$

$$P_{ci} = \sum_j x_j MW_j P_{cj} \,/\, \sum_j x_j MW_i \qquad (7.13)$$

$$\omega_i = \sum_j x_j MW_j \omega_j \,/\, \sum_j x_j MW_i \qquad (7.14)$$

Where index j runs from the lowest tot he highest carbon number within a group i. For the C_{7+} carbon number fractions, P_{cj}, T_{cj}, and ω_j are calculated using one of characterization procedures as described in paragraph §7.2. The average values of $C_{p,1}$-$C_{p,4}$ for the ideal gas heat capacity are also calculated as a weight mean average.
As an example the gas-condensate mixture GC7 (full composition given in appendix F) was lumped, and the results are shown in table 7.3. All three procedures described in the previous paragraph were used to characterize the heavy fractions of the mixture.

Table 7.3 Lumped Gas-Condensate mixture GC7 using different characterization methods

Components	mole%	weight%	Pedersen			Aasberg-Petersen			Cavett & Kesler/Lee †		
			T_c (K)	P_c (bar)	ω	T_c (K)	P_c (bar)	ω	T_c (K)	P_c (bar)	ω
N_2	0.64	0.63	126.2	33.90	0.0390	126.2	33.90	0.0390	126.2	33.90	0.0390
CO_2	9.16	14.17	304.1	73.80	0.2390	304.1	73.80	0.2390	304.1	73.80	0.2390
C_1	68.80	38.81	190.4	46.00	0.0110	190.4	46.00	0.0110	190.4	46.00	0.0110
C_2-C_3	13.54	16.93	335.7	45.84	0.1244	335.7	45.84	0.1244	335.7	45.84	0.1244
C_4-C_6	3.94	9.19	450.8	34.87	0.2285	450.8	34.87	0.2285	450.8	34.87	0.2285
C_7-C_9	2.30	8.58	556.9	29.89	0.4999	576.1	31.89	0.4449	565.0	32.02	0.4080
C_{10}-C_{14}	0.94	5.19	629.5	22.15	0.6474	672.1	19.15	0.6120	653.3	25.43	0.6560
C_{15}-C_{45}	0.68	6.57	745.3	15.40	0.9518	825.1	10.98	0.9108	766.2	15.00	1.0850

† Values taken from Pedersen et al. (1989)

7.4 Isenthalpic expansion and inversion curves

Isenthalpic calculations were performed for several gas-condensate systems. As an example gas-condensate N° 1 (GC1) from appendix F was chosen for these first calculations. The SRK equation of state was used for this example and the heavy fractions were characterized using the Aasberg-Petersen procedure. Figure 7.2 shows several isenthalpic lines with different starting temperatures (initial pressure was 1000 bar for all cases). The calculated inversion curve is also included for this system together with a part of the phase envelope, which was calculated using the program Thermopak™. Unfortunately, the algorithm this program uses is not very robust and calculation of both the bubble and the dewcurve are terminated before the critical point is reached. Other problems are

encountered using the Michelsen flash procedures. For low temperatures starting points, calculations stop at the phase boundary because the Michelsen flash fails to converge. For higher temperatures the same problems are encountered, not near to the phase boundary, but somewhere in the middle of the two phase region. To avoid this, the computer program has the possibility to switch off the stability test as soon as the two-phase region has been reached. However, one must be very careful doing this because figure 7.2 shows that if temperatures are between approximately 450 and 550 K, the systems can return to the single-phase region at low pressures (retrograde behaviour). If the stability test is switched off this will not be noticed.

Now a very strange thing happens. To examine the influence of lumping on this system, components were grouped together as explained in paragraph §7.3. Calculations were then repeated using the same initial conditions. Very surprisingly no problems were encountered at all: the stability test could be used at any point and calculations continued with out problems in the two-phase region even at low temperatures, as is shown in figure 7.3. Because for both systems the heavy fraction C_{10+} - the fraction is treated as a separate group - is the same, this contradictory behaviour can not be explained by small concentrations of the heavy fraction in the gas-phase as was suggested by Loggers (1991). Fortunately, lumping has little effect on the calculated isenthalpic lines (and only a slight effect on the inversion curve) and therefore, if these problems are encountered for other gas-condensate systems, they can be avoided by grouping the components.

Returning to figures 7.2 and 7.3, we see that it is indeed possible to calculate temperature increase at expansion of high pressure high temperature gas-condensate reservoirs, one of the main purposes of this work. Although reservoir temperatures are in the region where temperatures will drop, reservoir pressures may well be above the maximum inversion pressure. The figures show that for reservoir pressures higher than approximately 550 bar, temperatures will increase at expansion until the pressure reaches the inversion curve. This shows that for reservoir fluids not the reservoir temperature, but the reservoir pressure is of interest when examining the Joule-Thomson inversion effect.
However, temperature increases are not that dramatic as they may have voiced in the introduction. In the temperature region of interest (300-500 K), maximum temperature increases are 9 K when starting at 500 K, and 21 K for 300 K. If however temperature decrease is expected and extra heat is added to avoid for instance condensation or hydrate formation, even these small increases may cause surprises. In the following paragraph we will examine what effect gas-condensate composition has on these temperature increases.

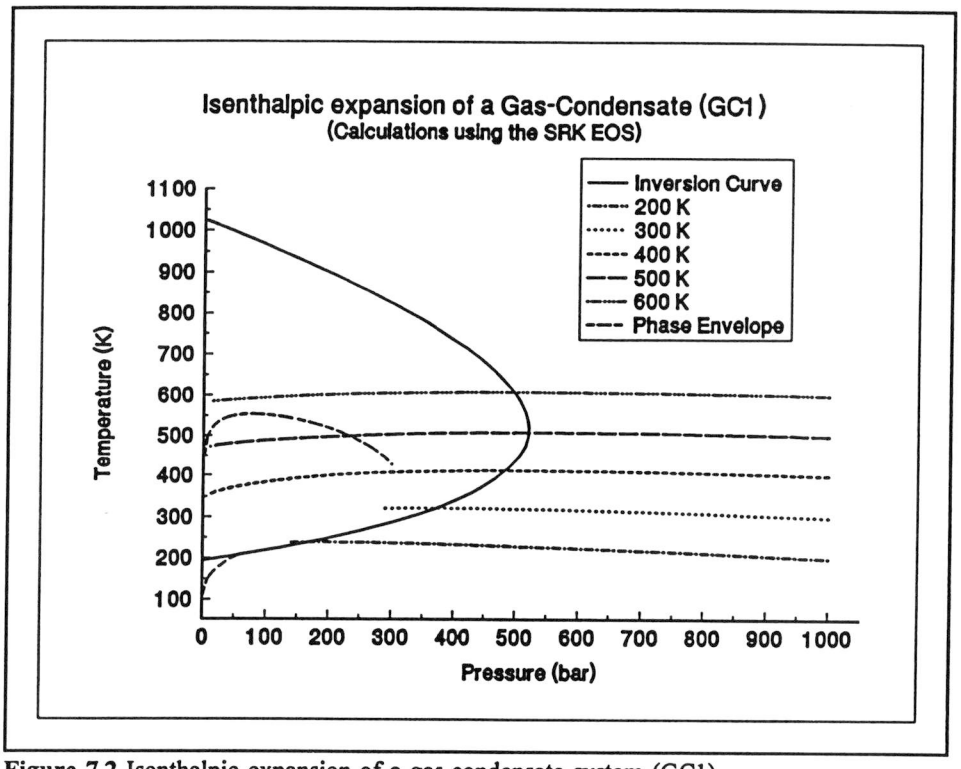


Figure 7.2 Isenthalpic expansion of a gas-condensate system (GC1)

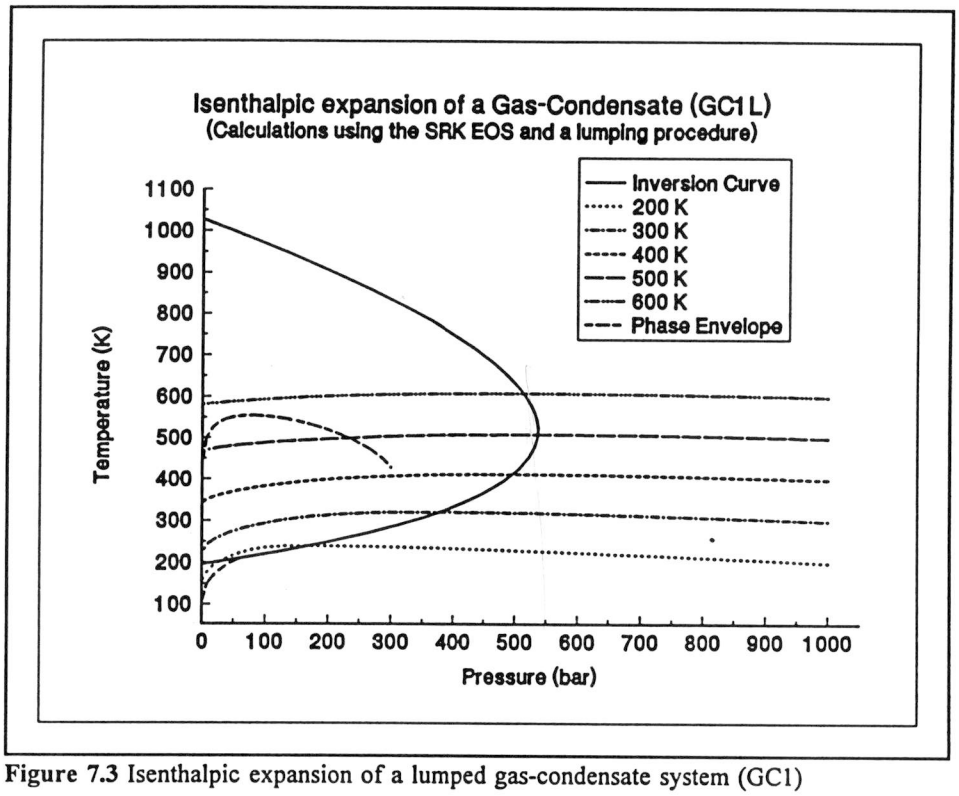


Figure 7.3 Isenthalpic expansion of a lumped gas-condensate system (GC1)

Now that we have seen that is indeed possible for gas-condensate reservoirs to heat when pressure drops, we would like to know how accurate these predictions are and, therefore, they must be compared with experimental data. As mentioned before, isenthalpic data and inversion data is not available for gas-condensate systems and, therefore, a direct comparison is not possible.
Figures 7.2 and 7.3 show that the temperature region of interest lies in the area of the lower branch and the peak of the inversion curve. In chapter 5 it was shown that this part of the inversion curve is well determined by equations of state, especially by the SRK equation. Both SRK and PR accurately describe the lower branch and the SRK predictions also follow the peak of the curve nicely. To give an indication of how accurate isenthalpic expansion results are, a comparison is made for a natural gas for which isenthalpic data is available from Buckley (1941). Composition of this natural gas are given in appendix F. Isenthalpic lines were calculated for this system using both the SRK and the PR equation. The heptane-plus fraction of the mixture is treated as n-octane. Because deviations between SRK and PR are negligible, figure 7.4 only shows calculated results for SRK along with the experimental data. These results are very much in agreement with the results shown in figure 6.2 for methane throttling, where SRK also slightly over-predicted final temperatures. Although these results look promising, it must be emphasized that larger deviations are expected for systems containing components heavier than heptane, as is the case with gas-condensates. Therefore, it is essential that on continuation of this work experimental data of expanding gas-condensate-like systems must come available. Either data that is now still confidential or through own experimental work.

7.5 Sensitivity study

To examine the effect of several factors on the Joule-Thomson effect, a sensitivity study was performed. This study shows the influence of the choice of an equation of state, the choice of the characterization procedure, the composition of the reservoir fluid, and the reservoir conditions.

To examine the influence of equations of state, the isenthalpic calculations of paragraph §7.4 for GC1 (using lumping) were repeated using both the SRK and the PR EoS. Figure 7.5 shows the results both for equations. Only the temperature region of interest is shown and, therefore, inversion curves are cut off. Both curves follow the same trend as in chapter 5, i.e. increasing deviations in the area of the peak and the upper branch, where the maximum inversion pressure and maximum inversion temperature calculated by PR are much higher than those calculated by SRK.

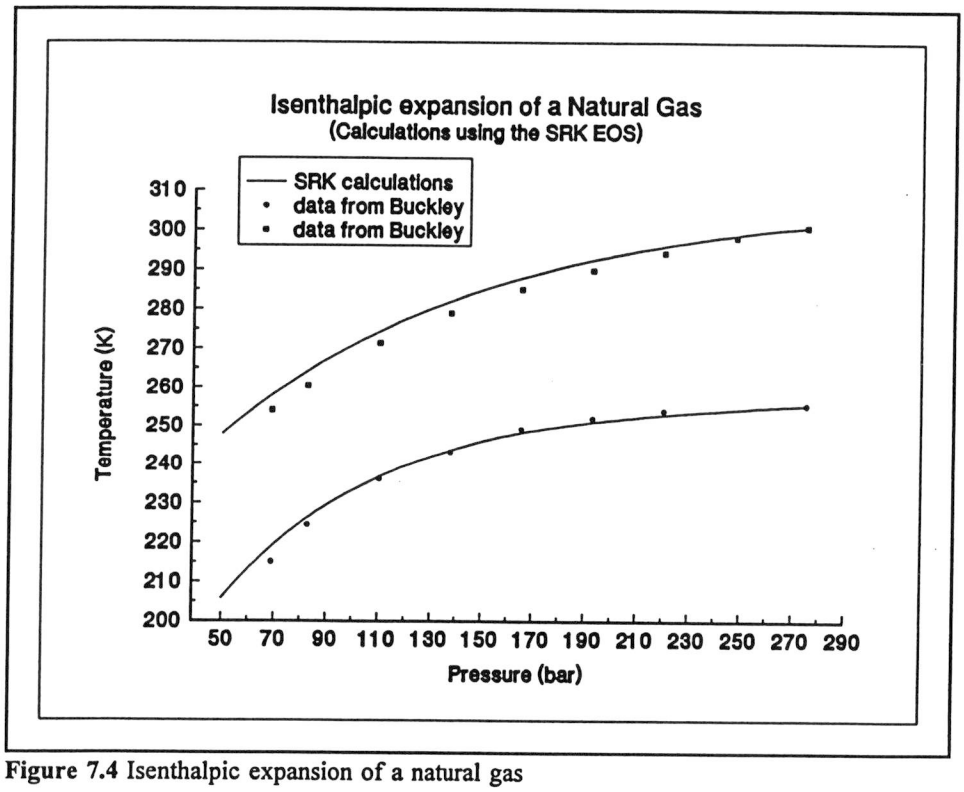


Figure 7.4 Isenthalpic expansion of a natural gas

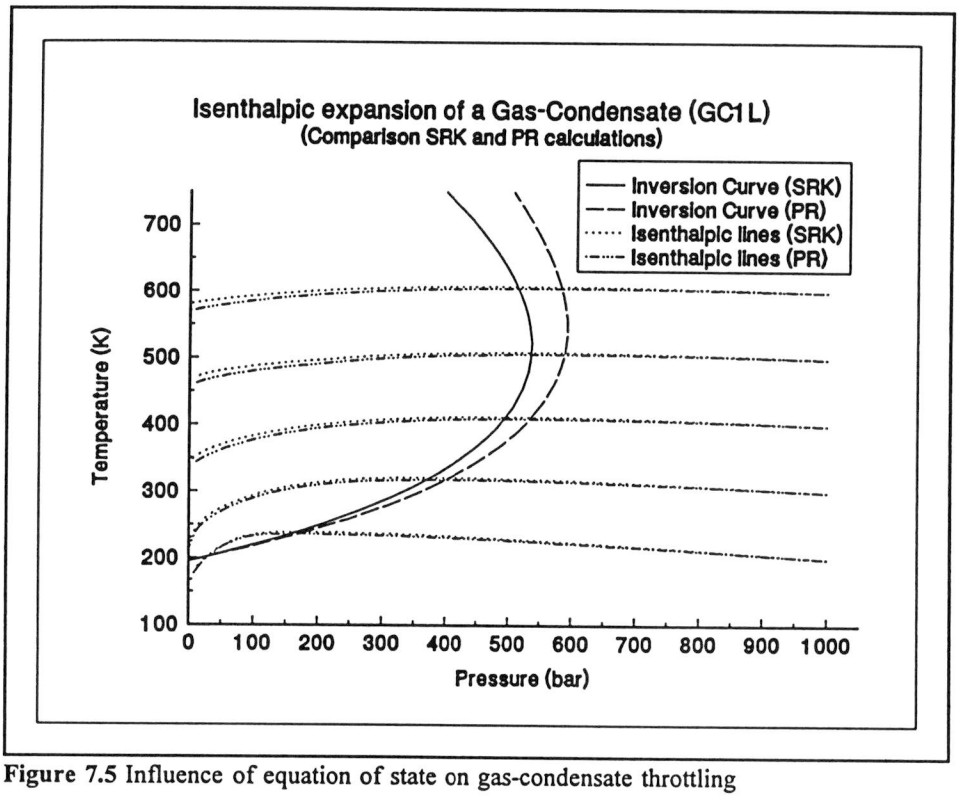


Figure 7.5 Influence of equation of state on gas-condensate throttling

Although the inversion curves differ much, the effect on isenthalpic lines is not large. When starting at 300 K, the maximum temperature increase is 21 K for SRK versus 18 K for PR, and the final temperatures (final pressure 1 bar) are 225 and 217 K, respectively. When starting at 500 K these values are 9 and 6 K for temperature increase and 471 and 461 K for final temperature.
Although the SRK equations gives better predictions for the inversion curve, it is not possible to draw any conclusions which equation gives the best results. The only conclusion that can be drawn is that, not only based on this example, the calculated isenthalpic lines using SRK lie somewhat above those calculated using PR and, therefore, SRK will predict higher temperature increases.

Figure 7.6 shows the influence of characterization of the heavy fractions on the temperature change. The gas-condensate system used was GC7 and all three characterization procedures were used to calculate the properties of the heavy fractions, and then components were grouped together.
The results of this characterization are shown in table 7.3. The figure shows that temperature deviations between the different procedures are not very large. Temperature increase using the Aasberg-Petersen relations are somewhat higher, but differences with the other two procedures diminish at high pressure drops. These differences are caused by the higher values of T_c and P_c that the Aasberg-Petersen relations predict for the plus fraction (C_{15}-C_{45}). Predictions using the Pedersen relations and the Cavett relations are very similar, which is reflected in figure 7.6, where differences between these two are so small, that they are hardly distinguishable. This example, therefore, shows that although temperatures differences for this example are not very large, the choice of the characterization procedure is not unimportant, because relatively small differences in predicted properties of the heavy fractions are directly converted in deviations in temperature change.

To examine the effect of composition on the Joule-Thomson effect, isenthalpic calculations were performed for several gas-condensate mixtures. The initial state is 400 K and 1000 bar for all cases. Figure 7.7 shows isenthalpic lines, calculated with SRK and the Aasberg-Petersen characterization procedure for six gas-condensates mixtures (GC1-GC6), a volatile oil (OIL1) and a North Sea Black oil (OIL2). Compositions of all petroleum mixtures are given in appendix F. Calculations for methane (C1) are included for comparison. The difference in composition is expressed as a difference in average molecular weight of the mixture. The figure clearly shows that with increasing molecular weight the temperature increase becomes larger.

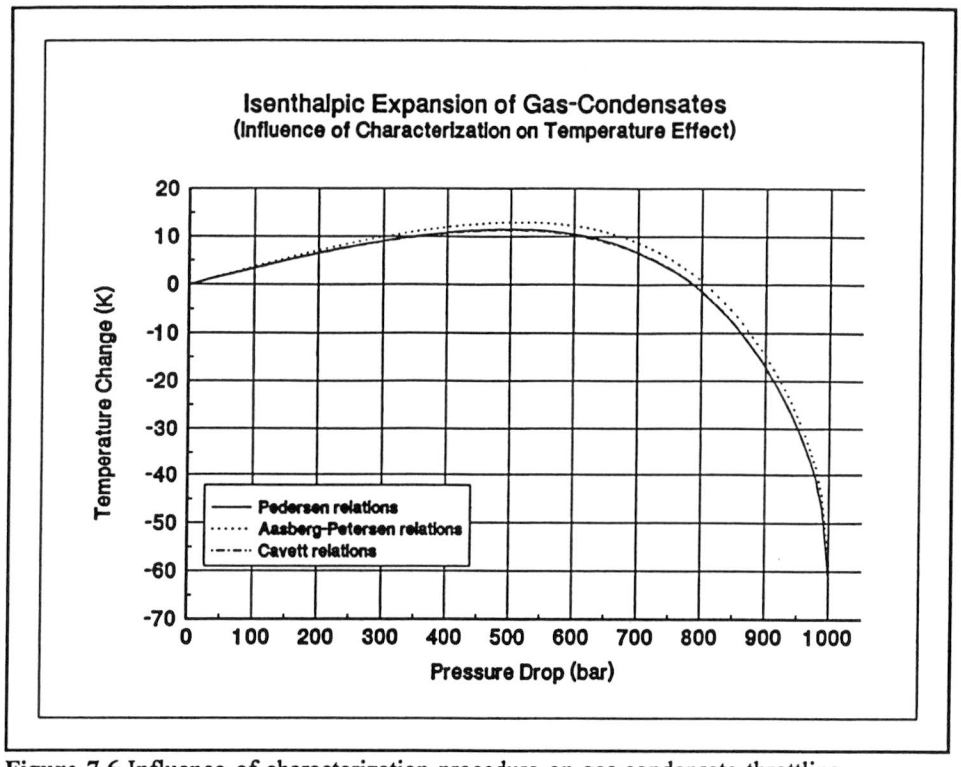


Figure 7.6 Influence of characterization procedure on gas-condensate throttling

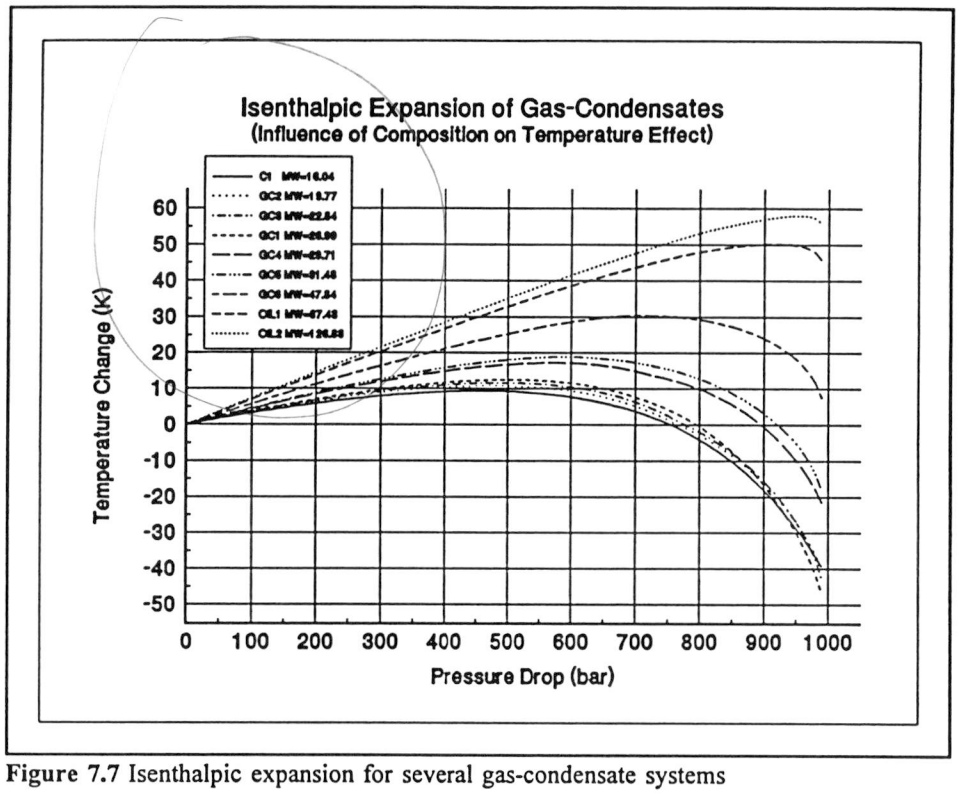


Figure 7.7 Isenthalpic expansion for several gas-condensate systems

From this one may conclude that (positive) temperature changes increase for systems with increasing amounts of heavy constituents. It also confirms what was found in the previous paragraph, i.e. for the most common gas-condensate reservoirs (MW ± 20-30 g/mole) the maximum temperature increase does not exceed 20 K (for this initial state).
To get an even clearer view on the influence of molecular weight on temperature change, a plot was constructed of temperature change versus molecular weight. Therefore, the temperature change for the mixtures shown in figure 7.7 were calculated for several pressure drops and plotted as shown in figure 7.8. Notice the dip for molecular weights between 20 and 30 g/mole at high pressure drops. This is caused by condensing of the heavy hydrocarbons.
Returning to the observations reported by Baker and Price (1991), who reported a temperature rise of some 15 to 20 °F (8.3 to 11.1 K) when the reservoir drawdown was approximately 7000 psia (482.3 bar). The exact composition of the reservoir fluid is not given, only that the molecular weight is 31.0 g/mole, which is very close to that of GC5 (31.4 g/mole). Therefore using the composition of GC5 an attempt was made to reproduce these observed results. The initial reservoir conditions reported by Baker were used, 16000 psia (1102.4 bar) and 400 °F (477.6 K), and pressure was relieved to 482.3 bar using both the SRK and PR equations of state and the Pedersen characterization procedure. The SRK equation calculated a temperature increase of 14.5 K and the PR equation 10.4 K. So both equations slightly overpredict the values reported by Baker and Price, but keeping in mind that the reported reservoir conditions are only approximations and the fact that GC5 is only a substitute, results look promising.
The Aasberg-Petersen characterization procedure resulted in calculated temperature increases of 18.5 and 14.8 K for SRK and PR, respectively and, therefore, it seems that by using the Pedersen procedure better results are obtained than with the Aasberg-Petersen procedure.

Finally, the influence of reservoir conditions was examined. At this point is must be emphasised that for the calculations performed so far, the actual reservoir conditions of the system were not taken into account. For instance the actual reservoir conditions of GC6 are 140-170 °C and 390 bar and, therefore, a pressure drop in this system of 1000 bar, as was done, is of course not possible. The only gas-condensate systems that were actually found at high pressure (1100 bar and 185-195 °C) and for which composition data are available, are two systems reported by Montel (1994). These systems are unfortunately useless for calculation purposes because molecular weights and densities are not given for the heavy fractions. However table F2 in appendix F shows that the compositions of these high pressure systems (MW = 37.99 g/mole and 32.15 g/mole) are

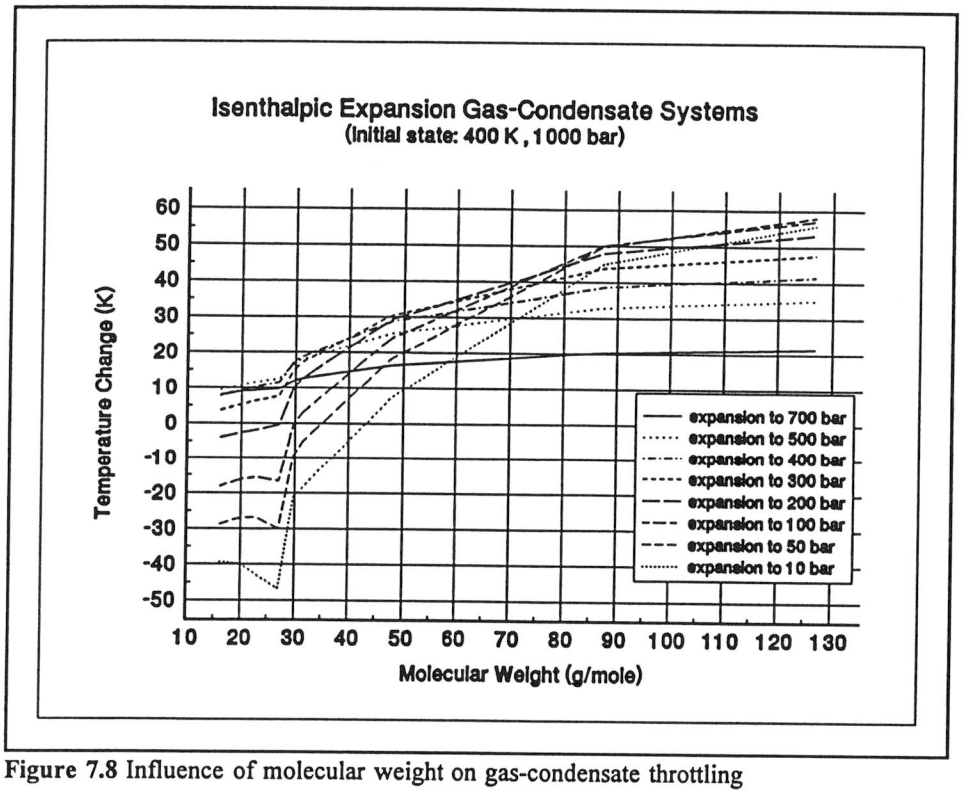


Figure 7.8 Influence of molecular weight on gas-condensate throttling

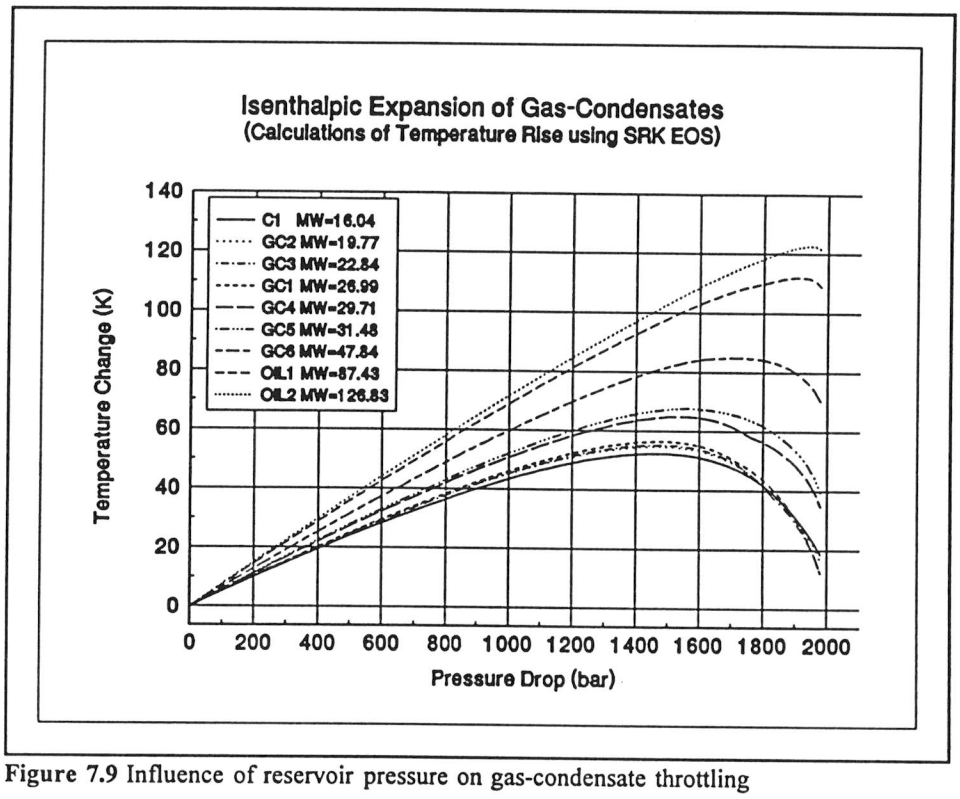


Figure 7.9 Influence of reservoir pressure on gas-condensate throttling

not much different from the other systems investigated (GC1-GC7) and, therefore, it seems to be admissible to use these systems for higher pressure drops.
The influence of reservoir temperature is best shown using figure 7.3, which shows isenthalpic lines for GC1 for different initial temperatures. The figure shows that with decreasing reservoir temperature, the maximum temperature rise increases. These results are in agreement with those found in chapter 4 for the Joule-Thomson coefficient. At low temperatures, i.e. the high density region, Joule-Thomson coefficients are smaller (i.e. more negative) at high pressures and larger at low pressures. They are also in agreement with results found in §5.3, where it was shown that with decreasing temperatures, the excess of pressure energy change over isothermal entropy change increases.

To show the influence of reservoir pressure, the calculations of figure 7.7 were repeated, but now starting at a pressure of 2000 bar. Figure 7.9 clearly shows that increasing pressure significantly influences the amount of temperature increase. This is not surprising because temperatures will rise until the inversion curve is reached, and the locus of the curve is not influenced by the initial pressure. Therefore, increasing reservoir pressure will cause increasing temperature rise.

Another factor that is of influence, is the drawdown or pressure drop. Because reservoir temperatures are well below the maximum inversion temperature, the temperature change as a function of pressure drop all follow the same trend. Initially temperatures increase (if the initial pressure is higher than the inversion pressure), and after reaching the inversion curve, temperatures will decrease. The maximum temperature increase is thus obtained if the pressure drops to inversion pressure.

7.6 Recapitulation

This chapter has dealt with the Joule-Thomson effect gas-condensate reservoirs. It was shown that temperatures in gas-condensate reservoirs can increase due to the Joule-Thomson inversion effect. If the reservoir pressure is higher than the inversion pressure (at reservoir temperature), the temperature will increase until the inversion curve is reached. In the temperature region of interest (300-500 K) the reservoir temperature does not influence the inversion effect itself, only the amount of temperature increase. The inversion effect is strongly affected by the composition of the reservoir fluid. With increasing amounts of heavy constituents, the maximum possible temperature rise will also increase. The maximum temperature increases one can expect are not very large. Although exact values depend on composition, reservoir conditions, the equation of state,

and characterization procedure used, typical maximum temperature increases are in the range of 10-30 K.

The results obtained so far look promising, but it must be kept in mind that these results are merely predictions. To draw any conclusions regarding the accuracy of the calculated temperature effects, experimental data have to come available for comparison. This could be data that is still confidential at the moment, or data obtained through own experimental work. Until this kind of data comes available, there is no certainty if (possible) deviations are caused by inaccurate calculation of enthalpies, heat capacity coefficients, molar volumes, or inaccurate description of the heavy fractions and ,therefore, no conclusions can be drawn at this moment on the capability of the equations of state and characterization procedures used in this work.

8. Conclusions and Recommendations

The purpose of this work was to examine Joule-Thomson expansion of high-pressure, high-temperature gas-condensate reservoirs and the possibility of these reservoir fluids to heat at expansion. Therefore, an investigation has been made on the Joule-Thomson inversion effect. Joule-Thomson coefficients have been calculated, inversion curves have been constructed, and isenthalpic expansions have been modelled for pure and multi-component mixtures in the single and two phase region. Calculations were performed using three well-known cubic equations of state: Redlich-Kwong, Soave-Redlich-Kwong and Peng-Robinson.

Because no experimental data was available on Joule-Thomson effects in gas-condensate systems, this work had an entirely predictable basis. To get more insight into the accuracy of the predictions, calculations were compared with available experimental data of some light constituents of gas-condensate mixtures: nitrogen, carbon dioxide, and methane.

Joule-Thomson coefficients were calculated for nitrogen, methane, a mixture of methane and butane, and two natural gases. The results showed that the Redlich-Kwong equation gave the best overall predictions despite its simplicity. In the low temperature and pressure region all equations showed significant deviations. RK and SRK predictions showed increasing accuracy with increasing temperatures and pressures. In the temperature region 250-400 K at low pressures, PR predictions were too high. All three equations gave satisfactory predictions in the high pressure region.

Inversion curves were calculated for nitrogen, carbon dioxide, methane, and a mixture of methane and butane. Although the RK equation gave good results for components with small acentric factors, e.g. nitrogen and methane, calculating the inversion curve of carbon dioxide showed the importance of including the acentric factor in inversion curve calculations. With increasing acentric factor, the maximum inversion temperature decreases and the maximum inversion pressure increases. Unlike the SRK and the PR equations, the RK equation was not capable to predict this trend and, therefore, it is expected that RK is less capable for gas-condensate calculations. Both the SRK and PR equation gave accurate predictions of the lower part of the inversion curve. The SRK gave the better description of the peak of the curve, where PR predicted a maximum

inversion pressure that is too high. However, the PR equation was capable of characterizing the upper part of the curve, where SRK predictions were too low.

To model the isenthalpic expansion of multi-component mixtures, a computer scheme was developed including a phase-stability test. In this scheme an initial estimate of the final temperature is made using the Joule-Thomson coefficient in the initial state. The stability is checked, i.e. the number of phases present is calculated, and then the isenthalpic expansion is iteratively solved for the single or two-phase problem. In the final state the stability is checked again. The Michelsen flash routines were used to perform the stability test.

Isenthalpic expansion simulations were performed using several gas-condensate mixtures found in the literature. These calculations showed that temperatures in gas-condensate reservoirs can rise at expansion due to the Joule-Thomson inversion effect. During isenthalpic expansion pressure energy is converted into entropy. Heating of the fluid occurs if the change in pressure energy exceeds the isothermal entropy change.
Heating of the reservoir fluids is caused by high reservoir pressures and not by high reservoir temperatures. For high pressure reservoirs, the pressure is higher than the inversion pressure (at reservoir temperature) and the temperature will increase until the inversion curve is reached. Because typical reservoir temperatures (300-500 K) are well below the calculated maximum inversion temperature (typically 1000 K), the reservoir temperature does not influence the inversion effect itself, only the amount of temperature increase, i.e. with decreasing reservoir temperature, maximum possible temperature rise increases. The inversion effect is strongly effected by the composition of the reservoir fluid. With increasing amounts of heavy constituents, and thus increasing molecular weight, the maximum possible temperature rise will also increase. Although it is possible to observe temperature increases, these increases are not that dramatic as the introduction may have suggested. Exact values depend on composition, reservoir conditions, pressure drop, and the equation of state and characterization procedure used, but typical maximum temperature rises are in the range of 10-30 K for reservoir pressures of 1000 bar. Increasing reservoir pressures will cause significantly more heating. Calculated temperature increases in reservoirs with pressures of 2000 bar varied from 55 to 80 K.

The results obtained so far look promising, but it must be kept in mind that the calculated results are merely predictions. No experimental data for isenthalpic expansion of gas-condensate systems was available at this time. To draw any conclusions regarding the accuracy of the calculated temperature effects, experimental data must come available for

comparison. This could be data that is still confidential at the moment, or data obtained through own experimental work. Until this kind of data comes available, there is no certainty if (possible) deviations are caused by inaccurate calculation of enthalpies, heat capacity coefficients, or inaccurate description of the heavy fractions and, therefore, no conclusions can be drawn on the applicability of the equations of state and characterization procedures used in this work.

Besides the availability of experimental data, a recommendation is made to improve the computer programs. The isenthalpic flash routine used, is a simple two-dimensional Newton-Raphson iteration scheme. In a few cases many iterations were required which could be caused by the numerical derivatives used in this algorithm. The first step should be to replace these by analytical expressions to rule out the possibility that deviations in the derivatives slow down convergence. Problems were also encountered with the Michelsen flash routines, were in several cases convergence failed in the two-phase region. Using lumping procedures to reduce the number of components eliminated these problems, but the reason for convergence failure remains unknown. Finally to calculate complete phase envelopes for gas-condensate systems, a program with is much more robust than ThermoPack™ must come available.

List of Symbols and Abbreviations

Nomenclature			Units
a,b	=	parameters in equations of state	
A,B	=	reduced parameters in equations of state	[-]
A	=	Helmholtz energy	J
C_p	=	specific heat at constant pressure	J $mole^{-1}$ K^{-1}
f	=	fugacity	Pa
g	=	gravitational constant	m s^{-2}
G	=	Gibbs energy	J
H	=	enthalpy	J $mole^{-1}$
k_{ij}	=	binary interaction parameter	[-]
K	=	equilibrium ratio	[-]
K	=	Watson characterization factor	[-]
n	=	mole number	mole
N	=	number of components	[-]
m	=	function of ω in EoS	[-]
MW	=	molecular weight	g $mole^{-1}$
P	=	pressure	Pa
Q	=	heat	kW
R	=	gas constant	J $mole^{-1}$ K^{-1}
S	=	entropy	J $mole^{-1}$ K^{-1}
SG	=	specific gravity	[-]
T	=	temperature	K
U	=	internal energy	J
v	=	velocity	m s^{-1}
V	=	volumem	3
W	=	work	kW
x	=	mole fraction in liquid phase	[-]
x	=	reduced volume (v/b)	[-]
y	=	mole fraction in vapour phase	[-]
z	=	height	m
z	=	mole fraction of feed	[-]
Z	=	compressibility factor	[-]

greek symbols

α	=	temperature dependant part of a-parameter	
β	=	vapour fraction	[-]
ϵ	=	infinitesimal phase amount	mole
μ	=	chemical potential	
μ_{JT}	=	Joule-Thomson coefficient	K Pa^{-1}

ρ	=	density	kg mole^{-1}
ϕ	=	fugacity coefficient	[-]
Ω	=	constant in EoS	[-]
ω	=	acentric factor	[-]

superscripts

id	=	ideal gas state
L	=	liquid
R	=	residual property
(r)	=	reference fluid
V	=	vapour

subscripts

0	=	initial state
1	=	final state
a	=	a-parameter
b	=	b-parameter
B	=	boiling point
c	=	critical property
i	=	component i
I	=	inversion
j	=	component j
r	=	reduced property
ref	=	reference state
sat	=	saturation property

Abbreviations

EoS	=	equation of state
JT	=	Joule-Thomson
PR	=	Peng-Robinson
RK	=	Redlich-Kwong
SRK	=	Soave-Redlich-Kwong

References

Aasberg-Petersen, K., Stengby, E., **Prediction of Thermodynamic Properties of Oil and Gas Condensate Mixtures**, *Ind. Eng. Chem. Res.*, **1991**, 30, 248-254.

Baker, A.C., Price, M., **Modelling the performance of high-pressure high-temperature wells**, SPE European Petroleum Conference, 21-24 october 1990, 217-230.

Benedict, M., *J. Am. Chem. Soc.*, 59 (2), **1937**, 2232.

Bradley, **Petroleum Engineering Handbook**, SPE, Richardson, TX, USA, 2nd ed. 1989, chapter 23.

Budenholzer, R.A., Sage, B.H., Lacey, W.N, **Joule-Thomson Coefficient of Methane**, *Ind. Eng. Chem.*, Vol.31, No.3, March 1939, 369-374.

Budenholzer, R.A., Sage, B.H., Lacey, W.N, **Joule-Thomson Coefficients for Gaseous Mixtures of Methane and Ethane**, *Ind. Eng. Chem.*, Vol.31, No.10, October 1939, 1288-1292.

Budenholzer, R.A., Sage, B.H., Lacey, W.N, **Joule-Thomson Coefficients in the Methane-Propane System**, *Ind. Eng. Chem.*, Vol.34, No.7, July 1942, 878-882.

Budenholzer, R.A., Sage, B.H., Lacey, W.N, **Joule-Thomson Coefficients for Gaseous Mixtures of Methane and n-Butane**, *Ind. Eng. Chem.*, Vol.32, No.3, March 1940, 384-387.

Cavett, R.H., **Physical Data for Distillation Calculation, Vapor-Liquid Equilibria**, 27th Midyear Meeting, API Division of Refining, San Francisco, CA, May 15, 1964.

Chou, F.C., Wu, S.M., Pai, C.F., **Prediction of final temperature following Joule-Thomson expansion of nitrogen gas**, *Cryogenics*, **1993** Vol.33, No.9, 857-862.

Corner, J., **The Joule-Thomson inversion curves of recent equations of state**, *Trans. Farad. Soc.*, 35 (1939), 784.

Deming, W.E., Deming, L.S., **Some Physical Properties of Compressed Gases. V. The Joule-Thomson Coefficient for Nitrogen**, *Physical Review*, Vol.48, **1935**, 448-457.

Dilay, G.W., Heidemann, R.A, **Calculation of Joule-Thomson Inversion Curves from Equations of State**, *Ind. Eng. Chem. Fundam.*, **1986**, 25, 152-158.

Din, F. **Thermodynamic Functions of Gases**, Vol.1, London, 1956, 116,156.

Din, F. **Thermodynamic Functions of Gases**, Vol.2, London, 1956, 125.

Din, F. **Thermodynamic Functions of Gases**, Vol.3, London, 1961, 35.

Edminster, W.C., **Applications of Thermodynamics to Hydrocarbon Processing; Part XV Joule-Thomson Effect**, *Petroleum Refiner*, Vol.28, No.1, January 1949, 128-133.

Edminster, W.C., **Applied Hydrocarbon Thermodynamics**, Volume 2, Houston, Gulf Publishing Company, 1988.

Geană, D., Feroiu, V., **Calculation of Joule-Thomson inversion curves from a general cubic equation of**

state, *Fluid Phase Equilibria*, 77 (1992), 121-132.

De Groot, S.R., Michels, A., **The Joule-Thomson Effect and The Specific Heat at Constant Pressure of Carbon Dioxide**, *Physica*, 14 No.4, May 1948, 218-222.

Gunn, R.D., Chueh, P.L., Prausnitz, J.M., **Inversion Temperatures and Pressures for Cryogenic Gases and their Mixtures**, *Cryogenics*, 6 (1966), 324-329.

Heyes, D.M., Llaguno C.T., **Computer simulation and equation of state study of the Boyle and inversion temperature of simple fluids**, *Chemical Physics*, 168 (1992) 61-68.

Yasuo Hirose, Tsuyoshi Kitazawa, Toshinobu Yoshida, **The Joule-Thomson Coefficient by Formula Manipulation**, *Ind. Eng. Chem. Res.*, **1990**, 29, 1555-1558.

International Thermodynamic Tables of the Fluid State - 5 Methane, Pergamon Press, Oxford UK, 1st ed., 1978.

Juris, K., Wenzel, L.A., **A Study of Inversion Curves**, *AIChE Journal* (Vol.18, No.4), July 1972, 684-688.

Katz, D.L. and Firoozabadi, A., **Predicting Phase Behavior of Condensate/Crude-Oil Systems Using Methane Interaction Coefficients**, *J. Pet. Technol.*, 20, 1978, 1649-1655.

Kesler, M.G., and Lee, B.I., **Improve Prediction of Enthalpy of Fractions**, *Hydrocarbon Processing*, 55, 1976, 153-158.

Koeppe, W., **On the Inversion Curve at Low Temperatures and Theorem of Corresponding States**, Proceedings of 10th International Congress on Refrigeration (Copenhagen 1959), Vol. 1, New York (1960), 156-163.

Michelsen, M.L., **The Isothermal Flash Problem. Part I. Stability**, *Fluid Phase Equilibria*, 9 (1982) 1-19.

Michelsen, M.L., **Multiphase Isenthalpic and Isentropic Flash Algorithms**, *Fluid Phase Equilibria*, 33 (1987) 13-27.

Miller, D.G., **Joule-Thomson Inversion Curve, Corresponding States, and Simpler Equations of State**, *Ind. Eng. Chem. Fundam.*, Vol.9, No.4, 1970, 585-589.

Montel, F., **Elf Aquitaine Production**, private communication, 1994.

Nagy, S., **The influence of hydrocarbon condensation on natural gas throttling temperature**, *Archiwum Termodynamiki*, Vol.12 (1991) No 1-4, 101-115.

Najjar, Y.S.H., Al-Beirutty, M.H., Ismail, M.S., **Two-constant equation of state for accurate prediction of the Joule-Thomson inversion curve for air in cryogenic applications**, *Cryogenics* 1993, Vol.33, No.2, 169-174.

Pedersen, K.S., Thomassen, P., Fredenslund, A., **SRK-EOS Calculation for Crude Oils**, *Fluid Phase Equilibria*, 14 (1983) 209-218.

Pedersen, K.S., Thomassen, P., Fredenslund, A., **Thermodynamics of Petroleum Containing Heavy Hydrocarbons. 1. Phase Envelope Calculations by Use of the Soave-Redlich-Kwong Equation of State**, *Ind. Eng. Chem. Process Des. Dev.*, **1984**, 23, 163-170.

Pedersen, K.S., Thomassen, P., Fredenslund, A., **Thermodynamics of Petroleum Containing Heavy Hydrocarbons. 2. Flash and PVT Calculations with the SRK Equation of State,** *Ind. Eng. Chem. Process Des. Dev.*, **1984**, 23, 566-573.

Pedersen, K.S., Thomassen, P., Fredenslund, A., **Thermodynamics of Petroleum Containing Heavy Hydrocarbons. 3. Efficient Flash Calculation Procedures Using the SRK Equation of State,** *Ind. Eng. Chem. Process Des. Dev.*, **1985**, 24, 948-954.

Pedersen, K.S., Thomassen, P., **Characterization of Gas Condensate Mixtures**, Pare presented at the 1988 AIChE Spring National Meeting New Orleans, Louisiana, MArch 6-10, 1988.

Pedersen, K.S., Thomassen, P., Fredenslund, A., **Properties of Oils and Natural Gases**, Gulf Publishing Company, Houston, USA (1989).

Pitzer, K.S., *J. Am. Chem. Soc.*, Vol.77, **1955**, 3427.

Prausnitz, J.M., Anderson, T., Grens, E., Eckert, C., Hsieh, H. and O'Connell, J., **Computer Calculations for Multicomponent Vapor-Liquid and Liquid-Liquid Equilibria**, Prentence-Hall, New Jersey (1980), Chapter 7.

Price, D., **Thermodynamic Functions of Carbon Dioxide**, *Chem. Eng. Data Ser.*, 1 (1956), 83-86.

Reid, R.C., Prausnitz, J.M. and Sherwood, T.K., **The Properties of Gases and Liquids**, McGraw-Hill, New York (1987).

Roebuck, J.R., Osterberg,H., **The Joule-Thomson Effect in Nitrogen**, *Physical Review*, 48 (1935), 450.

Roebuck, J.R.,Murrell, T.A., Miller, E.E. **The Joule-Thomson Effect in Carbon Dioxide**, *J. Am. Chem. Soc.*, 64 (1942), 400-411.

Sage, B.H., Botkin, D.F., Lacey, W.N., **Joule-Thomson Coefficients for Two Natural Gases**, *Amer. Inst. of Mining & Metallugical Engineering*, Tech. Pub. No. 1504, *Petroleum Technology*, (Sept. 1942).

Sage, B.H., Webster, Lacey, W.N., **Joule-Thomson Coefficient of Ethane**, *Ind. Eng. Chem.*, Vol.29, No.6, June 1937, 665.

Sage, B.H., Kennedy, E.R., Lacey, W.N., **Joule-Thomson Coefficient of Propane**, *Ind. Eng. Chem.*, Vol.28, No.5, May 1936, 601-604.

Smith, J.M., Van Ness, H.C., **Introduction to Chemical Engineering Thermodynamics**, Fourth Edition, McGraw-Hill, Singapore (1987).

Ullmann's Encyclopedia of Industrial Chemistry, Verlag Chemie, 5th ed., Weinheim, 1985, Vol.B Unit Operations II, 20-6.

Wark, K., **Thermodynamics**, McGraw-Hill, New York, USA (1988), 5th ed., 512-515.

Wolfe, J.F., **Predicting Phase and Thermodynamic Properties of Natural Gases with the Benedict-Webb-Rubin Eauation of State**, *Journal of Petroleum Technology*, March 1966, 364-371.

Zemanski, M.W, Dittman, R.H., **Heat and Thermodynamics**, Sixth Edition, McGraw-Hill, Singapore (1981).

Appendices

Appendix A:	Physical Properties
Appendix B:	Temperature Derivative of Molar Volume
Appendix C:	Calculated Joule-Thomson Coefficients
Appendix D:	Extraction of T from a(t)
Appendix E:	Analytical Derivatives for Isenthalpic Flash Routine
Appendix F:	Petroleum Reservoir Fluid Compositions
Appendix G1:	Description of the Computer Program
Appendix G2:	Joule-Thomson FORTRAN Input File
Appendix G3:	Michelsen Connection FORTRAN file
Appendix G4:	Input File Component.Dat
Appendix G5:	Input File Gas_Condensate.Dat
Appendix G6:	Input File Lumping.Dat

Table A.1 Physical properties of some common petroleum mixtures constituents†

Component	M_w (g/mole)	T_c (K)	P_c (bar)	ω	$C_{p,1}$	$C_{p,2}$ (10^{-1})	$C_{p,3}$ (10^{-4})	$C_{p,4}$ (10^{-8})
N_2	28.013	126.2	33.9	0.039	31.15	-0.1357	0.2680	-1.168
CO_2	44.010	304.1	73.8	0.239	19.80	0.7344	-0.5602	1.715
C_1	16.043	190.4	46.0	0.011	19.25	0.5213	0.1197	-1.132
C_2	30.070	305.4	48.8	0.099	5.409	1.781	-0.6938	0.8713
C_3	44.094	369.8	42.5	0.153	-4.224	3.063	-1.586	3.215
n-C_4	58.124	425.2	38.0	0.199	9.487	3.313	-1.108	-0.2882
i-C_4	58.124	408.2	36.5	0.183	-1.390	3.847	-1.846	2.895
n-C_5	72.151	469.7	33.7	0.251	-3.626	4.873	-2.580	5.305
i-C_5	72.151	460.4	33.9	0.227	-9.525	5.066	-2.729	5.723
n-C_6	86.178	507.5	30.1	0.299	-4.413	5.820	-3.119	6.494
n-C_7	100.205	540.3	27.4	0.349	-5.146	6.762	-3.651	7.658
n-C_8	114.232	568.8	24.9	0.398	-6.096	7.712	-4.195	8.855
n-C_{10}	142.286	617.7	21.2	0.489	-7.931	9.609	-5.288	11.31
n-C_{12}	170.340	658.2	18.2	0.575	-9.328	11.49	-6.347	13.59
n-C_{15}	212.421	707.0	15.2	0.706	-11.92	14.33	-7.972	17.20
n-C_{20}	298.555	767.0	11.1	0.907	-22.38	19.39	-11.17	25.28
n-C_{24}‡	338.659	818.0	9.0	0.942	-18.928	22.84	-12.82	27.44

† Reid et al. (1987)
‡ SUPRTRAP

Table A.2 Values for the binary interaction parameter kij†

Component	N_2		CO_2	
	PR	SRK	PR	SRK
N_2	0.0	0.0	-0.017	-0.032
CO_2	-0.017	-0.032	0.0	0.0
C_1	0.031	0.028	0.092	0.093
C_2	0.052	0.041	0.132	0.136
C_3	0.085	0.076	0.124	0.129
n-C_4	0.080	0.070	0.133	0.143
i-C_4	0.103	0.094	0.120	0.128
n-C_5	0.100	0.088	0.122	0.131
i-C_5	0.092	0.087	0.122	0.131
n-C_6	0.150	0.150	0.110	0.118

† Reid et al. (1987)

‡ for alkanes with a carbon number larger than 6, values are the same as for n-C6

Method of Joback

The method of Joback is a group-contribution method to calculate the values of $C_{p,1}$-$C_{p,4}$. Group values of interest for alkanes are shown in table A.3. Values for $C_{p,1}$-$C_{p,4}$ can be estimated from the following relations:

$$C_{p,1} = \sum_i n_i \Delta_a - 37.93 \qquad \textbf{(A.1)}$$

$$C_{p,2} = \sum_i n_i \Delta_b + 0.210 \qquad \textbf{(A.2)}$$

$$C_{p,3} = \sum_i n_i \Delta_c - 3.91\,10^{-4} \qquad \textbf{(A.3)}$$

$$C_{p,4} = \sum_i n_i \Delta_d + 2.06\,10^{-7} \qquad \textbf{(A.4)}$$

where n_i is the number of groups of the i^{th} type and the Δ contribution are for i^{th} atomic or molecular group.

Table A.3 Joback Group Contribution for Ideal-Gas Properties

Group	Δ_a	Δ_b	Δ_c	Δ_d
$-CH_3$	1.95E+1	-8.08E-3	1.53E-4	-9.67E-8
$>CH_2$	-9.09E-1	9.50E-2	-5.44E-5	1.19E-8

Table A.4 shows the calculated values of $C_{p,1}$-$C_{p,4}$ for C_{15}, C_{20}, and C_{24} using the Joback method. Comparing the calculated coefficient values for C_{15} and C_{20} with those in table A.1 shows that predictions are quite good.

Table A.4 Calculated values of $C_{p,1}$-$C_{p,4}$ using the Joback method

Component	$C_{p,1}$	$C_{p,2}$	$C_{p,3}$ (10^{-4})	$C_{p,4}$ (10^{-7})
n-C_{15}	-10.747	1.429	-7.922	1.673
n-C_{20}	-15.292	1.904	-10.642	2.268
n-C_{24}	-18.928	2.284	-12.818	2.744

Cubic equations of state cannot be written by v in an explicit form and therefore, to get an analytical expression for the temperature derivative of the molar volume the following cyclic relation is used:

$$\left(\frac{\partial V}{\partial T}\right)_P = -\left(\frac{\partial P}{\partial T}\right)_V \left(\frac{\partial P}{\partial V}\right)_T^{-1} \tag{B.1}$$

The derivative of pressure to temperature and volume for the different equations of state are as follows:

The RK equation:

$$\left(\frac{\partial P}{\partial T}\right)_V = \frac{R}{V-b} + \frac{1}{2}\frac{a}{T^{1.5}V(V+b)} \tag{B.2}$$

$$\left(\frac{\partial P}{\partial V}\right)_T = -\frac{RT}{(V-b)^2} + \frac{a}{\sqrt{T}V^2(V+b)} + \frac{a}{\sqrt{T}V(V+b)^2} \tag{B.3}$$

The SRK equation:

$$\left(\frac{\partial P}{\partial T}\right)_V = \frac{R}{V-b} - \frac{\frac{\partial}{\partial T}a(T)}{V(V+b)} \tag{B.4}$$

$$\left(\frac{\partial P}{\partial V}\right)_T = -\frac{RT}{(V-b)^2} + \frac{a(T)}{V^2(V+b)} + \frac{a(T)}{V(V+b)^2} \tag{B.5}$$

The PR equation:

$$\left(\frac{\partial P}{\partial T}\right)_V = \frac{R}{V-b} - \frac{\frac{\partial}{\partial T}a(T)}{V(V+b)+b(V-b)} \tag{B.6}$$

$$\left(\frac{\partial P}{\partial V}\right)_T = -\frac{RT}{(V-b)^2} + \frac{a(T)(2V+2b)}{(V(V+b)+b(V-b))^2} \tag{B.7}$$

Table C.1 Calculated and experimental JT coefficients for methane †

T (K)	P (bar)	μ_{JT} (K/bar)			
		experiment	RK	SRK	PR
310.9	17.24	0.3698	0.3918	0.4082	0.4477
377.6	17.24	0.2361	0.2451	0.2531	0.2901
310.9	51.71	0.3602	0.3570	0.3681	0.3981
377.6	51.71	0.2296	0.2227	0.2274	0.2577
310.9	103.42	0.2901	0.2828	0.2894	0.3062
377.6	103.42	0.1773	0.1839	0.1862	0.2077

† Budenholzer et al. (1939)

Table C.2 Calculated and experimental JT coefficients for natural gas A†

T (K)	P (bar)	μ_{JT} (K/bar)			
		experiment	RK	SRK	PR
294.26	13.79	0.5620	0.5911	0.6234	0.6629
	27.58	0.5652	0.5874	0.6150	0.6499
	34.47	0.5628	0.5823	0.6081	0.6403
360.93	13.79	0.3693	0.3603	0.3800	0.4175
	27.58	0.3741	0.3518	0.3690	0.4032
	34.47	0.3717	0.3469	0.3628	0.3953
427.59	13.79	0.2290	0.2350	0.2441	0.2786
	27.58	0.2314	0.2280	0.2355	0.2675
	34.47	0.2306	0.2243	0.2310	0.2618

† Composition given in table F.1

The a-parameter is equal to:

$$a_m = \sum_i^N \sum_j^N x_i x_j (a_i a_j)^{0.5} (1 - k_{ij})$$

$$= \sum_i^N \sum_j^N x_i x_j \left(\Omega_a \frac{(R T_{ci})^2}{P_{ci}} \alpha_i \, \Omega_a \frac{(R T_{cj})^2}{P_{cj}} \alpha_j \right)^{0.5} (1 - k_{ij})$$

$$= \sum_i^N \sum_j^N x_i x_j \, \Omega_a R^2 \left(\frac{T_{ci}^2 T_{cj}^2}{P_{ci} P_{cj}} \alpha_i \alpha_j \right)^{0.5} (1 - k_{ij})$$

where: $$\alpha_i = \left[1 + m_i \left(1 - \left(\frac{T}{T_{ci}} \right)^{0.5} \right) \right]^2$$

$$\alpha_j = \left[1 + m_j \left(1 - \left(\frac{T}{T_{cj}} \right)^{0.5} \right) \right]^2$$

thus: $$a_m = \sum_i^N \sum_j^N x_i x_j \, \Omega_a R^2 \left[1 + m_i \left(1 - \left(\frac{T}{T_{ci}} \right)^{0.5} \right) \right] \left[1 + m_j \left(1 - \left(\frac{T}{T_{cj}} \right)^{0.5} \right) \right] \left(\frac{T_{ci}^2 T_{cj}^2}{P_{ci} P_{cj}} \right)^{0.5} (1 - k_{ij})$$

the term between brackets [ ][ ] is equal to:

$$= 1 + m_i \left(1 - \left(\frac{T}{T_{ci}} \right)^{0.5} \right) + m_j \left(1 - \left(\frac{T}{T_{cj}} \right)^{0.5} \right) + m_i m_j \left(1 - \left(\frac{T}{T_{ci}} \right)^{0.5} \right) \left(1 - \left(\frac{T}{T_{cj}} \right)^{0.5} \right)$$

$$= 1 + m_i - m_i T_{ri}^{0.5} + m_j - m_j T_{ri}^{0.5} + m_i m_j \left(1 - T_{ri}^{0.5} - T_{rj}^{0.5} + (T_{ri} T_{rj})^{0.5} \right)$$

$$= 1 + m_i + m_j + m_i m_j + T^{0.5} \left[- \frac{m_i}{T_{ci}^{0.5}} - \frac{m_j}{T_{cj}^{0.5}} - \frac{m_i m_j}{T_{ci}^{0.5}} - \frac{m_i m_j}{T_{cj}^{0.5}} \right] + T \left[\frac{m_i m_j}{T_{ci}^{0.5} T_{cj}^{0.5}} \right]$$

$$= (1 + m_i)(1 + m_j) - T^{0.5}\left[+ \frac{m_i(1 + m_j)}{T_{ci}^{0.5}} + \frac{m_j(1 + m_i)}{T_{cj}^{0.5}} \right] + T\left[\frac{m_i m_j}{T_{ci}^{0.5} T_{cj}^{0.5}} \right]$$

and therefore the a-parameter can be written as:

$$a_m = \Omega_a R^2 \sum_i^N \sum_j^N x_i x_j \frac{T_{ci} T_{cj}}{(P_{ci} P_{cj})^{0.5}} (1 + m_i)(1 + m_j)(1 - k_{ij})$$

$$- \Omega_a R^2 T^{0.5} \sum_i^N \sum_j^N x_i x_j \left(T_{cj} T_{ci}^{0.5} m_i (1 + m_j) + T_{ci} T_{cj}^{0.5} m_j (1 + m_i) \right) (P_{ci} P_{cj})^{-0.5} (1 - k_{ij})$$

$$+ \Omega_a R^2 T \sum_i^N \sum_j^N x_i x_j m_i m_j \left(\frac{T_{ci} T_{cj}}{P_{ci} P_{cj}} \right)^{0.5} (1 - k_{ij})$$

or simplified:

$$a_m = \Omega_a R^2 \left[Sum1 - T^{0.5} Sum2 + T Sum3 \right]$$

where Sum1, Sum2, Sum3 are given by equations 5.11 through 5.13.

De numerical derivatives in the two-dimensional isenthalpic flash procedure can be replaced by analytical derivatives. Reference is made to the original work by Michelen (1987) for further details on this matter. The four elements of the Jacobian matrix can be calculated as follows:

$$\frac{\partial g_1}{\partial \beta} = - \sum_{i=1}^{N} z_i (K_i - 1)^2 / \zeta_i^2 \qquad \textbf{(E.1)}$$

$$\frac{\partial g_2}{\partial \beta} = H^V - H^L \qquad \textbf{(E.2)}$$

$$\frac{\partial g_1}{\partial T} = \sum_{i=1}^{N} z_i \left(K_i \frac{\partial \ln K_i}{\partial T} - (K_i - 1)\tau_i^* / \zeta_i \right) / \zeta_i \qquad \textbf{(E.3)}$$

$$\frac{\partial g_2}{\partial T} = \beta C_p^V + (1 - \beta) C_p^L + RT^2 \sum_{i=1}^{N} z_i / \zeta_i / (\tau_i^{**} - (\tau_i^*)^2 / \zeta_i) \qquad \textbf{(E.4)}$$

where:

$$\zeta_i = 1 + \beta (K_i - 1)$$

$$\tau_i^* = \beta K_i \frac{\partial \ln K_i}{\partial T} \qquad \textbf{(E.5)}$$

$$\tau_i^{**} = \beta K_i \left(\frac{\partial \ln K_i}{\partial T} \right)^2$$

Table F.1 Composition of Natural Gases

Components	Natural Gas A †	Natural Gas B †	Natural Gas ‡
	Mole %		
CO_2	0.90	4.20	-
C_1	82.55	83.85	94.11
C_2	9.01	6.37	2.67
C_3	4.62	3.74	0.89
n-C_4	1.26	0.95	0.34
i-C_4	0.74	0.39	0.21
n-C_5	0.22	0.11	0.10
i-C_5	0.32	0.16	0.20
n-C_{6+}	0.38	0.23	0.29
C_{7+}	-	-	1.19

† Data taken from Sage et al. (1942)
‡ Data taken from Wolfe (1966)

Table F.2 Molar compositions of Gas-Condensate systems

Component	GC1 †	GC2 ‡	GC3 †	GC4 ♯	GC5 ‡	GC6 †	GC7 ‡	Montel ♮	Montel ♮	OIL1 ♭	OIL2 ‡
N_2	0.71	-	0.85	0.23	0.12	0.46	0.64	0.554	0.255	-	0.40
CO_2	8.64	-	0.65	2.12	2.49	3.36	9.16	2.619	3.598	-	0.10
C_1	70.85	91.34	83.58	73.15	76.43	62.23	68.80	68.444	74.117	52.00	45.40
C_2	8.53	4.03	5.95	9.39	7.46	8.88	8.43	8.981	7.944	3.81	4.20
C_3	4.95	1.53	2.91	4.86	3.12	5.30	5.11	4.240	3.294	2.37	0.89
n-C_4	1.26	0.43	1.11	1.78	1.21	2.07	1.45	1.517	1.242	0.96	0.52
i-C_4	0.75	0.39	0.45	1.02	0.59	0.91	0.81	0.801	0.680	0.76	0.56
n-C_5	0.40	0.19	0.48	0.49	0.59	0.85	0.53	0.658	0.608	0.51	0.29
i-C_5	0.41	0.15	0.36	0.60	0.50	0.73	0.52	0.676	0.552	0.69	0.65
C_6	0.46	0.39	0.60	0.93	0.79	1.04	0.63	1.390	0.874	2.06	1.01
C_7	0.61	0.361	0.80	5.43	0.95	1.84	0.83	1.707	1.147	2.63	2.88
C_8	0.71	0.285	0.76		1.08	1.73	0.95	1.495	1.072	2.34	4.11
C_9	0.39	0.222	0.47		0.78	1.39	0.52	1.121	0.748	2.35	3.32
C_{10}	1.33	0.672	1.03		0.592	9.21	0.26	0.912	0.565	29.52	35.67
C_{11}					0.467		0.20	4.884	3.302		
C_{12}					0.345		0.17				
C_{13}					0.375		0.16				
C_{14}					0.304		0.15				
C_{15}					0.237		0.11				
C_{16}					0.208		0.086				
C_{17}					0.220		0.078				
C_{18}					0.169		0.068				
C_{19}					0.140		0.050				
C_{20}					0.833		0.284				

† Pedersen, Thomassen and Fredenslund (1988)
‡ Pedersen, Thomassen and Fredenslund (1989)
♯ Aasberg-Petersen and Stenby (1991)
♮ Montel (1994)
♭ Hoffman et al. (1953)

Table F.3a Measured molecular weight and specific gravity of the heavy fractions

	GC1 †		GC2 ‡		GC3 †		GC4 ♯		GC5 ‡	
Component	MW	SG	MW	SG	MW	SG	MW	SG	MW	SG
C_7	91.0	0.820	100.0	0.745	95.0	0.7243	162.0	0.7957	95.0	0.726
C_8	105.0	0.777	114.0	0.753	103.0	0.7476			106.0	0.747
C_9	120.0	0.803	128.0	0.773	116.0	0.7764			116.0	0.769
C_{10}	222.0	0.853	179.0	0.814	167.0	0.8120			133.0	0.781
C_{11}									152.0	0.778
C_{12}									164.0	0.785
C_{13}									179.0	0.802
C_{14}									193.0	0.815
C_{15}									209.0	0.817
C_{16}									218.0	0.824
C_{17}									239.0	0.825
C_{18}									250.0	0.831
C_{19}									264.0	0.841
C_{20}									377.0	0.873

† Pedersen, Thomassen and Fredenslund (1988)
‡ Pedersen, Thomassen and Fredenslund (1989)
♯ Aasberg-Petersen abd Stenby (1991)

Table F.3b Measured molecular weight and specific gravity of the heavy fractions

	GC6 †		GC7 ‡		OIL1 ♭		OIL2 ‡			
Component	MW	SG	MW	SG	MW	SG	MW	SG	MW	SG
C_7	95.0	0.7325	96.0	0.741	99.0	0.749	91.68	0.745		
C_8	107.0	0.7561	107.0	0.780	110.0	0.758	104.99	0.767		
C_9	121.0	0.7722	121.0	0.807	121.0	0.779	117.60	0.798		
C_{10}	241.0	0.8604	134.0	0.819	221.0	0.852	293.53	0.902		
C_{11}			147.0	0.810						
C_{12}			161.0	0.828						
C_{13}			175.0	0.849						
C_{14}			190.0	0.857						
C_{15}			206.0	0.868						
C_{16}			222.0	0.872						
C_{17}			237.0	0.859						
C_{18}			251.0	0.854						
C_{19}			263.0	0.866						
C_{20}			339.0	0.873						

† Pedersen, Thomassen and Fredenslund (1988)
‡ Pedersen, Thomassen and Fredenslund (1989)
♭ Hoffman et al. (1953)

The Joule-Thomson Program

De Joule-Thomson program is written in HP FORTRAN/9000 and was run on a Hewlett Packard workstation. De source code of the program is called *jt.f*. To run the program it must be compiled and connected to the Michelsen flash routines. This is done using the following command:

f77 jt.f mfconnect.f mfcalc.f mfthermo.f mfoutput.f

The listing of jt.f is given in appendix G2. The listings of the Michelsen flash routines are given in the appendices to the report of Loggers[1] (1991). The parts of the Michelsen flash *mf.f* and *mfinput.f* used in Loggers work, were replaced by the routine *mfconnect.f*, which connects the Joule-Thomson program with the Michelsen program. The listing of mfconnect.f is given in appendix G3.
The Joule-Thomson program can perform the following calculations:

- Calculation of the inversion curve for pure components and multi-component mixtures. Three cubic equations of state are available for calculation purposes: the *Redlich-Kwong*, the *Soave-Redlich-Kwong*, and the *Peng-Robinson* equations. Inversion curves can also be calculated using the Gunn-Chueh-Prausnitz correlation. For single components it is possible obtain an inversion curve in terms of reduced or normal properties.

- Calculation of Joule-Thomson coefficients for pure components and multi-component mixtures. Calculations can be performed using all three equations of state, but calculations for multi-component mixtures are not possible using the Redlich-Kwong equation because the Michelsen flash routines do not support this equation and the phase stability can therefore not be checked. The Joule-Thomson coefficient can either be calculated for a single temperature or over a range of temperatures.

- Calculation of enthalpies and simulation of the isenthalpic expansion for pure components and multi-component mixtures. Again the Redlich-Kwong equation cannot be used for multi-component systems. Enthalpies can be calculated for a single temperature or a range of temperatures. For the isenthalpic expansion the initial pressure and temperature must be specified, as well as the final pressure. The pressure is reduced in steps to calculate the whole isenthalpic line.

Input files

To run the program, one must input critical properties T_c and P_c, molecular weight MW,

---

[1] Loggers, L.J., **In Search of Liquid-Liquid-Vapour Equilibria and Tricriticality in Binary Butane/n-Paraffin Systems and CO_2/n-Carboxylic Acid Systems**, *Master Thesis*, Laboratory of Applied Thermodynamics and Phase Equilibria, Delft University of Technology, June 1991.

acentric factors ω, and heat capacity coefficients $C_{p,1}$-$C_{p,4}$. Depending on the problem to be solved, the following input files are available:

Component.Dat: to be used for pure component calculations and multi-component mixtures. If one uses the component option, the Joule-Thomson program will ask how many components and which components are used. Binary interaction parameters must be input by hand and ,therefore, if one wants to repeat calculations for the same system it is recommended to make an own input file with all the necessary commands and run the problem using the following command: a.out < input file. Component.Dat is listed in appendix G4.

Gas_Condensate.Dat: to be used for calculations of gas-condensates. In the input file the number of components and heavy fractions must be specified, as well as the molecular weight and the specific gravity of the heavy fractions. An example of this input file is given in appendix G5.

Lumping.Dat: to be used to lump gas-condensates to reduce the number of components. In addition to the information needed in Gas-Condensates, the first and last component of each group must be given. An example is given in appendix G6.

Output file

After calculations are completed, the program will ask if the results should be saved. The output file can be given any name be typing it in.

Running the program

Running the program is very easy because at each step you will be asked to input the necessary information. The only way to find out now is by actually running it to see for yourself.

```
$INIT ON

      PROGRAM JOULE_THOMSON_EXPANSION

C ----------------------------------------------------------------------
C          JOULE-THOMSON EXPANSION PROGRAM
C
C   Author: Walter G. Kortekaas
C   Date  : Novermber 23, 1994
C   Place : Laboratory of Thermodynamics and Phase Equilibria
C           Faculaty of Chemical Engineering and Materials Science
C           Delft University of Technology
C
C   This program can calculate inversion curves, Joule-Thomson coefficients
C   , and temperature changes that occur at isenthalpic expansion of pure
C   components, gas-mixtures and gas-condensates
C
C ------------------------------------------------------ 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      CHARACTER ANSWER,EOSANW

      COMMON/EOS/OMEGA(2),IEOSTYPE

      WRITE (6,*) 'WELCOME TO THE JOULE-THOMSON PROGRAM'
      WRITE (6,*)
   10 WRITE (6,*) 'Which equation of state would you like to use?'
      WRITE (6,*) 'Soave-Redlich-Kwong (S)'
      WRITE (6,*) 'Peng-Robonson (P)'
      WRITE (6,*) 'Redlich-Kwong (R)'
      WRITE (6,*) 'GCP-correlation (G)   {Only for inversion curve!}'
      READ  (5,*) EOSANW
      IF (EOSANW .EQ. 'S' .OR. EOSANW .EQ. 's' ) THEN
        IEOSTYPE=0
      ELSE IF (EOSANW .EQ. 'P' .OR. EOSANW .EQ. 'p') THEN
        IEOSTYPE=1
      ELSE IF (EOSANW .EQ. 'R' .OR. EOSANW .EQ. 'r') THEN
        IEOSTYPE=2
      ELSE IF (EOSANW.EQ.'G' .OR. EOSANW.EQ.'g') THEN
        IEOSTYPE=3
      ELSE
        WRITE (6,*) 'wrong input, try again!'
        GOTO 10
      END IF
      CALL CONS
      WRITE (6,*)
   50 WRITE (6,*) 'Choose one of the following possibilities'
      WRITE (6,*)
      WRITE (6,*) '(A)  Calculation of inversion curve'
      WRITE (6,*) '(B)  Calculation of Joule-Thomson coefficient'
      WRITE (6,*) '(C)  Calculation of Isenthalpic expansion'
      READ (5,*) ANSWER
      IF (ANSWER .EQ. 'A' .OR. ANSWER .EQ. 'a') THEN
        CALL INVERSIONCURVES
      ELSE IF (ANSWER .EQ. 'B' .OR. ANSWER .EQ. 'b') THEN
        CALL JT_COEFFICIENT
      ELSE IF (ANSWER .EQ. 'C' .OR. ANSWER .EQ. 'c') THEN
        CALL JT_EXPANSION
      ELSE
        WRITE (6,*) 'wrong input, try again!'
        GOTO 50
      END IF
      WRITE (6,*) 'Do you want to continue calculations (Y/N)'
      READ (5,*) ANSWER
      IF (ANSWER .EQ. 'Y' .OR. ANSWER .EQ. 'y') THEN
        GOTO 10
      ELSE
        WRITE (6,*) 'Bye,bye!!'
      ENDIF
      STOP
   99 END


      SUBROUTINE JT_COEFFICIENT

C ----------------------------------------------------------------------
C
C     This subroutine calculates the Joule-Thomson coefficient for pure
C     components and gas-mixtures using the RK, SRK, and the PR EoS
```

```
C
C ------------------------------------------------------ 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION MU,JTCF,JTCFG,JTCFL,KIJ,LF
      CHARACTER ANSWER
      CHARACTER*15 OUTFILE

      DIMENSION X(15),Y(15),ZFACT(5),ZFRAC(15,5)
      DIMENSION MU(101,4)

      COMMON/XPROP/KIJ(15,15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON/FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/EXPANSION/TSAT,HINIT

      WRITE (6,*) 'THIS IS THE JOULE-THOMSON COEFFICIENT SUBROUTINE'
      WRITE (6,*)
      WRITE (6,*) 'How many components would you like to use?'
      READ *, NCOMP
      CALL COMPONENTS
      WRITE (6,*)
      WRITE (6,*) 'Give pressure (bar)'
      READ *, P
      P=P*1.D5
      TSAT=0.0D0
      IF (NCOMP.EQ.1) THEN
c --- Calculation of Saturation Properties -------------------------
         IF (P.LT.PC(1)) THEN
             CALL SATTEMPERATURE(TSAT,P,HSATVAP,HSATLIQ,XX)
             CALL JT_COEFF_CALC (XX,TSAT,P,JTCFG,VG,CPG,-1)
             CALL JT_COEFF_CALC (XX,TSAT,P,JTCFL,VL,CPL,1)
             WRITE (6,*)
             WRITE (6,*) 'JT coefficient Saturated liquid = ',JTCFL
             WRITE (6,*) 'JT coefficient Saturated vapour = ',JTCFG
             WRITE (6,*)
          ENDIF
c ------------------------------------------------------------------
   10     WRITE (6,*) 'Single Temperature (S) or Temperature Range (R)'
          READ *,ANSWER
          IF ((ANSWER .EQ. 'S') .OR. (ANSWER .EQ. 's')) THEN
             WRITE (6,*) 'Give Temperature (K)'
             READ *,T
             MSLOPE=-1
             IF (T.LT.TSAT) MSLOPE=1
             CALL JT_COEFF_CALC (XX,T,P,JTCF,V,CP,MSLOPE)
             WRITE (6,*) 'T = ',T
             WRITE (6,*) 'P = ',P/1.D5
             WRITE (6,*) 'JTCF = ',JTCF
             WRITE (6,*) 'V = ',V
             WRITE (6,*) 'CP = ',CP
          ELSE IF ((ANSWER .EQ. 'R') .OR. (ANSWER .EQ. 'r')) THEN
             WRITE (6,*) 'A temperature range must be specifified'
             WRITE (6,*) 'Give starting temperature (K)'
             READ *,TIN
             WRITE (6,*) 'Give final temperature (K)'
             READ *,TEND
             NN=101
             T=TIN
             DO I=1,NN
                MSLOPE=-1
                IF (T.LT.TSAT) MSLOPE=1
                CALL JT_COEFF_CALC (XX,T,P,JTCF,V,CP,MSLOPE)
                MU(I,1)=T
                MU(I,2)=JTCF
                MU(I,3)=CP
                MU(I,4)=V
                T=TIN+I*(TEND-TIN)/NN
                WRITE (6,110) (MU(I,J),J=1,4)
             ENDDO
          ELSE
             WRITE (6,*) 'wrong input, try again!'
             GOTO 10
          END IF
C ---- Multi-Component Systems ---------------------------------------
       ELSE
         WRITE (6,*) 'Give temperature (K)'
```

```
         READ (5,*) T
         PMF=P/1.D5
         CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,T,PMF,TC,PC,OM,KIJ,
     &                       NOFAS,X,Y,VF,LF,INITFLAG,IPH,zfact,zfrac)
         IF (NOFAS.EQ.1) THEN
            WRITE (6,*) 'ONE PHASE SYSTEM'
            CALL JT_COEFF_CALC (XX,T,P,JTCF,V,CP,-1)
            WRITE (6,*) 'T = ',T
            WRITE (6,*) 'P = ',P/1.D5
            WRITE (6,*) 'JTCF = ',JTCF
            WRITE (6,*) 'V = ',V
            WRITE (6,*) 'CP = ',CP
         ELSE IF (NOFAS.EQ.2) THEN
            WRITE (6,*) 'TWO PHASE SYSTEM'
            WRITE (6,*)
            WRITE (6,*) 'LIQUD PHASE'
            WRITE (6,*)
            CALL JT_COEFF_CALC (X,T,P,JTCF,V,CP,1)
            WRITE (6,*) 'T = ',T
            WRITE (6,*) 'P = ',P/1.D5
            WRITE (6,*) 'JTCF = ',JTCF
            WRITE (6,*) 'V = ',V
            WRITE (6,*) 'CP = ',CP
            WRITE (6,*)
            WRITE (6,*) 'VAPOUR PHASE'
            WRITE (6,*)
            CALL JT_COEFF_CALC (Y,T,P,JTCF,V,CP,-1)
            WRITE (6,*) 'T = ',T
            WRITE (6,*) 'P = ',P/1.D5
            WRITE (6,*) 'JTCF = ',JTCF
            WRITE (6,*) 'V = ',V
            WRITE (6,*) 'CP = ',CP
            WRITE (6,*)
            DO J=1,NCOMP
               WRITE (6,150) COMPONENT(J),X(J),Y(J)
            ENDDO
            WRITE (6,*)
            WRITE (6,*) 'VAPOURFRACTION = ',VF
         ELSE
            WRITE (6,*) 'MORE THAN TWO PHASES'
         ENDIF
      ENDIF
      WRITE (6,*) 'Save data to file (Y/N)'
      READ *,ANSWER
      IF ((ANSWER .EQ. 'Y') .OR. (ANSWER .EQ. 'y')) THEN
        WRITE (6,*) 'Name of outputfile'
        READ (5,*) OUTFILE
        OPEN  (9,FILE=OUTFILE,STATUS='UNKNOWN')
        WRITE (9,FMT=100) ((MU(i,j),j=1,2),i=1,101)
        CLOSE (9)
      END IF
  100 FORMAT (F12.5,F12.5)
  110 FORMAT (F10.5,F12.5,F12.5,F12.8)
  150 FORMAT (A8,3X,F8.6,4X,F8.6)
      RETURN
  199 END


       SUBROUTINE INVERSIONCURVES

C ---------------------------------------------------------------------
C     This subroutine calculates the inversion curve for pure components
C     and gas-mixtures for the RK, the SRK, and the PR Eos and for the
C     Gunn-Chueh-Prausnitz Correlation (1966)
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      CHARACTER ANSWER
      CHARACTER*15 OUTFILE

      DIMENSION TP(100,2)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON /FEEDFRAC/XX(15),COMPONENT(15)

    5 WRITE (6,*) 'Components of own choice (C)'
      WRITE (6,*) 'Gas-Condensate (G)'
```

```
      WRITE (6,*) 'Lumping (L)'
      READ (5,*) ANSWER
      IF ((ANSWER .EQ. 'C') .OR. (ANSWER .EQ. 'c')) THEN
        WRITE (6,*) 'How many components would you like to use?'
        READ (5,*) NCOMP
        CALL COMPONENTS
      ELSE IF ((ANSWER .EQ. 'G') .OR. (ANSWER .EQ. 'g')) THEN
        CALL GAS_CONDENSATES
      ELSE IF (ANSWER.EQ.'L' .OR. ANSWER.EQ.'l') THEN
        CALL LUMPING
      ELSE
        WRITE (6,*) 'wrong input, try again!'
        GOTO 5
      END IF

      IF (IEOSTYPE.EQ.1) CALL INVCURVEPR (TP)
      IF (IEOSTYPE.EQ.0) CALL INVCURVESRK (TP)
      IF (IEOSTYPE.EQ.2) CALL INVCURVERK (TP)
      IF (IEOSTYPE.EQ.3) CALL GCP_CORRELATION (TP,TCM,PCM)

      IF (NCOMP.EQ.1) THEN
        WRITE (6,*) 'Do you want a reduced pressure curve (Y/N)'
        READ (5,*) ANSWER
        IF (ANSWER .EQ. 'Y') THEN
          IF (IEOSTYPE.NE.3) THEN
            DO I=1,100
              TP(I,1)=TP(I,1)/TC(1)
              TP(I,2)=TP(I,2)/PC(1)
            END DO
          END IF
        ELSE
          IF (IEOSTYPE.EQ.3) THEN
            DO I=1,100
              TP(I,1)=TP(I,1)*TCM
              TP(I,2)=TP(I,2)*PCM/1.D5
            ENDDO
          ELSE
            DO I=1,100
              TP(I,2)=TP(I,2)/1.D5
            ENDDO
          ENDIF
        ENDIF
      ELSE
        DO I=1,100
          TP(I,2)=TP(I,2)/1.D5
        END DO
      ENDIF
   20 FORMAT ('TR = ',F12.5,'  PR = ',F12.5,'  tt  ',F10.5)
      WRITE (6,20) ((TP(i,j),j=1,3),i=1,10)
      WRITE (6,*) 'Save data to file (Y/N)'
      READ (5,*) ANSWER
      IF ((ANSWER .EQ. 'Y').OR.(ANSWER .EQ. 'y')) THEN
        WRITE (6,*) 'Name of outputfile'
        READ (5,*) OUTFILE
        OPEN  (9,FILE=OUTFILE,STATUS='UNKNOWN')
        WRITE (9,FMT=90) ((TP(i,J),J=1,2),i=1,100)
        CLOSE (9)
      END IF
   90 FORMAT (F12.3,F15.3)
      RETURN
      END


      SUBROUTINE JT_EXPANSION

C -----------------------------------------------------------------------
C     This subroutine models the isenthalpic Joule-Thomson expansion
C     for pure components and gas-mixtures using the RK, the SRK, and
C     the PR EoS
C
C     !! Beware !! .... The Michelsen flach routines do not support the
C                       RK Eos and, therefore, the two-phase region for
C                       multi-component mixtures cannot be checked for
C                       this equation
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      CHARACTER ANSWER
```

```
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/EOS/OMEGA(2),IEOSTYPE

   10 WRITE (6,*) 'Enthalpy as a function of temperature (1)'
      WRITE (6,*) 'Calculate isenthalpic expansion (2)'
      READ (5,*) IEXP
      IF ((IEXP.NE.1) .AND. (IEXP.NE.2)) THEN
         WRITE (6,*) IEXP
         WRITE (6,*) 'wrong input, try again!'
         GOTO 10
      ENDIF
      WRITE (6,*)
   25 WRITE (6,*) 'Components of own choice (C)'
      WRITE (6,*) 'Gas-Condensate (G)'
      WRITE (6,*) 'Lumping (L)'
      READ (5,*) ANSWER
      IF ((ANSWER .EQ. 'C') .OR. (ANSWER .EQ. 'c')) THEN
         WRITE (6,*) 'How many components would you like to use?'
         READ (5,*) NCOMP
         CALL COMPONENTS
      ELSE IF ((ANSWER .EQ. 'G') .OR. (ANSWER .EQ. 'g')) THEN
         CALL GAS_CONDENSATES
      ELSE IF (ANSWER.EQ.'L' .OR. ANSWER.EQ.'l') THEN
         CALL LUMPING
      ELSE
         WRITE (6,*) 'wrong input, try again!'
         GOTO 25
      END IF
      IF (NCOMP.EQ.1) CALL EXP_SINGLE(IEXP)
      IF (NCOMP.GT.1) CALL EXP_MULTIPLE(IEXP)
      RETURN
   99 END


      SUBROUTINE EXP_SINGLE (IANW)

C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      CHARACTER ANSWER
      CHARACTER*15 OUTFILE

      DIMENSION EXP(0:100,4),ENTH(0:100,2)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON/FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/EXPANSION/TSAT,HINIT
      COMMON/INVERSION/DVDT

      PARAMETER (XACC=1.D-8)

      EXTERNAL NEWTON_ENTHALPY,NEWTON_FUGACITY

      TSAT=0.D0
      IF (IANW.EQ.1) THEN
         WRITE (6,*) 'Give pressure (bar)'
         READ (5,*) P
         P=P*1.D5
C --- Calculation of saturation temperature -------------------------
         IF (P.LT.PC(1)) THEN
            CALL SATTEMPERATURE(TSAT,P,HSATVAP,HSATLIQ,XX)
         ENDIF
C -------------------------------------------------------------------
   30     WRITE (6,*) 'Single Temperature (S) or Temperature Range (R)'
         READ (5,*)ANSWER
         IF ((ANSWER .EQ. 'S').OR.(ANSWER .EQ. 's')) THEN
            WRITE (6,*) 'Give Temperature'
            READ (5,*)T
            MSLOPE=1
            IF (T.GT.TSAT) MSLOPE=-1
            CALL ENTHALPY(MSLOPE,XX,T,P,H,IER,V,HDT,DUM)
            WRITE (6,*) 'T = ',T
            WRITE (6,*) 'P = ',P/1.D5
            WRITE (6,*) 'H = ',H
            WRITE (6,*) 'Z = ',P*V/(RK*T)
         ELSE IF ((ANSWER .EQ. 'R').OR.(ANSWER .EQ. 'r')) THEN
```

```
            NN=100
            WRITE (6,*) 'You must specify temperature range'
            WRITE (6,*) 'Give starting temperature (K)'
            READ (5,*)TINIT
            WRITE (6,*) 'Give final temperature (K)'
            READ (5,*)TFINAL
            T=TINIT
            DO I=0,NN
               T=TINIT+I*(TFINAL-TINIT)/NN
               MSLOPE=-1
               IF (T.LT.TSAT) MSLOPE=1
               CALL ENTHALPY(MSLOPE,XX,T,P,H,IER,V,HDT,DUM)
               ENTH(I,1)=T
               ENTH(I,2)=H
               WRITE (6,105) (ENTH(I,J),J=1,2)
            ENDDO
         END IF
       ELSE IF (IANW.EQ. 2) THEN
         WRITE (6,*) 'Give initial pressure (bar)'
         READ (5,*) PINIT
         WRITE (6,*) 'Give initial temperature (K)'
         READ (5,*) TINIT
         WRITE (6,*) 'Give final pressure (bar)'
         READ (5,*) PFINAL
         PINIT=PINIT*1.D5
         PFINAL=PFINAL*1.D5
         CALL ENTHALPY(1,XX,TINIT,PINIT,HINIT,IER,V,HDT,DUM)
         X1=TINIT-25.D0
         X2=TINIT+25.D0
         NN=100
         P=PINIT
         TSAT=0.D0
         DO I=0,NN
            P=PINIT+I*(PFINAL-PINIT)/NN
            VAPFRAC=0.D0
C --- Calculation of saturation temperature -------------------------
            IF (P.LT.PC(1)) THEN
               IF (DABS(PC(1)-P).LT.1.5D5) P=P-2.D5
               CALL SATTEMPERATURE(TSAT,P,HSATVAP,HSATLIQ)
               IF ((HINIT.GE.HSATLIQ).AND.(HINIT.LE.HSATVAP)) THEN
                   TF=TSAT
                   H=HINIT
                   VAPFRAC=(HINIT-HSATLIQ)/(HSATVAP-HSATLIQ)
                   WRITE (6,*) 'VAPOURFRACTION = ',VAPFRAC
                   GOTO 40
               ENDIF
            ENDIF
C ---------------------------------------------------------------------
            CALL RTSAFES(NEWTON_ENTHALPY,X1,X2,P,XACC,TF)
            IF (TF.LE.TSAT) THEN
               CALL ENTHALPY(1,XX,TF,P,H,IER,V,HDT,DUM)
            ELSE
               CALL ENTHALPY(-1,XX,TF,P,H,IER,V,HDT,DUM)
            ENDIF
   40        EXP(I,1)=P/1.D5
            EXP(I,2)=TF
            EXP(I,3)=H
            EXP(I,4)=VAPFRAC
            X1=TF-50.D0
            X2=TF+50.D0
            WRITE (6,100) I,(EXP(I,J),J=1,4)
         ENDDO
      END IF
      WRITE (6,*) 'Save data to file (Y/N)'
      READ (5,*) ANSWER
      IF ((ANSWER .EQ. 'Y').OR.(ANSWER .EQ. 'y')) THEN
         WRITE (6,*) 'Name of outputfile'
         READ (5,*) OUTFILE
         OPEN (9,FILE=OUTFILE,STATUS='UNKNOWN')
         IF (IANW .EQ. 1) THEN
            WRITE (9,FMT=110) ((ENTH(I,J),J=1,2),I=0,100)
         ELSE
            WRITE (9,FMT=120) ((EXP(I,J),J=1,2),I=0,100)
         ENDIF
         CLOSE (9)
      END IF
  100 FORMAT (I3,2X,'P = ',F12.5,' T = ',F12.7,' H = ',F11.4,
     &       ' V = ',F11.9)
  105 FORMAT (' T = ',F12.7,' H = ',F12.5)
  110 FORMAT (F12.4,F12.4)
```

```
  120 FORMAT (F12.5,F12.7)
      RETURN
  199 END


      SUBROUTINE EXP_MULTIPLE(IANW)

C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION KIJ,LF,JTCF
      CHARACTER ANSWER,ASTAB
      CHARACTER*15 OUTFILE

      DIMENSION EXP(0:100,4),X(15),Y(15),XOLD(15),YOLD(15),ENTH(101,2)
      dimension zfact(5),zfrac(15,5),xmf(15),ymf(15)

      COMMON/XPROP/KIJ(15,15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON/FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/EXPANSION/TSAT,HINIT

      PARAMETER (XACC=1.D-8)

      EXTERNAL NEWTON_ENTHALPY,NEWTON_FUGACITY

      INITFLAG=0
      IPH=0

      IF (IANW .EQ. 1) THEN
         WRITE (6,*) 'Give pressure (bar)'
         READ (5,*) P
         P=P*1.D5
   30     WRITE (6,*) 'Single Temperature (S) or Temperature Range (R)'
         READ (5,*) ANSWER
         IF ((ANSWER .EQ. 'S').OR.(ANSWER .EQ. 's')) THEN
            WRITE (6,*) 'Give Temperature'
            READ (5,*) T
            PMF=P/1.D5
            CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,T,PMF,TC,PC,OM,KIJ,
     &                  NOFAS,X,Y,VF,LF,INITFLAG,IPH,zfact,zfrac)
            IF (NOFAS.EQ.1) THEN
               WRITE (6,*) 'ONE PHASE SYSTEM'
               CALL ENTHALPY(-1,XX,T,P,H,IER,V,HDT,DUM)
            ELSE IF (NOFAS.EQ.2) THEN
               WRITE (6,*) 'TWO PHASE SYSTEM'
               CALL ENTHALPY(1,X,T,P,HL,IER,VL,DUMMY,DUM)
               CALL ENTHALPY(-1,Y,T,P,HG,IER,VG,DUMMY,DUM)
               H=VF*HG+(1.D0-VF)*HL
               WRITE (6,*)
               WRITE (6,155)
               WRITE (6,*)
               DO J=1,NCOMP
                  WRITE (6,150) COMPONENT(J),X(J),Y(J)
               ENDDO
               WRITE (6,*)
               WRITE (6,130) HG,MAX(ZFACT(1),ZFACT(2))
               WRITE (6,140) HL,MIN(ZFACT(1),ZFACT(2))
               WRITE (6,*)
               WRITE (6,*) 'VAPOURFRACTION = ',VF
            ELSE
               WRITE (6,*) 'MORE THAN TWO PHASES'
            ENDIF
            WRITE (6,*)
            WRITE (6,*) 'T = ',T
            WRITE (6,*) 'P = ',P/1.d5
            WRITE (6,*) 'H = ',H
         ELSE IF ((ANSWER .EQ. 'R').OR.(ANSWER .EQ. 'r')) THEN
            NN=100
            WRITE (6,*) 'You must specify temperature range'
            WRITE (6,*) 'Give starting temperature (K)'
            READ (5,*) TINIT
            WRITE (6,*) 'Give final temperature (K)'
            READ (5,*) TFINAL
            T=TINIT
            PMF=P/1.D5
            DO I=0,NN
```

```
            CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,T,PMF,TC,PC,OM,
     &                  KIJ,NOFAS,X,Y,VF,LF,INITFLAG,IPH,ZFACT,ZFRAC)
            IF (NOFAS.EQ.1) THEN
c               WRITE (6,*) 'ONE PHASE'
               CALL ENTHALPY(-1,XX,T,P,H,IER,V,HDT,DUM)
            ELSE IF (NOFAS.EQ.2) THEN
c               WRITE (6,*) 'TWO PHASES'
               CALL ENTHALPY(1,X,T,P,HL,IER,VL,DUMMY,DUM)
               CALL ENTHALPY(-1,Y,T,P,HG,IER,VG,DUMMY,DUM)
               H=VF*HG+(1.D0-VF)*HL
            ELSE
               WRITE (6,*) 'MORE THAN TWO PHASES'
            ENDIF
            ENTH(I,1)=T
            ENTH(I,2)=H
            T=TINIT+I*(TFINAL-TINIT)/NN
            WRITE (6,105) (ENTH(I,J),J=1,2)
         ENDDO
      END IF
   ELSE IF (IANW .EQ. 2) THEN
      WRITE (6,*) 'Give initial pressure (bar)'
      READ (5,*) PINIT
      WRITE (6,*) 'Give initial temperature (K)'
      READ (5,*) TINIT
      WRITE (6,*) 'Give final pressure (bar)'
      READ (5,*) PFINAL
      WRITE (6,*) 'Use stability test (Y/N)'
      READ (5,*) ASTAB
      ISTAB=0
      IF (ASTAB.EQ.'N' .OR. ASTAB.EQ.'n') ISTAB=1
      PINIT=PINIT*1.D5
      PFINAL=PFINAL*1.D5
      PMFI=PINIT/1.D5
      CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,TINIT,PMFI,TC,PC,
     &          OM,KIJ,NOFAS,X,Y,VF,LF,INITFLAG,IPH,ZFACT,ZFRAC)
      IF (NOFAS.EQ.1) THEN
         CALL ENTHALPY(-1,XX,TINIT,PINIT,HINIT,IER,V,HDT,DUM)
      ELSE IF (NOFAS.EQ.2) THEN
         CALL ENTHALPY(1,X,TINIT,PINIT,HL,IER,VL,DUMMY,DUM)
         CALL ENTHALPY(-1,Y,TINIT,PINIT,HG,IER,VG,DUMMY,DUM)
         HINIT=VF*HG+(1.D0-VF)*HL
         DO J=1,NCOMP
            XOLD(J)=X(J)
            YOLD(J)=Y(J)
         ENDDO
         VFOLD=VF
         POLD=PINIT
         TOLD=TINIT
      ELSE
         WRITE (6,*) 'MORE THAN TWO PHASES'
         STOP
      ENDIF
      EXP(0,1)=PINIT/1.D5
      EXP(0,2)=TINIT
      EXP(0,3)=HINIT
      EXP(0,4)=VF
      NN=100
      IMF=0
      P=PINIT
      TF=TINIT
      DO I=0,NN
         P=PINIT+I*(PFINAL-PINIT)/NN
         IF (I.EQ.0) GOTO 75
C ---- perform stability test at P and initial T ---------------------
         IF (IMF.EQ.1 .AND. ISTAB.EQ.1) GOTO 40
         WRITE (6,*) 'checking stability initial state'
         IF (NOFAS .EQ. 1) THEN
            CALL JT_COEFF_CALC (XX,TOLD,P,JTCF,V,CP,MSLOPE)
         ELSE
            CALL JT_COEFF_CALC (Y,TOLD,P,JTCF,V,CP,MSLOPE)
         ENDIF
         TF=TF+(P-POLD)/1.D5*JTCF
c        WRITE (6,*) 'initial T',TF
         PMF=P/1.D5
         CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,TF,PMF,TC,PC,
     &          OM,KIJ,NOFAS,XMF,YMF,VF,LF,INITFLAG,IPH,zfact,zfrac)
         WRITE (6,*) 'INITIAL STATE IS ',NOFAS,' PHASE SYSTEM'
C -------------------------------------------------------------------
         IF (NOFAS.EQ.1) THEN
            X1=TF-10.D0
```

```
                X2=TF+10.D0
                CALL RTSAFES(NEWTON_ENTHALPY,X1,X2,P,XACC,TF)
                CALL ENTHALPY(-1,XX,TF,P,H,IER,V,HDT,DUM)
C ---- perform stability test at P and final T for single phase ------
                CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,TF,PMF,TC,PC,
     &               OM,KIJ,NOFAS,Xmf,Ymf,VF,LF,INITFLAG,IPH,zfact,zfrac)
                IF (NOFAS.EQ.1) THEN
                   WRITE (6,*) 'FINAL STATE STABLE'
                   GOTO 70
                ENDIF
                WRITE (6,*) 'FINAL STATE INSTABLE'
C ---------------------------------------------------------------
             ENDIF
   40         IF (NOFAS.EQ.2 .OR. IMF.EQ.1) THEN
                A=VFOLD
                if (imf.LE.1) then
                   a=vf
                   do j=1,ncomp
                      x(j)=xmf(j)
                      y(j)=ymf(j)
                   enddo
                endif
                IMF=1
   50            CALL FLASH(2,VFOLD,XOLD,YOLD,TOLD,POLD,HINIT,A,X,Y,TF,P,
     &                     IERF,VL,VG,H)
                WRITE (6,*)
                WRITE (6,*) 'FRACTIONS'
                DO J=1,NCOMP
                   WRITE (6,150) COMPONENT(J),X(J),Y(J)
                ENDDO
                IF (ISTAB.EQ.1) THEN
                   VF=A
                   GOTO 70
                ENDIF
C ---- perform stability test at P and final T for two phases ---------
                WRITE (6,*) 'checking stability final stage'
                CALL MICHELSENFLASH(IEOSTYPE,NCOMP,XX,TF,PMF,TC,PC,
     &               OM,KIJ,NOFAS,Xmf,Ymf,VF,LF,INITFLAG,IPH,zfact,zfrac)
                VF=A
                IF (NOFAS.EQ.2) THEN
                   WRITE (6,*) NOFAS,' PHASE SYSTEM STABLE'
                   GOTO 70
                ENDIF
                IF (NOFAS.EQ.1) THEN
                   WRITE (6,*) '1 PHASE SYSTEM'
                   GOTO 70
                ENDIF
                WRITE (6,*) NOFAS, 'PHASE SYSTEM'
                WRITE (6,*) 'COMPRESSIBILTY',(ZFACT(J),J=1,NOFAS)
                DO II=1,NCOMP
                   WRITE (6,*) II,(' ',ZFRAC(II,J),J=1,3)
                ENDDO
                STOP
C ---------------------------------------------------------------
             ENDIF
   70         DO J=1,NCOMP
                XOLD(J)=X(J)
                YOLD(J)=Y(J)
              ENDDO
   75         VFOLD=VF
              POLD=P
              TOLD=TF
              EXP(I,1)=P/1.D5
              EXP(I,2)=TF
              EXP(I,3)=H
              EXP(I,4)=VF
              WRITE (6,100) I,(EXP(I,J),J=1,4)
           ENDDO
        END IF
        WRITE (6,*) 'Save data to file (Y/N)'
        READ (5,*)ANSWER
        IF ((ANSWER .EQ. 'Y').OR.(ANSWER .EQ. 'y')) THEN
          WRITE (6,*) 'Name of outputfile'
          READ (5,*) OUTFILE
          OPEN  (9,FILE=OUTFILE,STATUS='UNKNOWN')
          IF (IANW .EQ. 1) THEN
              WRITE (9,FMT=80) ((ENTH(I,J),J=1,2),i=1,NN)
          ELSE
              WRITE (9,FMT=90) ((EXP(I,J),J=1,3),I=0,NN)
          ENDIF
```

```
        CLOSE (9)
      ELSE
        GOTO 99
      END IF
   80 FORMAT (F12.5,F12.4)
   90 FORMAT (F10.5,F12.5,F12.4)
  100 FORMAT (I3,2X,'P = ',F10.5,' T = ',F12.7,' H = ',F12.5,
     & ' V = ',F12.9)
  105 FORMAT (' T = ',F12.7,' H = ',F12.5)
  110 FORMAT (F12.5,F12.4)
  120 FORMAT (F12.5,F12.7,F12.5,F12.9)
  130 FORMAT ('ENTHALPY VAPOR IS  ',F10.3,5X,' Z IS ',F6.4)
  140 FORMAT ('ENTHALPY LIQUID IS ',F10.3,5X,' Z IS ',F6.4)
  150 FORMAT (A8,3X,F8.6,4X,F8.6)
  155 FORMAT ('COMPONENT',3X,'LIQUID',6X,'VAPOR')
      RETURN
  199 END


      SUBROUTINE JT_COEFF_CALC (XX,T,P,JTCF,V,CP,MSLOPE)

C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I,N)

      DOUBLE PRECISION JTCF

      DIMENSION VV(10),XX(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/HEATCPCOEFF/CPC(15,4)
      COMMON/EOS/OMEGA(2),IEOSTYPE

      CALL VOLUME (XX,T,P,MSLOPE,V,DUMMY)

      CPID=0.D0
      DO I=1,NCOMP
          U=XX(I)*(CPC(I,1)+CPC(I,2)*T+CPC(I,3)*T**2+CPC(I,4)*T**3)
          CPID=CPID+U
      END DO

      IF (IEOSTYPE.EQ.0) THEN
         VV(1)=V-BM
         VV(2)=V+BM
         VV(3)=AM(2)/V
         VV(4)=RK/VV(1)
         VV(5)=AM(1)/V
         VV(6)=VV(4)-VV(3)/VV(2)
         VV(7)=T*VV(6)/(-T*VV(4)/VV(1)+VV(5)/(V*VV(2))+VV(5)/VV(2)**2)
         C=AM(3)/BM
         CP=CPID-VV(6)*VV(7)+C*T*DLOG(VV(2)/V)-RK
         JTCF=-((VV(7)+V)/CP)*1.D5
      ELSE IF (IEOSTYPE.EQ.1) THEN
         VV(1)=V-BM
         VV(2)=V+BM
         VV(3)=AM(2)
         VV(4)=RK/VV(1)
         VV(5)=VV(3)/(V*VV(2)+BM*VV(1))
         VV(6)=AM(1)/(V*VV(2)+BM*VV(1))**2
         VV(7)=VV(4)-VV(5)
         VV(8)=T*VV(7)/(-VV(4)*T/VV(1)+2.0D0*VV(6)*VV(2))
         VV(9)=(V+(DSQRT(2.D0)+1.D0)*BM)/(V-(DSQRT(2.D0)-1.D0)*BM)
         C=DSQRT(2.D0)/(4.0D0*BM)*AM(3)
         CP=CPID-VV(7)*VV(8)+C*T*DLOG(VV(9))-RK
         JTCF=-((VV(8)+V)/CP)*1.D5
      ELSE IF (IEOSTYPE.EQ.2) THEN
         VV(1)=V-BM
         VV(2)=V+BM
         VV(3)=RK/VV(1)
         VV(4)=AM(1)/(T*V*VV(2))
         VV(5)=VV(3)+VV(4)/2.0D0
         VV(6)=VV(5)/(-VV(3)/VV(1)+VV(4)/VV(2)+VV(4)/V)
         C=0.75D0*AM(1)/BM
         CP=CPID-VV(5)*VV(6)+DLOG(VV(2)/V)*C/T-RK
         JTCF=-((VV(6)+V)/CP)*1.D5
      ENDIF
      RETURN
  199 END
```

```
      SUBROUTINE INVCURVEPR (TP)

C ---------------------------------------------------------------------
C
C     This subroutine calculates the inversion curve for the PR EoS
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION M,KIJ

      DIMENSION TP(100,2),M(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON/FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)

      B=0.D0
      SUM1=0.D0
      SUM2=0.D0
      SUM3=0.D0
      N=NCOMP

      DO I=1,N
         M(I)=0.37464D0+1.54226D0*OM(I)-0.26992D0*OM(I)**2
      ENDDO
      DO I=1,N
         T=OMEGA(2)*RK*XX(I)*TC(I)/PC(I)
         B=B+T
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*TC(I)*TC(J)/(DSQRT(PC(I)*PC(J)))*
     &         (1.0D0+M(I))*(1.0D0+M(J))*(1.D0-KIJ(I,J))
            SUM1=SUM1+T
         ENDDO
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*(TC(J)*DSQRT(TC(I))*M(I)*
     &         (M(J)+1.0D0)+TC(I)*DSQRT(TC(J))*M(J)*
     &         (M(I)+1.0D0))/(DSQRT(PC(I)*PC(J)))*(1.D0-KIJ(I,J))
            SUM2=SUM2+T
         ENDDO
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*M(I)*M(J)*DSQRT(TC(I)*TC(J)/((PC(I)*PC(J))))*
     &         (1.D0-KIJ(I,J))
            SUM3=SUM3+T
         ENDDO
      ENDDO

      X = 1.3D0
      DO I=1,100
         AA=(OMEGA(1)*RK**2*SUM3*(X-1.0D0)**2*(X**2+1.0D0)-
     &       RK*B*(X**2+2.0D0*X-1.0D0)**2)
         BB=(-OMEGA(1)*RK**2*SUM2*(X-1.0D0)**2*(1.5D0*X**2+X+0.5D0))
         CC=(2.0D0*OMEGA(1)*RK**2*SUM1*X*(X+1.0D0)*(X-1.0D0)**2)
         TR=((-BB-DSQRT(BB**2-4.0D0*AA*CC))/(2.0D0*AA))**2
         ALPHA=OMEGA(1)*RK**2*(SUM1-DSQRT(TR)*SUM2+TR*SUM3)
         PR=RK*TR/(B*(X-1.D0))-(ALPHA/B**2)/(X**2+2.0D0*X-1.0D0)
         TP(I,1)=TR
         TP(I,2)=PR
         IF (X .LE. 2.0D0) THEN
            X=X+0.05D0
         ELSE IF (X .LE. 4.0D0) THEN
            X=X+0.1D0
         ELSE IF (X .LE. 15.0D0) THEN
            X=X+1.0D0
         ELSE
            X=X+10.0D0
         ENDIF
      ENDDO
      RETURN
      END
```

```
      SUBROUTINE INVCURVESRK (TP)

C ---------------------------------------------------------------------
C
C     This subroutine calculates the inversion curve for the SRK EoS
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION M,KIJ

      DIMENSION TP(100,2),M(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)

      B=0.D0
      SUM1=0.D0
      SUM2=0.D0
      SUM3=0.D0
      N=NCOMP

      DO I=1,N
         M(I)=0.480D0+1.574D0*OM(I)-0.176D0*OM(I)**2
      ENDDO
      DO I=1,N
         T=OMEGA(2)*RK*XX(I)*TC(I)/PC(I)
         B=B+T
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*TC(I)*TC(J)/(DSQRT(PC(I)*PC(J)))*
     &         (1.0D0+M(I))*(1.0D0+M(J))*(1.D0-KIJ(I,J))
            SUM1=SUM1+T
         ENDDO
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*(TC(J)*DSQRT(TC(I))*M(I)*
     &         (M(J)+1.0D0)+TC(I)*DSQRT(TC(J))*M(J)*
     &         (M(I)+1.0D0))/(DSQRT(PC(I)*PC(J)))*(1.D0-KIJ(I,J))
            SUM2=SUM2+T
         ENDDO
      ENDDO
      DO I=1,N
         DO J=1,N
            T=XX(I)*XX(J)*M(I)*M(J)*DSQRT(TC(I)*TC(J)/((PC(I)*PC(J))))*
     &         (1.D0-KIJ(I,J))
            SUM3=SUM3+T
         ENDDO
      ENDDO

      X = 1.3D0
      DO I=1,100
         AA=(-RK*B*X**2*(X+1.0D0)**2+OMEGA(1)*RK**2*X**2*(X-1.0D0)
     &       **2*SUM3)
         BB=(-OMEGA(1)*RK**2*SUM2*X*(X-1.0D0)**2*(1.5D0*X+0.5D0))
         CC=(OMEGA(1)*RK**2*SUM1*X*(2.0D0*X+1.0D0)*(X-1.0D0)**2)
         TR=((-BB-DSQRT(BB**2-4.0D0*AA*CC))/(2.0D0*AA))**2
         ALPHA=OMEGA(1)*RK**2*(SUM1-DSQRT(TR)*SUM2+TR*SUM3)
         PR=RK*TR/(B*(X-1.D0))-(ALPHA/B**2)/(X**2+X)
         TP(I,1)=TR
         TP(I,2)=PR
         IF (X .LE. 2.0D0) THEN
            X=X+0.05D0
         ELSE IF (X .LE. 4.0D0) THEN
            X=X+0.1D0
         ELSE IF (X .LE. 15.0D0) THEN
            X=X+1.D0
         ELSE
            X=X+10.D0
         ENDIF
      ENDDO
      RETURN
      END
```

```
      SUBROUTINE INVCURVERK (TP)

C ---------------------------------------------------------------------
C
C     This subroutine calculates the inversion curve for the PR EoS
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION KIJ

      DIMENSION TP(100,2)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)

      A=0.0D0
      B=0.0D0
      N=NCOMP

      DO I=1,N
         DO J=1,N
            T=OMEGA(1)*RK**2*XX(I)*XX(J)*DSQRT(TC(I)**(2.5D0)
     &           /PC(I)*TC(j)**(2.5D0)/PC(J))*(1.D0-KIJ(I,J))
            A=A+T
         ENDDO
      ENDDO
      DO I=1,N
         T=OMEGA(2)*RK*XX(I)*TC(I)/PC(I)
         B=B+T
      ENDDO

      X = 1.3D0
      DO I=1,100
         AA=(-B*RK*(X**2+X)**2)
         BB=(A*X*(X-1.D0)**2*(2.5D0*X+1.5D0))
         TR=(-BB/AA)**(2.0D0/3.0D0)
         PR=TR*RK/(B*(X-1.D0))-(A/(DSQRT(TR)*B**2))/(X**2+X)
         TP(I,1)=TR
         TP(I,2)=PR
         IF (X .LE. 2.0D0) THEN
            X=X+0.05D0
         ELSE IF (X .LE. 4.0D0) THEN
            X=X+0.1D0
         ELSE IF (X .LE. 15.0D0) THEN
            X=X+1.D0
         ELSE
            X=X+10.D0
         END IF
      END DO
      RETURN
      END


      SUBROUTINE GCP_CORRELATION (TP,TCM,PCM)

C ---------------------------------------------------------------------
C
C     This subroutine calculates the inversion curve for the GCP correlation
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DIMENSION TP(100,2)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON /FEEDFRAC/XX(15),COMPONENT(15)

      TCM=0.D0
      PCM=0.D0
      N=NCOMP

      DO I=1,N
         T=XX(I)*TC(I)
```

```
        TCM=TCM+T
      ENDDO

      DO I=1,N
        T=XX(I)*PC(I)
        PCM=PCM+T
      ENDDO

      TR=0.8D0
      DO I=1,100
        PR=-36.275D0+71.598D0*TR-41.567D0*TR**2+11.826D0*TR**3
     +      -1.6721D0*TR**4+0.091167D0*TR**5
        TP(I,1)=TR
        TP(I,2)=PR
        TR=TR+0.05D0
      ENDDO
      RETURN
      END


      SUBROUTINE SATTEMPERATURE(TSAT,P,HSATVAP,HSATLIQ)

C ------------------------------------------------------ 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/PURE/TC(15),PC(15),OM(15)

      PARAMETER (XACC=1.D-8)

      EXTERNAL NEWTON_ENTHALPY,NEWTON_FUGACITY

C --- Calculation of saturation temperature for pure components --------
      X1=TC(1)
      X2=50.D0
  10  CALL VOLUME (XX,X1,P,0,V,IROOTNUMBER)
      IF (IROOTNUMBER.NE.3) THEN
        X1=X1-.05D0
        GOTO 10
      ENDIF
  20  CALL VOLUME (XX,X2,P,0,V,IROOTNUMBER)
      IF (IROOTNUMBER.NE.3) THEN
        X2=X2+.05D0
        GOTO 20
      ENDIF
   50 CALL RTSAFES(NEWTON_FUGACITY,X1,X2,P,XACC,TSAT)
      CALL ENTHALPY(1,XX,TSAT,P,HSATLIQ,IER,VSATLIQ,DUMMY,DUM)
      CALL ENTHALPY(-1,XX,TSAT,P,HSATVAP,IER,VSATVAP,DUMMY,DUM)
      CALL FUGACITY(1,XX,TSAT,VSATLIQ,P,FUGLIQ)
      CALL FUGACITY(1,XX,TSAT,VSATVAP,P,FUGGAS)
      WRITE (6,100) TSAT
      WRITE (6,*)
      WRITE (6,110) FUGGAS,FUGLIQ
      WRITE (6,*)
      WRITE (6,120) HSATLIQ,VSATLIQ
      WRITE (6,130) HSATVAP,VSATVAP
      WRITE (6,*)
  100 FORMAT ('TSAT = ',F12.5,' K')
  110 FORMAT ('FUGGAS = ',F12.9,' FUGLIQ = ',F12.9)
  120 FORMAT ('HSATLIQ = ',F14.6,' J/mole',5X,'V = ',F12.9,' m3/mole')
  130 FORMAT ('HSATVAP = ',F14.6,' J/mole',5X,'V = ',F12.9,' m3/mole')
      RETURN
  199 END


      SUBROUTINE NEWTON_FUGACITY (T,P,DFUG,DFUGDT)

C ------------------------------------------------------ 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP

      CALL ENTHALPY(1,XX,T,P,HL,IER,VL,DUMMY,HRESL)
      CALL ENTHALPY(-1,XX,T,P,HG,IER,VG,DUMMY,HRESG)
      CALL VOLUME (XX,T,P,1,VL,DUMMY)
      CALL FUGACITY (1,XX,T,VL,P,FUGL,DFUGLDTBKP)
```

```
      CALL VOLUME (XX,T,P,-1,VG,DUMMY)
      CALL FUGACITY (1,XX,T,VG,P,FUGG,DFUGGDTBKP)

      DFUG=FUGL-FUGG

      DFUGLDT=-HRESL/(RK*T**2)
      DFUGGDT=-HRESG/(RK*T**2)
      DFUGDT=DFUGLDT-DFUGGDT
      RETURN
      END


      SUBROUTINE NEWTON_ENTHALPY (T,P,DH,DHDT)

C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON /EXPANSION/TSAT,HINIT

      MSLOPE=-1
      IF (T.LT.TSAT) MSLOPE=1
      CALL ENTHALPY(MSLOPE,XX,T,P,H,IER,V,HDT,DUM)
      DH=HINIT-H
      DHDT=-HDT
      RETURN
      END


      SUBROUTINE COMPONENTS

C ---------------------------------------------------------------------
C     This subroutine is used to input all the necessary data for the
C     components
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)

      DOUBLE PRECISION KIJ,MW(15)
      CHARACTER*8 COMP(15),COMPONENT

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/HEATCPCOEFF/CPC(15,4)

      N=NCOMP

      SUMMOL=0.0D0
      DO I=1,15
        DO J=1,15
          KIJ(I,J)=0.D0
        ENDDO
      ENDDO

      WRITE (6,*) '(1) Nitrogen'
      WRITE (6,*) '(2) Carbonmonoxide'
      WRITE (6,*) '(3) Carbondioxide'
      WRITE (6,*) '(4) Methane'
      WRITE (6,*) '(5) Ethane'
      WRITE (6,*) '(6) Propane'
      WRITE (6,*) '(7) N-Butane'
      WRITE (6,*) '(8) I-Butane'
      WRITE (6,*) '(9) N-Pentane'
      WRITE (6,*) '(10) I-Pentane'
      WRITE (6,*) '(11) N-Hexane'
      WRITE (6,*) '(12) N-Heptane'
      WRITE (6,*) '(13) N-Octane'
      WRITE (6,*) '(14) N-Decane'
      WRITE (6,*) '(15) N-Dodecane'
      WRITE (6,*) '(16) N-Pentadecane'
      WRITE (6,*) '(17) N-Eicosane (C20)'
      WRITE (6,*) '(18) N-Tetracosane (C24)'
      DO I=1,N
        WRITE (6,*) 'What is component no. ',I,' ?'
        READ (5,*)IANSWER
        IF (IANSWER .EQ. 1) COMP(I)='N2'
```

```
          IF (IANSWER .EQ. 2) COMP(I)='CO'
          IF (IANSWER .EQ. 3) COMP(I)='CO2'
          IF (IANSWER .EQ. 4) COMP(I)='C1'
          IF (IANSWER .EQ. 5) COMP(I)='C2'
          IF (IANSWER .EQ. 6) COMP(I)='C3'
          IF (IANSWER .EQ. 7) COMP(I)='NC4'
          IF (IANSWER .EQ. 8) COMP(I)='IC4'
          IF (IANSWER .EQ. 9) COMP(I)='NC5'
          IF (IANSWER .EQ. 10) COMP(I)='IC5'
          IF (IANSWER .EQ. 11) COMP(I)='NC6'
          IF (IANSWER .EQ. 12) COMP(I)='NC7'
          IF (IANSWER .EQ. 13) COMP(I)='NC8'
          IF (IANSWER .EQ. 14) COMP(I)='NC10'
          IF (IANSWER .EQ. 15) COMP(I)='NC12'
          IF (IANSWER .EQ. 16) COMP(I)='NC15'
          IF (IANSWER .EQ. 17) COMP(I)='NC20'
          IF (IANSWER .EQ. 18) COMP(I)='NC24'
       END DO

    1  SUMX=0.0
       IF (N .EQ. 1) THEN
          XX(N)=1.0D0
          SUMX=1.0
       ELSE
          DO I=1,N
             WRITE (6,*) 'Molefraction ',COMP(I)
             READ (5,*)XX(I)
             SUMX=SUMX+XX(I)
          END DO
       END IF
       DO I=1,N
          SUM=SUM+XX(I)
       ENDDO
       DO I=1,N
          XX(I)=XX(I)/SUM
       ENDDO

    10 OPEN (8,FILE='COMPONENT.DAT')
       DO I=1,N
          DO J=1,7
             READ(8,*)
          END DO
    20    READ (8,*) COMPONENT(I),MW(I),TC(I),PC(I),OM(I),CPC(I,1),
      &             CPC(I,2),CPC(I,3),CPC(I,4)
          IF (COMPONENT(I) .EQ. COMP(I)) THEN
             GOTO 30
          ELSE
             GOTO 20
          END IF
    30    CONTINUE
          REWIND 8
       END DO
       CLOSE (8)
       IF (N.EQ.1) GOTO 99
       WRITE (6,*) 'Give interaction parameters'
       write (6,*) n
       DO I=1,N
          READ (5,*) (KIJ(I,J),J=1,N)
          write (6,110) (kij(i,j),j=1,n)
       ENDDO
    99 DO I=1,N
          SUMMOL=SUMMOL+XX(I)*MW(I)
          WRITE (6,100) COMPONENT(I),XX(I),MW(I),TC(I),PC(I)/1.D5,OM(I)
       ENDDO
       WRITE (6,120) SUMMOL
   100 FORMAT (A,2X,F8.4,2X,F8.4,2X,F8.2,2X,F10.5,2X,F8.4)
   110 format (12f6.3)
   120 FORMAT ('Moleculair weight:',4X,F8.4,' g/mole')
   199 RETURN
       END

       SUBROUTINE GAS_CONDENSATES

C ---------------------------------------------------------------------
C      This subroutine is used to input all the necessary data for
C      gas-condesates
C
C ------------------------------------------------------- 23/11/1994 ---

       IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)
```

```
      DOUBLE PRECISION MW(15),KIJ,M
      DIMENSION SG(15)
      CHARACTER ANSWER
      CHARACTER*8 COMPONENT

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/HEATCPCOEFF/CPC(15,4)
      COMMON/EOS/OMEGA(2),IEOSTYPE

C --- Parameters for Pedersen-Thomassen-Fredenslund correlations --------
      PARAMETER(E1=1.6312D2,E2=8.6052D1,E3=4.3475D-1,E4=-1.8774D3)
      PARAMETER(E5=-1.3408D-1,E6=2.5019D0,E7=2.0846D2,E8=-3.9872D3)
      PARAMETER(E9=7.4310D-1,E10=4.8122D-3,E11=9.6707D-3,E12=-3.7184D-6)
C --- Parameters for Aasberg-Petersen correlations----------------------
      PARAMETER (D1=190.8D0,D2=87.70D0,D3=0.6387D0,D4=-5247.7D0)
      PARAMETER (D5=-0.5836D0,D6=2.3211D0,D7=282.2D0,D8=-4484.9D0)
      PARAMETER (D9=2.829D-2,D10=4.248D-3,D11=1.156D-2,D12=-4.069D-6)
C ------------------------------------------------------------------------

      SUMMOL=0.D0
      DO I=1,15
         DO J=1,15
            KIJ(I,J)=0.D0
         ENDDO
      ENDDO

   10 OPEN (8,FILE='GAS_CONDENSATE.DAT')
      DO I=1,6
         READ (8,*)
      ENDDO
      READ(8,*) NCOMP
      READ(8,*)
      READ(8,*) NPLUS
      DO I=1,3
         READ (8,*)
      ENDDO
      NN=NCOMP-NPLUS
      DO I=1,NN
         READ (8,*) COMPONENT(I),MW(I),TC(I),PC(I),OM(I),
     &          CPC(I,1),CPC(I,2),CPC(I,3),CPC(I,4)
      END DO
      DO I=1,9
         READ (8,*)
      ENDDO
      DO I=1,NCOMP
         IF (I .LE. NN) THEN
            READ (8,*) COMPONENT(I),XX(I)
         ELSE
            READ (8,*) COMPONENT(I),XX(I),MW(I),SG(I)
c ----- Relations from Aasberg-Petersen, Stengby (1991) ---------------
            TC(I)=D1*SG(I)+D2*DLOG(MW(I))+D3*MW(I)+D4/MW(I)
            PC(I)=D5+D6*SG(I)+D7/MW(I)+D8/MW(I)**2
            PC(I)=DEXP(PC(I))/1.01325D-5
            OM(I)=D9+D10*MW(I)+D11*SG(I)+D12*MW(I)**2
c ----- Relations from Pedersen, Thomassen & Fredenslund (1988) -------
c            TC(I)=E1*SG(I)+E2*DLOG(MW(I))+E3*MW(I)+E4/MW(I)
c            PC(I)=DEXP(E5+E6*SG(I)+E7/MW(I)+E8/MW(I)**2)/1.01325D-5
c            M=E9+E10*MW(I)+E11*SG(I)+E12*MW(I)**2
c            IF (IEOSTYPE.EQ.0) THEN
c               OM(I)=(-1.574D0+DSQRT(1.574D0**2+0.704D0*(0.48D0-M)))/
c     &                -0.352D0
c            ELSE
c               OM(I)=(-1.54226D0+DSQRT(1.54266D0**2+4.D0*0.26992D0*
c     &              (0.37464D0-M)))/(2.D0*-0.26992D0)
c            ENDIF
c ----- Katz & Firoozabadi (1978) -------------------------------------
c            TB=97.58D0*MW(I)**(0.3323D0)*SG(I)**(0.04609D0)
c ----- Cavett Relations (1964) ---------------------------------------
c            API=141.5D0/SG(I)-131.5D0
c            TB=TB*1.8D0-459.69D0
c            TC(I)=768.071D0+1.7134D0*TB-0.10834D-2*TB**2+0.3889D-6*
c     &            TB**3-0.89213D-2*TB*API+0.53095D-6*TB**2*API+
c     &            0.32712D-7*TB**2*API**2
c            PCLOG=2.829+0.9412D-3*TB-0.30475D-5*TB**2+0.15141D-8*
c     &            TB**3-0.20876D-4*TB*API+0.11048D-7*TB**2*API+
c     &            0.1395D-9*TB**2*API**2-0.4827D-7*TB*API**2
c            PC(I)=10.D0**PCLOG
```

```
c             TC(I)=TC(I)/1.8D0+255.37D0
c             TB=TB/1.8D0+255.37D0
c             PB=14.5037D0
c             PBR=PB/PC(I)
c             TBR=TB/TC(I)
c             PC(I)=PC(I)/14.5037D-5
c             WCF=4.5579D0*MW(I)**(0.15178D0)*SG(I)**(-0.84573D0)
C ---- Lee-Kesler relations (1976) ------------------------------------
c             IF (TBR.LT.0.8D0) THEN
c                OM(I)=(DLOG(PBR)-5.92714D0+6.09648D0/TBR+1.28862D0*DLOG
c     &               (TBR)-0.169347D0*TBR**6)/(15.2518D0-15.6875D0/TBR
c     &               -13.4721D0*DLOG(TBR)+0.43577D0*TBR**6)
c             ELSE
c                OM(I)=-7.904D0+0.1352D0*WCF-0.007465D0*WCF**2+8.359D0*
c     &               TBR+(1.408D0-0.01063D0*WCF)/TBR
c             ENDIF
c ----- Ideal Gas Heat Capacity Coefficients by Kesler & Lee (1976) ---
            WCF=4.5579D0*MW(I)**(0.15178D0)*SG(I)**(-0.84573D0)
            CF=((12.8D0-WCF)*(10.D0-WCF)/(10.D0*OM(I)))**2
            CPC(I,1)=(-0.33886D0+0.02827D0*WCF-0.26105D0*CF+0.59332D0*
     &           OM(I)*CF)*4.19D0*MW(I)
            CPC(I,2)=(-1.D-4*(.9291D0-1.1543D0*WCF+.0368*WCF**2)+CF*
     &           (4.56D0-9.48D0*OM(I))*1.D-4)*4.19D0*1.8D0*MW(I)
            CPC(I,3)=(-1.6658D-7+CF*(0.536D0-0.6828D0*OM(I))*1.D-7)
     &           *4.19D0*1.8D0**2*MW(I)
            CPC(I,4)=0.0D0
c ---------------------------------------------------------------------
         ENDIF
       ENDDO
       SUM=0.D0
       NMIN=NCOMP-1
       DO I=1,NMIN
          SUM=SUM+XX(I)
       ENDDO
       XX(NCOMP)=1.D0-SUM
       DO I=1,9
          READ (8,*)
       ENDDO
       WRITE (6,*) 'Use binary interaction coefficients kij (Y/N)'
       READ (5,*) ANSWER
       IF ((ANSWER .EQ. 'Y').OR.(ANSWER .EQ. 'y')) THEN
          IF (IEOSTYPE.EQ.0) THEN
             DO I=1,11
                READ (8,*) COMPONENT(I),KIJ(I,1),KIJ(I,2)
                KIJ(1,I)=KIJ(I,1)
                KIJ(2,I)=KIJ(I,2)
             ENDDO
          ELSE IF (IEOSTYPE.EQ.1) THEN
             DO I=1,14
                READ (8,*)
             ENDDO
             DO I=1,11
                READ (8,*) COMPONENT(I),KIJ(I,1),KIJ(I,2)
                KIJ(1,I)=KIJ(I,1)
                KIJ(2,I)=KIJ(I,2)
             ENDDO
          ENDIF
          DO I=12,NCOMP
             KIJ(I,1)=KIJ(11,1)
             KIJ(I,2)=KIJ(11,2)
             KIJ(1,I)=KIJ(I,1)
             KIJ(2,I)=KIJ(I,2)
          ENDDO
       ENDIF
    30 CLOSE (8)
       DO I=1,NCOMP
          SUMMOL=SUMMOL+XX(I)*MW(I)
          WRITE (6,100) COMPONENT(I),XX(I),MW(I),TC(I),PC(I)/1.D5,OM(I)
       ENDDO
       WRITE (6,120) SUMMOL
c       do i=1,ncomp
c          write (6,fmt=110) (kij(i,j),j=1,ncomp)
c       enddo
   100 FORMAT (A,2X,F8.4,2X,F8.4,2X,F8.2,2X,F10.5,2X,F8.4)
   110 FORMAT (14F5.3)
   120 FORMAT ('Average moleculair weight:',4X,F8.4,' g/mole')
       RETURN
       END

       SUBROUTINE LUMPING
```

```
C -----------------------------------------------------------------------
C     This subroutine is used to input all the necessary data for
C     lumped gas-condensate systems
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)

      DOUBLE PRECISION MWL(50),MW(15),KIJ,M
      DIMENSION SGL(50),TCL(50),PCL(50),CPCL(50,4),XXL(50),OML(50)
      DIMENSION NLUMP(10,2)
      CHARACTER*8 COMPONENTL(50),COMPONENT

      COMMON /FEEDFRAC/XX(15),COMPONENT(15)
      COMMON/XPROP/KIJ(15,15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/HEATCPCOEFF/CPC(15,4)
      COMMON/EOS/OMEGA(2),IEOSTYPE

C --- Parameters for Pedersen-Thomassen-Fredenslund correlations --------
      PARAMETER(E1=1.6312D2,E2=8.6052D1,E3=4.3475D-1,E4=-1.8774D3)
      PARAMETER(E5=-1.3408D-1,E6=2.5019D0,E7=2.0846D2,E8=-3.9872D3)
      PARAMETER(E9=7.4310D-1,E10=4.8122D-3,E11=9.6707D-3,E12=-3.7184D-6)
C --- Parameters for Aasberg-Petersen correlations----------------------
      PARAMETER (D1=190.8D0,D2=87.70D0,D3=0.6387D0,D4=-5247.7D0)
      PARAMETER (D5=-0.5836D0,D6=2.3211D0,D7=282.2D0,D8=-4484.9D0)
      PARAMETER (D9=2.829D-2,D10=4.248D-3,D11=1.156D-2,D12=-4.069D-6)
C -----------------------------------------------------------------------

      SUMMOL=0.D0

   10 OPEN (8,FILE='LUMPING.DAT')
      DO I=1,6
         READ (8,*)
      ENDDO
      READ(8,*) NCOMP
      READ(8,*)
      READ(8,*) NPLUS
      DO I=1,3
         READ (8,*)
      ENDDO
      NN=NCOMP-NPLUS
      DO I=1,NN
         READ (8,*) COMPONENTL(I),MWL(I),TCL(I),PCL(I),OML(I),
     &          CPCL(I,1),CPCL(I,2),CPCL(I,3),CPCL(I,4)
      END DO
      DO I=1,9
         READ (8,*)
      ENDDO
      DO I=1,NCOMP
         IF (I .LE. NN) THEN
            READ (8,*) COMPONENTL(I),XXL(I)
         ELSE
            READ (8,*) COMPONENTL(I),XXL(I),MWL(I),SGL(I)
c ----- Relations from Aasberg-Petersen, Stengby (1991) ---------------
            TCL(I)=D1*SGL(I)+D2*DLOG(MWL(I))+D3*MWL(I)+D4/MWL(I)
            PCL(I)=D5+D6*SGL(I)+D7/MWL(I)+D8/MWL(I)**2
            PCL(I)=DEXP(PCL(I))/1.01325D-5
            OML(I)=D9+D10*MWL(I)+D11*SGL(I)+D12*MWL(I)**2
c ----- Relations from Pedersen, Thomassen & Fredenslund (1988) -------
c            TCL(I)=E1*SGL(I)+E2*DLOG(MWL(I))+E3*MWL(I)+E4/MWL(I)
c            PCL(I)=DEXP(E5+E6*SGL(I)+E7/MWL(I)+E8/MWL(I)**2)/1.01325D-5
c            M=E9+E10*MWL(I)+E11*SGL(I)+E12*MWL(I)**2
c            IF (IEOSTYPE.EQ.0) THEN
c               OML(I)=(-1.574D0+DSQRT(1.574D0**2+0.704D0*(0.48D0-M)))/
c     &                -0.352D0
c            ELSE
c               OML(I)=(-1.54226D0+DSQRT(1.54266D0**2+4.D0*0.26992D0*
c     &              (0.37464D0-M)))/(2.D0*-0.26992D0)
c            ENDIF
c ----- Katz & Firoozabadi (1978) -------------------------------------
c            TB=97.58D0*MWL(I)**(0.3323D0)*SGL(I)**(0.04609D0)
c ----- Cavett Relations (1964) ---------------------------------------
c            API=141.5D0/SGL(I)-131.5D0
c            TB=TB*1.8D0-459.69D0
c            TCL(I)=768.071D0+1.7134D0*TB-0.10834D-2*TB**2+0.3889D-6*
c     &           TB**3-0.89213D-2*TB*API+0.53095D-6*TB**2*API+
c     &           0.32712D-7*TB**2*API**2
c            PCLOG=2.829+0.9412D-3*TB-0.30475D-5*TB**2+0.15141D-8*
```

```
c     &            TB**3-0.20876D-4*TB*API+0.11048D-7*TB**2*API+
c     &            0.1395D-9*TB**2*API**2-0.4827D-7*TB*API**2
c            PCL(I)=10.D0**PCLOG
c            TCL(I)=TCL(I)/1.8D0+255.37D0
c            TB=TB/1.8D0+255.37D0
c            PB=14.5037D0
c            PBR=PB/PCL(I)
c            TBR=TB/TCL(I)
c            PCL(I)=PCL(I)/14.5037D-5
c            WCF=4.5579D0*MWL(I)**(0.15178D0)*SGL(I)**(-0.84573D0)
C ---- Lee-Kesler relations (1976) ----------------------------------
c            IF (TBR.LT.0.8D0) THEN
c               OML(I)=(DLOG(PBR)-5.92714D0+6.09648D0/TBR+1.28862D0*DLOG
c     &               (TBR)-0.169347D0*TBR**6)/(15.2518D0-15.6875D0/TBR
c     &               -13.4721D0*DLOG(TBR)+0.43577D0*TBR**6)
c            ELSE
c               OML(I)=-7.904D0+0.1352D0*WCF-0.007465D0*WCF**2+8.359D0*
c     &              TBR+(1.408D0-0.01063D0*WCF)/TBR
c            ENDIF
c ----- Ideal Gas Heat Capacity Coefficients by Kesler & Lee (1976) ---
            WCF=4.5579D0*MWL(I)**(0.15178D0)*SGL(I)**(-0.84573D0)
            CF=((12.8D0-WCF)*(10.D0-WCF)/(10.D0*OML(I)))**2
            CPCL(I,1)=(-0.33886D0+0.02827D0*WCF-0.26105D0*CF+0.59332D0*
     &             OML(I)*CF)*4.19D0*MWL(I)
            CPCL(I,2)=(-1.D-4*(.9291D0-1.1543D0*WCF+.0368*WCF**2)+CF*
     &              (4.56D0-9.48D0*OML(I))*1.D-4)*4.19D0*1.8D0*MWL(I)
            CPCL(I,3)=(-1.6658D-7+CF*(0.536D0-0.6828D0*OML(I))*1.D-7)
     &              *4.19D0*1.8D0**2*MWL(I)
            CPCL(I,4)=0.0D0
c ------------------------------------------------------------------------
         ENDIF
       ENDDO
       READ (8,*)
       READ (8,*) NLMP
       DO I=1,NLMP
          READ (8,*) NLUMP(I,1),NLUMP(I,2)
       ENDDO
       SUM=0.D0
       NMIN=NCOMP
       DO I=1,NMIN
          SUM=SUM+XXL(I)
       ENDDO
       DO I=1,NCOMP
          XXL(I)=XXL(I)/SUM
       ENDDO
       DO I=1,NCOMP
          SUMMOL=SUMMOL+XXL(I)*MWL(I)
          WRITE (6,100) COMPONENTL(I),XXL(I),MWL(I),TCL(I),
      &          PCL(I)/1.D5,OML(I)
       ENDDO
       open (10,file='lumping',status='unknown')
       do i=1,ncomp
         write (10,140) componentL(i),xxL(i),xxL(i)*mwL(i)/summol,mwL(i),
      &    tcL(i),pcL(i)/1.d5,omL(i)
       enddo
       WRITE (10,*)
       do i=1,ncomp
          write (10,150) componentL(i),(cpcL(i,j),j=1,4)
       enddo
       NN=NLUMP(1,1)-1
       DO I=1,NN
          COMPONENT(I)=COMPONENTL(I)
          XX(I)=XXL(I)
          MW(I)=MWL(I)
          TC(I)=TCL(I)
          PC(I)=PCL(I)
          OM(I)=OML(I)
          CPC(I,1)=CPCL(I,1)
          CPC(I,2)=CPCL(I,2)
          CPC(I,3)=CPCL(I,3)
          CPC(I,4)=CPCL(I,4)
       ENDDO

       write (10,*)
       DO J=1,NLMP
          NN=NN+1
          sumx=0.d0
          sumxw=0.d0
          sumtc=0.d0
          sumpc=0.d0
```

```
          sumom=0.d0
          summw=0.d0
          sumcpc1=0.d0
          sumcpc2=0.d0
          sumcpc3=0.d0
          sumcpc4=0.d0
          do I=NLUMP(J,1),NLUMP(J,2)
             sumx=sumx+xxl(i)
             sumxw=sumxw+xxl(i)*mwl(i)/summol
             sumtc=sumtc+tcl(i)*xxl(i)*mwl(i)/summol
             sumpc=sumpc+pcl(i)*xxl(i)*mwl(i)/summol
             sumom=sumom+oml(i)*xxl(i)*mwl(i)/summol
             summw=summw+mwl(i)*xxl(i)*mwl(i)/summol
             sumcpc1=sumcpc1+cpcl(i,1)*xxl(i)*mwl(i)/summol
             sumcpc2=sumcpc2+cpcl(i,2)*xxl(i)*mwl(i)/summol
             sumcpc3=sumcpc3+cpcl(i,3)*xxl(i)*mwl(i)/summol
             sumcpc4=sumcpc4+cpcl(i,4)*xxl(i)*mwl(i)/summol
          enddo
          READ (8,*) COMPONENT(NN)
          XX(NN)=SUMX
          MW(NN)=SUMMW/SUMXW
          TC(NN)=SUMTC/SUMXW
          PC(NN)=SUMPC/SUMXW
          OM(NN)=SUMOM/SUMXW
          CPC(NN,1)=SUMCPC1/SUMXW
          CPC(NN,2)=SUMCPC2/SUMXW
          CPC(NN,3)=SUMCPC3/SUMXW
          CPC(NN,4)=SUMCPC4/SUMXW
        ENDDO
        NCOMP=NN
        WRITE (6,*)
        write (6,*) 'Composition after lumping'
        write (6,*)
        do i=1,ncomp
          write (6,160) component(i),xx(i),mw(i),Tc(i),pc(i)/1.d5,om(i)
          write (10,160) component(i),xx(i),mw(i),Tc(i),pc(i)/1.d5,om(i)
        ENDDO
        WRITE (10,*)
        DO I=1,NCOMP
          WRITE (10,150) COMPONENT(I),(CPC(I,J),J=1,4)
        ENDDO
        WRITE (6,*)
        WRITE (6,170) SUMMOL
        WRITE (6,*)
        CLOSE (8)
        CLOSE (10)
  100 FORMAT (A,2X,F8.4,2X,F8.4,2X,F8.2,2X,F10.5,2X,F8.4)
  110 FORMAT (14F5.3)
  120 FORMAT ('Average moleculair weight:',4X,F8.4,' g/mole')
  130 FORMAT (A,2X,F8.5,2X,F7.5,2X,F8.6,2X,F8.6)
  140 format (a,f7.4,2x,f7.4,2x,f6.2,2x,f8.4,2x,f8.4,2x,f7.5)
  150 format (a,f8.4,2x,e11.4e2,2x,e11.4e2,2x,e11.4e2)
  160 format (a,f7.4,2x,f6.2,2x,f8.4,2x,f8.4,2x,f7.5)
  170 FORMAT ('Average moleculair weight:',4X,F8.4,' g/mole')
        RETURN
  199 END


      SUBROUTINE ENTHALPY(MSLOPE,XX,T,P,H,IER,V,HDT,HRES)

C -----------------------------------------------------------------------
C     This subroutine is used to input to calculate the enthalpy and
C     temperature derivative uisng EoS
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION INTCP

      DIMENSION XX(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/TYPE/ICALC
      COMMON/HEATCPCOEFF/CPC(15,4)
      COMMON/EOS/OMEGA(2),IEOSTYPE
      COMMON/INVERSION/DVDT

      PARAMETER (HREF=0.D0,T0=298.15D0)
```

```
      IER=0

      CALL VOLUME (XX,T,P,MSLOPE,V,DUMMY)

      AAM=AM(1)*P/(RK*RK*T*T)
      BBM=(BM*P)/(RK*T)
      ZM=(V*P)/(RK*T)

      CP=0.D0
      DO I=1,NCOMP
          U=XX(I)*(CPC(I,1)+CPC(I,2)*T+CPC(I,3)*T**2
     +          +CPC(I,4)*T**3.D0)
          CP=CP+U
      END DO

      INTCP=0.D0
      DO I=1,NCOMP
         U=XX(I)*(CPC(I,1)*(T-T0)+0.5D0*CPC(I,2)*(T**2-T0**2)+
     +      CPC(I,3)/3.D0*(T**3-T0**3)+CPC(I,4)/4.D0*(T**4-T0**4))
         INTCP=INTCP+U
      END DO

      ENTH1=RK*T*(ZM-1.D0)

      IF (IEOSTYPE.EQ.0) THEN
         ENTH2=(AM(1)-T*AM(2))/BM
         ENTH3=DLOG(1.0D0+BM/V)
         DPDT=RK/(V-BM)-AM(2)/(V*(V+BM))
         DPDV=-RK*T/(V-BM)**2+AM(1)/(V**2*(V+BM))+AM(1)/(V*(V+BM)**2)
         ENTHDT1=T/BM*AM(3)*DLOG(1.D0+BM/V)
         ENTHDT2=(P+(AM(1)-AM(2)*T)/(V*(V+BM)))
      ENDIF
      IF (IEOSTYPE.EQ.1) THEN
         ENTH2=DSQRT(2.D0)/(4.D0*BM)*(AM(1)-T*AM(2))
         ENTH3=DLOG((V+(SQR2+1.D0)*BM)/(V-(SQR2-1.D0)*BM))
         DPDT=RK/(V-BM)-AM(2)/(V*(V+BM)+BM*(V-BM))
         DPDV=-RK*T/(V-BM)**2+2.D0*AM(1)*(V+BM)/(V*(V+BM)+BM*
     +        (V-BM))**2
         ENTHDT1=SQR2*T/(4.D0*BM)*AM(3)
         ENTHDT1=ENTHDT1*DLOG((V+(SQR2+1.D0)*BM)/(V-(SQR2-1.D0)*BM))
         ENTHDT2=(P+(AM(1)-AM(2)*T)/(V*(V+BM)+BM*(V-BM)))
      ENDIF
      IF (IEOSTYPE.EQ.2) THEN
         ENTH2=3.D0*AM(1)/(2.D0*BM)
         ENTH3=DLOG(1.D0+BM/V)
         DPDT=RK/(V-BM)+AM(1)/(2.D0*T*V*(V+BM))
         DPDV=-RK*T/(V-BM)**2+AM(1)/(V**2*(V+BM))+AM(1)/
     +            (V*(V+BM)**2)
         ENTHDT1=3.D0/(4.D0*BM*T)*AM(1)*DLOG(1.D0+BM/V)
         ENTHDT2=(P+3.D0*AM(1)/(2.D0*V*(V+BM)))
      ENDIF

      H0=HREF+INTCP
      DVDT=-DPDT/DPDV

      H=H0+(ENTH1-ENTH2*ENTH3)
      HDT=CP-RK+ENTHDT1+DVDT*ENTHDT2
      HRES=ENTH1-ENTH2*ENTH3

      RETURN
      END


      SUBROUTINE FUGACITY(I,Z,T,V,P,FUGCOEFF)
C
C
C         THIS SUBROUTINE GIVES THE FUGACITY COEFFICIENT
C         OF COMPONENT *I* AT THE SPECIFIED COMPOSITION,TEMPERATURE
C         AND VOLUME.
C
C         VARIABLES
C             Z(I)      COMPOSITION OF COMPONENT I(MOL FRACTION)
C             V         VOLUME (LIT/GM-MOL)
C             T         TEMPERATURE (DEGREES KELVIN)
C             FUGCOEFF FUGACITY COEFFICIENT OF COMPONENT I
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DIMENSION Z(15)
```

```
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/IJPROP/A(15,15),B(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE

      AAM=(AM(1)*P)/(RK*RK*T*T)
      BBM=(BM*P)/(RK*T)
      ZM=(V*P)/(RK*T)

      DUM1=-DLOG(ZM-BBM)
      DUM2=(ZM-1.0D0)*B(I)/BM

c --- Calculations for Redlich-Kwong & Soave-Redlich-Kwong ------------
      IF (IEOSTYPE.EQ.0 .OR. IEOSTYPE.EQ.2) THEN
        DUM3=AAM/BBM*(2.0D0*CHEM1(I)/AM(1)-B(I)/BM)
        DUM4=DLOG(1.D0+BBM/ZM)
      ENDIF
c --- Calculations for Peng-Robinson ----------------------------------
      IF (IEOSTYPE.EQ.1) THEN
        DUM3=AAM/(2.0D0*SQR2*BBM)*(2.0D0*CHEM1(I)/AM(1)-B(I)/BM)
        DUM4=DLOG((ZM+2.414D0*BBM)/(ZM-0.414D0*BBM))
      ENDIF

      FUGCOEFF=DUM1+DUM2-DUM3*DUM4

      IF(IPRINT.GE.5)WRITE(6,102)P,V,AM(1),BM
      IF(IPRINT.GE.5)WRITE(6,102)AAM,BBM,ZM,DUM2

      IF(IPRINT.GE.5)WRITE(6,102)DUM1,DUM2,DUM3,DUM4
  102 FORMAT(' SUB CHP. DUM1-4 **',5E12.4,'**')
      IF(IPRINT.GE.5)WRITE(6, 303) CPI
  303 FORMAT(2X,'CPI =',E12.4)
      RETURN
      END


      SUBROUTINE KFACTR(X,Y,T,P,VL,VG,K,IER)

C             X(I)  APPROXIMATE LIQUID PHASE MOL FRACTION OF
C                   COMPONENT I
C             Y(I)  APPROXIMATE VAPOR PHASE MOL FRACTION OF
C                   COMPONENT I
C             K(I)  NEW ESTIMATE OF KFACTOR FOR COMPONENT I
C             VL    LIQUID VOLUME (LITERS/MOLE)
C             VG    VAPOR VOLUME  (LITERS/MOLE)
C
C             MSLOPE CONTROL DIRECTION OF SEARCH IN SUB VOLUME
C                    +1 = LIQUID PHASE CALCULATION
C                    -1 = VAPOR PHASE CALCULATION.
C             I     COMPONENT INDEX

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)

      DOUBLE PRECISION K(15)

      DIMENSION X(15),Y(15),FUGLIQ(15),FUGGAS(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/TYPE/ICALC

      IER=0

c --- Calculation of Liquid Properties --------------------------------
      MSLOPE=1
      CALL VOLUME (X,T,P,MSLOPE,VL,DUMMY)
      DO I = 1,NCOMP
        CALL FUGACITY(I,X,T,VL,P,FUGCOEFF)
        FUGLIQ(I)=FUGCOEFF
      ENDDO
c --- Calculation of Vapor Properties ---------------------------------
      IF (ICALC.EQ.0) THEN          !VLE CALCULATIONS
        MSLOPE=-1
      ENDIF
      IF (ICALC.EQ.1) THEN          !LLE CALCULATIONS
        MSLOPE=1
      ENDIF
      CALL VOLUME(Y,T,P,MSLOPE,VG,DUMMY)
c ---------------------------------------------------------------------
      DO I = 1,NCOMP
```

```
         CALL FUGACITY(I,Y,T,VG,P,FUGCOEFF)
         FUGGAS(I)=FUGCOEFF
      ENDDO

      DO I = 1,NCOMP
         K(I)=DEXP(FUGLIQ(I)-FUGGAS(I))
      ENDDO

      IF(IPRINT.GE.1) WRITE(6,300) (I,K(I),I=1,NCOMP)

  300 FORMAT('SUB K-FACTOR'/,3('K(',I1,') = ',F8.6,2X)/,
     + 4('K(',I1,') = ',F8.6,2X)/)
  301 FORMAT(2X,'FUGACITY COEFFICIENT OF COMPONENT',I3,' ='G15.4)
  310 FORMAT(10X,'ENTERING SUB KFACTOR',5X,'T = ',F15.3,5X,'P = ',
     1 E15.5,5X,10(/50X,2F15.4))
      IF(IPRINT.EQ.6) STOP
      RETURN
      END


      SUBROUTINE MIXPRP (Z,T)

C -----------------------------------------------------------------------
C     This subroutine calculates all necessaru mixture properties
C
C ------------------------------------------------------- 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)

      DOUBLE PRECISION KIJ,M(15)

      DIMENSION Z(15),AC(15)

      COMMON/XPROP/KIJ(15,15)
      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/IJPROP/A(15,15),B(15)
      COMMON/PURE/TC(15),PC(15),OM(15)
      COMMON/EOS/OMEGA(2),IEOSTYPE

      AM(1) = 0.0D0
      AM(2) = 0.0D0
      AM(3) = 0.0D0
      BM = 0.0D0

      DO I=1,NCOMP
         IF (IEOSTYPE.EQ.0) THEN
            M(I)=0.48D0+1.574D0*OM(I)-0.176D0*OM(I)**2
         ELSE IF (IEOSTYPE.EQ.1) THEN
            M(I)=0.37464D0+1.54226D0*OM(I)-0.26992D0*OM(I)**2
         ELSE
            M(I)=0.D0
         ENDIF
      ENDDO

      DO I=1,NCOMP
         B(I)=OMEGA(2)*RK*TC(I)/PC(I)
         BM=BM+Z(I)*B(I)
         ALPHA=(1.0D0+M(I)*(1.0D0-DSQRT(T/TC(I))))**2
         IF (IEOSTYPE.EQ.2) THEN
            ATC=OMEGA(1)*(RK*TC(I))**2*DSQRT(TC(I))/(PC(I)*DSQRT(T))
         ELSE
            ATC=OMEGA(1)*(RK*TC(I))**2/PC(I)
         ENDIF
         A(I,I)=ATC*ALPHA
      ENDDO

      DO I=1,NCOMP
         DO J=1,NCOMP
            A(I,J)=(1.D0-KIJ(I,J))*DSQRT(A(I,I)*A(J,J))
            AM(1)=AM(1)+Z(I)*Z(J)*A(I,J)
         ENDDO
      ENDDO

      IF (IEOSTYPE.NE.2) GOTO 50

      DO I=1,NCOMP
         DO J=1,NCOMP
             U2=-0.5D0/T*Z(I)*Z(J)*DSQRT(A(I,I)*A(J,J))*(1.D0-KIJ(I,J))
            U3=0.75D0/T**2*Z(I)*Z(J)*DSQRT(A(I,I)*A(J,J))*
```

```
     &          (1.D0-KIJ(I,J))
           AM(2)=AM(2)+U2
           AM(3)=AM(3)+U3
         ENDDO
       ENDDO
       GOTO 100

c --- AM(1) is the first derivative of a(T) with respect to temperature
   50 DO I=1,NCOMP
         DO J=1,NCOMP
           AC(I)=OMEGA(1)*(RK*TC(I))**2.0D0/PC(I)
           AC(J)=OMEGA(1)*(RK*TC(J))**2.0D0/PC(J)
           U=0.5D0*Z(I)*Z(J)*DSQRT(AC(I)*AC(J))*(2.D0*M(I)*M(J)/
     +          DSQRT(TC(I)*TC(J))-(M(I)+M(I)*M(J))/DSQRT(T*TC(I))-
     +           (M(J)+M(I)*M(J))/DSQRT(T*TC(J)))*(1.D0-KIJ(I,J))
           AM(2)=AM(2)+U
         END DO
       END DO
c --- AM(2) is the second derivative of a(T) with respect to temperature
       DO I=1,NCOMP
         DO J=1,NCOMP
           AC(I)=OMEGA(1)*(RK*TC(I))**2.0D0/PC(I)
           AC(J)=OMEGA(1)*(RK*TC(J))**2.0D0/PC(J)
           U=0.25D0*Z(I)*Z(J)*DSQRT(AC(I)*AC(J))*((M(I)+M(I)*
     +       M(J))/(T*DSQRT(T*TC(I)))+(M(J)+M(I)*M(J))/(T*DSQRT
     +        (T*TC(J))))*(1.D0-KIJ(I,J))
           AM(3)=AM(3)+U
         END DO
       END DO

  100 DUM=0.0D0
       DO I=1,NCOMP
         DUM=0.0D0
         DO J=1,NCOMP
           DUM=DUM+Z(J)*A(J,I)
         ENDDO
         CHEM1(I)=DUM
       ENDDO
       RETURN
  199 END


C      ***CUBIC*************************************************08.11.1986
C      SOLUTION OF A CUBIC EQUATION
C      EQUATIONS OF LESSER ORDER ARE SOLVED BY THE APPROPRIATE FORMULAS.
C      THE SOLUTIONS ARE ARRANGED IN ASCENDING ORDER.
C      ********************************************************************
C      A(0:3)       (I)  VECTOR CONTAINING THE POLYNOMIAL COEFFICIENTS
C      X(1:L)       (O)  RESULTS
C      L            (O)  NUMBER OF VALID SOLUTIONS (BEGINNING WITH X(1))
C      ====================================================================
       SUBROUTINE ROOTFINDER(A,X,L)
       IMPLICIT DOUBLE PRECISION(A-H,O-Z)
       DIMENSION A(0:3),X(3),U(3)
       PARAMETER(PI=3.1415926535897932D+0,THIRD=1.D+0/3.D+0)
C      ==== DEFINE CUBIC ROOT AS STATEMENT FUNCTION ====
       CBRT(Z)=SIGN(ABS(Z)**THIRD,Z)
C      ==== IS THIS REALLY A CUBIC EQUATION? ====
       IF (A(3).EQ.0.D+0) GOTO 300
       W=A(2)/A(3)*THIRD
       P=(A(1)/A(3)*THIRD-W**2)**3
       Q=-.5D+0*(2.D+0*W**3-(A(1)*W-A(0))/A(3))
       DIS=Q**2+P
       IF (DIS.LT.0.D+0) THEN
C        ==== THREE REAL SOLUTIONS ====
C              CONFINE THE ARGUMENT OF ACOS TO THE INTERVAL [-1;1]
         PHI=ACOS(MIN(1.D+0,MAX(-1.D+0,Q/DSQRT(-P))))
         P=2.D+0*(-P)**(5.D-1*THIRD)
         DO 200 I=1,3
  200      U(I)=P*COS((PHI+DBLE(2*I)*PI)*THIRD)-W
       X(1)=MIN(U(1),U(2),U(3))
       X(2)=MAX(MIN(U(1),U(2)),MIN(U(1),U(3)),MIN(U(2),U(3)))
       X(3)=MAX(U(1),U(2),U(3))
         L=3
       ELSE
C        ==== ONLY ONE REAL SOLUTION ====
         DIS=DSQRT(DIS)
         X(1)=CBRT(Q+DIS)+CBRT(Q-DIS)-W
         L=1
       END IF
```

```
      RETURN
C     ==== DEAL WITH EQUATION OF LESS THAN CUBIC ORDER ====
  300 IF (A(2).NE.0.D+0) THEN
C       ==== QUADRATIC EQUATION ====
        P=5.D-1*A(1)/A(2)
        DIS=P**2-A(0)/A(2)
        IF (DIS.GE.0.D+0) THEN
C         ==== 2 SOLUTIONS ====
          X(1)=-P-DSQRT(DIS)
          X(2)=-P+DSQRT(DIS)
          L=2
        ELSE
C         ==== NO SOLUTION ====
          L=0
        END IF
      ELSE IF (A(1).NE.0.D+0) THEN
C       ==== LINEAR EQUATION ====
        X(1)=-A(0)/A(1)
        L=1
      ELSE
C       ==== NO EQUATION ====
        L=0
      END IF
C     ==== PERFORM ONE STEP OF A NEWTON ITERATION IN ORDER TO MINIMIZE
C          ROUND-OFF ERRORS ====
      DO 310 I=1,L
        X(I)=X(I)-(A(0)+X(I)*(A(1)+X(I)*(A(2)+X(I)*A(3))))
     *  /(A(1)+X(I)*(2.D+0*A(2)+X(I)*3.D+0*A(3)))
  310 CONTINUE
      RETURN
      END


      SUBROUTINE RTSAFES(FUNCT,X1,X2,P,XACC,RTSAFE)

C ---------------------------------------------------------------------
C     One dimensional Newton-Raphson iteration procedure
C
C ------------------------------------------------------ 23/11/1994 ---


      IMPLICIT DOUBLE PRECISION(A-H,O-Z), INTEGER (I-M)

      PARAMETER (MAXIT=100)

      EXTERNAL FUNCT

      CALL FUNCT (X1,P,FL,DF)
      CALL FUNCT (X2,P,FH,DF)
      IF (FL*FH .GE. 0.D0) PAUSE 'root must be bracketed'
      IF (FL .LT. 0.D0) THEN
          XL=X1
          XH=X2
      ELSE
          XH=X1
          XL=X2
      ENDIF
      RTSAFE=.5D0*(X1+X2)
      DXOLD=DABS(X2-X1)
      DX=DXOLD
      CALL FUNCT (RTSAFE,P,F,DF)
      DO 11 J=1,MAXIT
          IF (((RTSAFE-XH)*DF-F)*((RTSAFE-XL)*DF-F) .GE. 0.D0
     +          .OR. DABS(2.D0*F) .GT. DABS(DXOLD*DF)) THEN
              DXOLD=DX
              DX=0.5D0*(XH-XL)
              RTSAFE=XL+DX
              IF (XL .EQ. RTSAFE) RETURN
          ELSE
              DXOLD=DX
              DX=F/DF
              TEMP=RTSAFE
              RTSAFE=RTSAFE-DX
              IF (TEMP .EQ. RTSAFE) RETURN
          ENDIF
          IF (DABS(DX) .LT. XACC) RETURN
          CALL FUNCT (RTSAFE,P,F,DF)
          IF (F .LT. 0) THEN
              XL=RTSAFE
          ELSE
```

```
            XH=RTSAFE
         ENDIF
   11 CONTINUE
      PAUSE 'RTSAFE exceeding maximum iterations'
      RETURN
      END


      SUBROUTINE CONS

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/EOS/OMEGA(2),IEOSTYPE

      SQR2 = DSQRT(2.D0)
      PI = 3.141592654D0
      RK = 8.314413D0
      C74=PI*SQR2/6.D0
      IF (IEOSTYPE.EQ.0 .OR. IEOSTYPE.EQ.2) THEN
         OMEGA(1)=0.42748D0
         OMEGA(2)=0.08664D0
      ELSE IF (IEOSTYPE.EQ.1) THEN
         OMEGA(1)=0.457235D0
         OMEGA(2)=0.077796D0
      ENDIF
      RETURN
      END


      SUBROUTINE VOLUME (XY,T,P,MSLOPE,V,IROOTNUMBER)
C
C         MSLOPE = 0  FOR FINDING NUMBER OF ROOTS
C         MSLOPE = +1 FOR LIQUID PHASE
C         MSLOPE = -1 FOR VAPOR PHASE
C
      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER (I-N)

      DIMENSION XY(15)
      DIMENSION Z(0:3),ROOTS(3)

      COMMON/MIX/AM(3),BM,CHEM1(15)
      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
      COMMON/EOS/OMEGA(2),IEOSTYPE

      IROOTNUMBER=0
      ROOTS(1)=0
      ROOTS(2)=0
      ROOTS(3)=0

      CALL MIXPRP (XY,T)
      AA=AM(1)*P/(RK*T)**2
      BB=BM*P/(RK*T)
      IF (IEOSTYPE .EQ. 1) THEN
         Z(3)=1.0D0
         Z(2)=-1.0D0+BB
         Z(1)=AA-2.0D0*BB-3.0D0*BB**2
         Z(0)=-AA*BB+BB**2+BB**3
      ELSE
         Z(3)=1.0D0
         Z(2)=-1.0D0
         Z(1)=AA-BB-BB**2
         Z(0)=-AA*BB
      ENDIF
      CALL ROOTFINDER(Z,ROOTS,IROOTNUMBER)
      IF (MSLOPE.EQ.0) RETURN
      IF (IROOTNUMBER.EQ.1) THEN
         V=DMAX1(ROOTS(1),ROOTS(2),ROOTS(3))*RK*T/P
      ELSE
         IF (DMIN1(ROOTS(1),ROOTS(2),ROOTS(3))*RK*T/P.LT.BM) THEN
            V=DMAX1(ROOTS(1),ROOTS(2),ROOTS(3))*RK*T/P
         ELSE IF (MSLOPE.EQ.1) THEN
            V=DMIN1(ROOTS(1),ROOTS(2),ROOTS(3))*RK*T/P
         ELSE IF (MSLOPE.EQ.-1) THEN
            V=DMAX1(ROOTS(1),ROOTS(2),ROOTS(3))*RK*T/P
         ENDIF
      ENDIF
      RETURN
      END
```

```
      SUBROUTINE FLASH(ITYPE,VF,XF,YF,TL,PF,HF,A,X,Y,T,P,IERF,VL,
     &                 VG,HFIN)

C ---------------------------------------------------------------------
C     Two-dimensional Newton-Raphson scheme to solve isenthalpic flash
C ------------------------------------------------------ 23/11/1994 ---

      IMPLICIT DOUBLE PRECISION (A-H,O-Z), INTEGER(I-N)

      DOUBLE PRECISION K(15),K1,KV

      DIMENSION XF(15),YF(15),X(15),Y(15),Z(15),U(15),W(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP

      PARAMETER (EPS=1.D-7)

      N=NCOMP
  100 IERF=0
      KV=1
* CHECK IF FEED IS LIQUID (KV=1), VAPOR (KV=3), OR BOTH (KV=2)
      IF(VF.GT.1.D-3) KV=2
      IF(VF.GT.0.999D0) KV=3
      SX=0.D0
      SY=0.D0
      SXY=0.D0
      SYF=0.D0
* SUM FEED AND PHASE MOLE FRACTIONS
      DO 109 I=1,N
         SX=SX+X(I)
         SY=SY+Y(I)
         SXF=SXF+XF(I)
  109 SYF=SYF+YF(I)
* CHECK THAT SUM OF FEED MOL FRACTIONS IS UNITY
      IF (KV.LE.2.AND.DABS(SXF-1.D0).GT.0.01D0) GOTO 903
      IF (KV.GE.2.AND.DABS(SYF-1.D0).GT.0.01D0) GOTO 903
* CHECK THAT FLASH PRESSURE IS WITHIN LIMITS
      IF (P.LT.1.D-1.OR.P.GT.1000.D5) GOTO 903
  110 GOTO (112,114,116),KV
* FOR LIQUID FEED USE XF AS ESTIMATE FOR X AND Y IF NOT GIVEN AND SET
* TOTAL FEED Z TO XF
  112 DO 113 I=1,N
         IF (DABS(SX-1.D0).GT.0.01D0) X(I)=XF(I)
         IF (DABS(SY-1.D0).GT.0.01D0) Y(I)=XF(I)
  113 Z(I)=XF(I)
      GOTO 120
* FOR MIXED FEED USE XF AS ESTIMATE FOR X AND YF FOR Y IF NOT GIVEN AND
* FIND TOTAL FEED Z.
  114 DO 115 I=1,N
         IF (DABS(SX-1.D0).GT.0.01D0) X(I)=XF(I)
         IF (DABS(SY-1.D0).GT.0.01D0) Y(I)=YF(I)
  115 Z(I)=(1.D0-VF)*XF(I)+VF*YF(I)
      GOTO 120
* FOR VAPOR FEED USE YF AS ESTIMATE FOR X AND Y IF NOT GIVEN AND SET
* TOTAL FEED Z TO YF
  116 DO 117 I=1,N
         IF (DABS(SX-1.D0).GT.0.01D0) X(I)=YF(I)
         IF (DABS(SY-1.D0).GT.0.01D0) Y(I)=YF(I)
  117 Z(I)=YF(I)
* INITIALIZE LIQUID AND VAPOR PRODUCT COMPOSITIONS
  120 DO 121 I=1,N
        U(I)=XF(I)
  121 W(I)=YF(I)
* FIND INITIAL ESTIMATE FOR A (IF NOT GIVEN) FOR ISOTHERMAL FLASH
c       TB=TL
c       IF (T.GT.100.D0.AND.T.LT.600.D0) TB=T
c       TD=TB
c       CALL BUDET(1,Z,W,TB,P,DUM1,DUM2,IEB)
c       IF (IEB.GT.1) GOTO 900
c       CALL BUDET(2,U,Z,TD,P,DUM1,DUM2,IER)
c       IF (IER.GT.1) GOTO 900
c       IF (ITYPE.EQ.2) GOTO 125
c       IF (IEB.LT.-2) TB=(T-TD*VF)/(1.1D0-VF)
c       IF (T.LT.TB) GOTO 901
c       IF (T.GT.TD) GOTO 902
c       IF (A.LT.1.D-3.OR.A.GT.0.999D0) A=(T-TB)/(TD-TB)
c       GOTO 150
* FIND FEED ENTHAPLY FOR ADIABATIC FLASH
c  125 HLF=0.D0
```

```
c       HVF=0.D0
c       IF (PF.LT.1.D-1.OR.PF.GT.500.D5) GOTO 903
c       IF (KV.GT.2) GOTO 126
c       IF (TL.LT.100.D0.OR.TL.GT.600.D0) GOTO 903
c       CALL ENTHALPY (1,XF,TL,PF,HLF,IER,VL,DUM,DUMMY)
c       IF (IER.GT.1) GOTO 900
c  126 IF (KV.LT.2) GOTO 127
c       IF (TV.LT.200.D0.OR.TV.GT.600.D0) GOTO 903
c       CALL ENTHALPY (-1,YF,TV,PF,HVF,IER,VG,DUM,DUMMY)
c       IF (IER.GT.1) GOTO 900
c  127 HF=(1.D0-VF)*HLF+VF*HVF
* FIND INITIAL ESTIMATE FOR A AND T (IF NOT GIVEN) FOR ADIABATIC FLASH
c       IF (IEB.LT.-2) TB=200.D0
c       IF (A.GT.1.D-3.OR.A.LT.0.999D0) GOTO 128
c       CALL ENTHALPY (1,Z,TB,P,HL,IER,VL,DUM,DUMMY)
c       IF (IER.GT.1) GOTO 900
c       CALL ENTHALPY (-1,Z,TD,P,HV,IER,VG,DUM,DUMMY)
c       IF (IER.GT.1) GOTO 900
c       IF (HF.LT.HL) GOTO 901
c       IF (HF.GT.HV) GOTO 902
c       A=(HF-HL)/(HV-HL)
c  128 IF (T.GT.TB.AND.T.LT.TD) GOTO 150
c       T=TB+A*(TD-TB)
* INITIALIZE ITERATIVE SOLUTION
  150 IT=0
      SL=1.D0
      KD=0
      TN=T
      AN=A
      write (6,*) 'starting initialization'
      GOTO 250
* CONDUCT NEWTON-RAPHSON ITERATION (200 SERIES STATEMENTS).
  200 IT=IT+1
      IF (IT.EQ.1) GOTO 205
      IF (ITYPE.EQ.2) GOTO 203
      IF (DABS(DFA).GE.0.1D0) GOTO 203
      EPF=EPS*10.D0*DABS(DFA)
      IF (DABS(F).LE.EPF) GOTO 300
      GOTO 204
* CHECK CONVERGENCE OF OBJECTIVE FUNCTION
  203 IF (FV.LE.EPS) GOTO 300
* EXIT WITHOUT SOLUTION AFTER 50 ITERATIONS
  204 IF (IT.GT.500) GOTO 900
  205 FO=FV
      KD=0
      IF (ITYPE.NE.2) GOTO 280
* DETERMINE KS AND ENTHALPIES AT 0.2 K T INCREASE FOR T DERIVATIVES
* (ADIABATIC)
  210 CALL KFACTR(X,Y,T+.1D-2,P,VLDT,VGDT,K,IER)
      IF (IER.GT.1) GOTO 900
      CALL ENTHALPY (1,X,T+.1D-2,P,HL,IER,VLDT,DUM,DUMMY)
      IF (IER.GT.1) GOTO 900
      CALL ENTHALPY (-1,Y,T+.1D-2,P,HV,VGDT,DUM,DUMMY)
      IF (IER.GT.1) GOTO 900
      FP=0.D0
* UPDATE PHASE COMPOSITIONS (ADIABATIC)
  211 DO 214 I=1,N
        X(I)=U(I)/SX
        Y(I)=W(I)/SY
        K1=K(I)-1.D0
  214 FP=FP+K1*Z(I)/(K1*A+1.D0)
      GP=(A*(HV-HL)+HL)/HF-1.D0
* FIND TEMPERATURE DERIVATIVES BY FINITE DIFFERNCE (ADIABATIC)
      DFT=(FP-F)/.1D-2
      DGT=(GP-G)/.1D-2
* SOLVE 2 DIMENSIONAL NEWTON-RAPHSON ITERATION FOR A AND T CORRECTIONS
* (ADIABATIC)
  220 DET=DFA*DGT-DFT*DGA
      DA=(F*DGT-G*DFT)/DET
      DT=(G*DFA-F*DGA)/DET
  230 AN=A-SL*DA
* LIMIT A TO RANGE 0-1
      IF (AN.LT.0.D0) AN=0.01D0
      IF (AN.GT.1.D0) AN=0.99D0
      IF (ITYPE.NE.2) GOTO 239
      TN=T-SL*DT
* LIMIT T TO RANGE (TB-TD) ADIABATIC
c       IF (TN.LT.TB) GOTO 235
c       IF (TN.GT.TD) GOTO 237
      GOTO 250
```

```
* CORRECT Y FOR T SET TO TB
  235 IF (IEB.LT.-2) GOTO 903
      CALL BUDET(1,Z,Y,TB,P,DUM1,DUM2,IEB)
      IF (IEB.GT.1) GOTO 900
      TN=TB
      GOTO 251
* CORRECT X FOR T SET TO TD
  237 CALL BUDET(2,X,Z,TD,P,DUM1,DUM2,IER)
      IF (IER.GT.1) GOTO 900
      TN=TD
      GOTO 251
  239 IF (KD.EQ.1) GOTO 251
* GET NEW K-VALUES
  250 CALL KFACTR(X,Y,T,P,VL,VG,K,IER)
      IF (IER.GT.1) GOTO 900
  251 SX=0.D0
      SY=0.D0
      DFA=0.D0
* FIND EQUILIBRIUM OBJECTIVE FUNCTION F AND UNNORMALIZED COMPOSITIONS
  252 DO 254 I=1,N
        K1=K(I)-1.D0
        U(I)=Z(I)/(K1*AN+1.D0)
        W(I)=K(I)*U(I)
        SX=SX+U(I)
        SY=SY+W(I)
* FIND DERIVATIVE OF F WRT A
        DFA=DFA-(W(I)-U(I))**2/Z(I)
  254 CONTINUE
      F=SY-SX
      IF (ITYPE.NE.2) GOTO 260
* FIND ENTHALPY OBJECTIVE FUNCTION G (ADIABATIC)
  255 CALL ENTHALPY(1,X,TN,P,HL,IER,VL,DUM,DUMMY)
      IF (IER.GT.1) GOTO 900
      CALL ENTHALPY(-1,Y,TN,P,HV,IER,VG,DUM,DUMMY)
      IF (IER.GT.1) GOTO 900
      G=(AN*(HV-HL)+HL)/HF-1.D0
* FIND DERIVATIVE OF G WRT A (ADIABATIC)
      DGA=(HV-HL)/HF
* FIND NORM FOR OBJECTIVE FUNCTION AND CHECK FOR DECREASE
  260 FV=DABS(F)
      IF (ITYPE.EQ.2) FV=DSQRT((F*F+G*G)/2.D0)
      IF (IT.EQ.0) GOTO 200
      IF (IT.LE.2) GOTO 270
      IF (KD.EQ.2) GOTO 270
      IF (FV.LE.FO) GOTO 270
* APPLY STEP-LIMITING PROCEDURE TO DECREASE OBJECTIVE FUNCTION
  265 KD=1
      SL=0.1D0*SL
* CHECK FOR FAILURE OF OF STEP-LIMITING
      IF (SL.LT.0.01D0) GOTO 268
      GOTO 230
* ABANDON STEP-LIMITING AND PROCEED
  268 AN=A
      TN=T
      KD=2
      GOTO 250
* PROCEED TO NEXT ITERATION
  270 A=AN
      T=TN
      SL=1.D0
      GOTO 200
* FIND NEWTON-RAPHSON CORRECTION TO A (ISOTHERMAL)
  280 DA=F/DFA
* UPDATE PHASE COMPOSITIONS (ISOTHERMAL)
  281 DO 284 I=1,N
        X(I)=U(I)/SX
  284 Y(I)=W(I)/SY
      GOTO 230
* FOR CONVERGED ITERATION GET FINAL NORMALIZED PHASE COMPOSITIONS
  300 DO 304 I=1,N
        X(I)=U(I)/SX
  304 Y(I)=W(I)/SY
      IERF=IER
      HFIN=A*(HV-HL)+HL
      WRITE (6,*) 'FLASH CONVERGED IN ',IT,' ITERATIONS'
      WRITE (6,*)
      WRITE (6,1000) HF
      WRITE (6,1001) PF/1.D5,TL
      WRITE (6,*)
      WRITE (6,1002) HFIN
```

```
      WRITE (6,1003) P/1.D5,T
      RETURN
* ON FAILURE TO CONVERGE ITERATION SET A TO 0 (T TO 0 ADIABATIC) AND
* IERF TO 2
  900 IERF=2
      GOTO 905
* FOR T LESS THAN TB SET IERF TO 3 (A TO 0)
  901 IERF=3
      GOTO 905
* FOR T GREATER THAN TD SET IERF TO 4 (A TO 1)
  902 IERF=4
      A=1.D0
      GOTO 906
* FOR BAD INPUT DATA SET IERF TO 5 (A TO 0)
  903 IERF=5
  905 A=0.D0
  906 IF(ITYPE.EQ.2) T=0.D0
      write (6,*) 'ierf',ierf
      write (6,*) 'flash did not converge in ',IT,' iterations'
      write (6,*) 'errors'
      write (6,*) 'F = ',F,' FV = ',FV
      write (6,*) 'eps = ',eps,' epf = ',epf
      write (6,*) 'G = ',G
 1000 FORMAT ('FEED ENTHALPY = ',F12.5,X,'J/mole')
 1001 FORMAT ('FEED PRESSURE = ',F9.4,X,'bar',3X,'FEED TEMPERATURE = ',
     &        F8.4,X,'K')
 1002 FORMAT ('FINAL ENTHALPY = 'F12.5,X,'J/mole')
 1003 FORMAT ('FINAL PRESSURE = ',F9.4,X,'bar',3X,'FINAL TEMPERATURE = '
     &       ,F8.4,X,'K')
      STOP
      END


C=======================================================================
C
C     SUBROUTINE BUDET
C
C     THIS SUBROUTINE PERFORMS EITHER BUBBLE OR DEW-TEMPERATURE
C     CALCULATION USING NEWTON RAPHSON ITERATION FOR T.
C
C     MEANING OF W AND Z DEPENDS ON CHOICE OF CALCULATION
C        NTYPE=1    BUBBLE CALCULATION
C                   Z:  LIQUID PHASE
C                   W:  VAPOR PHASE
C        NTYPE=2    DEW CALCULATION
C                   Z:  VAPOR PHASE
C                   W:  LIQUID PHASE
C
C=======================================================================
      SUBROUTINE BUDET(NTYPE,P,T,W,Z,XK)

      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      DIMENSION W(15),Z(15),XK(15),CN(15)

      COMMON/CONST/C74,PI,SQR2,IPRINT,RK,NCOMP
C      COMMON/TYPE/ICALC
C
      N=NCOMP

C     INITIALIZE ITERATIVE PROCEDURE
C
      ITMX = 50
      DTMX = 10.D0
      EPS = 1.D-9
      DT=0.2D0
C      DT = 0.0001D0
      IT = 0
C
C     BEGIN ITERATIONS
C
 100  IT = IT + 1
      IF (IT .GT. ITMX) GO TO 900
      ILP = 0
 110  IF (NTYPE .EQ. 1) THEN
C       SUBROUTINE KFACTR(X,Y,T,P,VL,VG,K,IER)
        CALL KFACTR(Z,W,T,P,VL,VV,XK,IER)
      ELSE
        CALL KFACTR(W,Z,T,P,VL,VV,XK,IER)
      ENDIF
```

```
      S = 0.D0
      DO 140 I=1,N
      IF (NTYPE .EQ. 1) CN(I) = XK(I)*Z(I)
      IF (NTYPE .EQ. 2) CN(I) = Z(I)/XK(I)
 140  S = S + CN(I)
      IF (ILP .NE. 2) THEN
C
C     CHECK FOR PROXIMITY TO CRITICAL REGION
C
       DV = DABS(1.D0 - VL/VV)
       IF (ILP .NE. 1) THEN
C
C     FAR FROM CRITICAL, EXIT FROM INSIDE LOOP
C     CLOSE TO CRITICAL, REPEAT INSIDE LOOP
C
          IF (DV .LE. 0.75D0) THEN
             ILP = 1
             DO 150 I=1,N
 150          W(I) = CN(I)/S
             GO TO 110
          ENDIF
        ELSE
 160       IF (DV .LE. 0.55) THEN
C
C     VERY CLOSE TO CRITICAL, REPEAT INSIDE LOOP TWICE
C
             ILP = 2
             DO 170 I=1,N
 170          W(I) = CN(I)/S
             GO TO 110
          ENDIF
        ENDIF
      ENDIF
 180  CONTINUE
C
C     CALCULATE OBJECTIVE FUNCTION
C
      F = DLOG (S)
      IF (DABS(F) .LE. EPS) RETURN
      K=0
 200  CONTINUE
      K=K+1
      IF (K .GT. 200) THEN
        WRITE(6,201)
 201     FORMAT('BUDET: INTERNAL LOOP ITERATION COUNT EXCEEDED')
        RETURN
      ENDIF
      IF (NTYPE.EQ.1) CALL KFACTR(Z,W,T,P,VL,VV,XK,IER)
      IF (NTYPE.EQ.2) CALL KFACTR(W,Z,T,P,VL,VV,XK,IER)
      S1=0.D0
      DO 220 I=1,N
       IF (NTYPE.EQ.1) S1=S1+XK(I)*Z(I)
       IF (NTYPE.EQ.2) S1=S1+Z(I)/XK(I)
 220  CONTINUE
      F1=DLOG(S1)
C
C     CALCULATE NUMERICAL DERIVATIVES OF OBJECTIVE FUNCTION
C
      IF (NTYPE .EQ. 1) THEN
        CALL KFACTR(Z,W,T+DT,P,VL,VV,XK,IER)
      ELSE
        CALL KFACTR(W,Z,T+DT,P,VL,VV,XK,IER)
      ENDIF
      S2 = 0.D0
      DO 260 I=1,N
       IF (NTYPE .EQ. 1) THEN
          S2 = S2 + Z(I)*XK(I)
       ELSE
          S2 = S2 + Z(I)/XK(I)
       ENDIF
 260  CONTINUE
      F2 = DLOG(S2)
      DINVT = 1.D0/(T+DT) - 1.D0/T
      DET = (F2-F1)/DINVT
C
C     OBTAIN NEW ESTIMATE OF T USING NEWTON-RAPHSON ITERATION ON 1/T
C
      TINV = 1.D0/T - F1/DET
      TNEW = 1.D0/TINV
C
```

```
C     OPTIONAL WRITE STATEMENT
C
      DEL = TNEW - T
      IF (DABS(DEL) .LT. EPS) THEN
         DO 270 I=1,N
 270      W(I)=CN(I)/S
         GO TO 100
C
C     LIMIT NEW ESTIMATE OF ITERATE
C
      ELSE
         IF (DABS(DEL) .LT. DTMX) THEN
            T = T + DEL
            GO TO 200
         ELSE
            DC = DEL
            DEL = DTMX
            IF (DC .LT. 0.D0) DEL = -DEL
            T = T + DEL
            GO TO 200
         ENDIF
      ENDIF
      RETURN
 900  WRITE(6,910) ITMX
 910  FORMAT('MAXIMUM NUMBER OF ITERATIONS EXCEEDED IN BUDET. ITMX =',
     *I2)
      RETURN
      END
```

```
*************************************************************************
*                                                                       *
*           M I C H E L S E N     F L A S H     P R O G R A M           *
*                                                                       *
*                                                                       *
*   Copyright : Michael L. Michelsen,                                   *
*               Instituttet for Kemiteknik,                             *
*               Bygning 229, dth, DK 2800 Lyngby,                       *
*               Denmark                                                 *
*   Original  : MFLASH.FOR, August 1989                                 *
*                                                                       *
*   Editor    : Ronald J. Loggers                                       *
*   Date      : 31st October 1990                                       *
*   Place     : Departement of Applied Thermodynamics and Phasebehaviour,*
*               Faculty of Chemical Engineering and Material Science,   *
*               Delft University of Technology                          *
*               Julianalaan 136, 2628 BL Delft                          *
*               The Netherlands                                         *
*                                                                       *
*************************************************************************
C--------T---------T---------T---------T---------T---------T---------T---------T
C
      SUBROUTINE MICHELSENFLASH (IEOSTYP,N,XX,T,P,TCRIT,PCRIT,OMEGA,
     &    KIJ,NOPHASE,X,Y,VAPFRAC,LIQFRAC,INITFLAG,IWAL,zfact,zfrac)

      IMPLICIT REAL*8 (A-H,O-Z), INTEGER (I-N)

      REAL*8 LIQFRAC,KIJ

      COMMON/SKIP/NOTRY(15)
      COMMON/TYPEOS/MODTYP
      COMMON/THERMPROP/AUX(11)
      COMMON/FPROP/FUG(15),FUGX(15,15)

      DIMENSION FEED(15),PHASES(5),ZFACT(5),ZFRAC(15,5),X(15),Y(15)
      DIMENSION WKAREA(3000),IFTYP(15)
      DIMENSION TCRIT(15),PCRIT(15),OMEGA(15),KIJ(15,15),XX(15)
      LOGICAL SELECT(15)
c      OPEN(11,FILE='input.dat',STATUS='UNKNOWN')
c      OPEN(1,FILE='temp.out',STATUS='UNKNOWN')
c      OPEN(5,FILE='results.dat',STATUS='UNKNOWN')
c      OPEN(6,FILE='[.plot]mfdata.dat',STATUS='UNKNOWN')

      MODTYP=IEOSTYP
      NDEBUG=-1
      ITRIANGLE=0
      IF (INITFLAG.EQ.1) ITRIANGLE=1
      IF (ITRIANGLE.EQ.0) THEN
         NOFAS=0
         DO I=1,15
            DO J=1,5
               ZFRAC(I,J)=0.D0
               ZFACT(J)=0.D0
               PHASES(J)=0.D0
            ENDDO
         ENDDO
      ELSE
         NOFAS=NOPHASE
         IF (IWAL.EQ.0) THEN
            DO I=1,N
               ZFRAC(I,1)=X(I)
               ZFRAC(I,2)=Y(I)
               PHASES(1)=LIQFRAC
               PHASES(2)=VAPFRAC
            ENDDO
         ELSE
           DO I=1,N
              ZFRAC(I,1)=Y(I)
              ZFRAC(I,2)=X(I)
              PHASES(1)=VAPFRAC
              PHASES(2)=LIQFRAC
           ENDDO
         ENDIF
      ENDIF
      IF (NDEBUG.GT.0) OPEN (NDEBUG,FILE='debug',STATUS='UNKNOWN')
      CALL PUREDATA(N,MODTYP,NG,MODEL7,TCRIT,PCRIT,OMEGA,KIJ)
      DO I=1,N
         FEED(I)=XX(I)
      ENDDO
      DO 25 I=1,N
```

```
          SELECT(I)=.TRUE.
          IF (NOTRY(I).EQ.0) SELECT(I)=.FALSE.
   25 CONTINUE
      CALL XFLASH(MODTYP,ITRIANGLE,N,SELECT,T,P,FEED,NOFAS,
     &            PHASES,ZFRAC,ZFACT,IFTYP,NDEBUG,WKAREA)
      NOPHASE=NOFAS
      IF (ZFRAC(N,1).GT.ZFRAC(N,2)) THEN
         DO I=1,N
            X(I)=ZFRAC(I,1)
            Y(I)=ZFRAC(I,2)
            VAPFRAC=PHASES(2)
            LIQFRAC=PHASES(1)
            IWAL=0
         ENDDO
      ELSE
        DO I=1,N
           X(I)=ZFRAC(I,2)
           Y(I)=ZFRAC(I,1)
           VAPFRAC=PHASES(1)
           LIQFRAC=PHASES(2)
           IWAL=1
        ENDDO
      ENDIF
      RETURN
      END


C-----------------------------------------------------------------------
C     Subroutine  : PureData
C     Called from : SystemData
C     Calls       : SvCut, Const, Modin
C     Last modif. : 12th November 1990, rjl
C     Modifies    : C1,C2
C     Remarks     : Data input routine corresponds to conventional routine.
C                   Note : at the binary interaction parameters are not used
C                   at all
C-----------------------------------------------------------------------
      SUBROUTINE PUREDATA(NCOMP,ICEQ,NG,GEMODEL,tcrit,pcrit,omega,kij)
      IMPLICIT REAL*8(A-H,O-Z)
      PARAMETER(MAXC=15,MAXC2=(MAXC*(MAXC+1))/2)
      REAL*8 KIJ
      INTEGER GEMODEL
      COMMON /HEATCP/ CPVAL(MAXC,4)
      COMMON /CUB   / C,C1,C2
      COMMON /CRIT  / TCR(MAXC),PCR(MAXC),OMG(MAXC)
      COMMON /STYR  / N,MODEL7,NDER,NTEMP
      COMMON /OVER  / AC(MAXC),Q(MAXC),TSQR(MAXC)
      COMMON /PAR   / CK(MAXC2),BC(MAXC),AC0(MAXC),AC1(MAXC),AC2(MAXC)
      COMMON /BPARL / BCL(MAXC),XCUT,SCUT,ACUT,D1,D2
      COMMON /SKIP  / NOTRY(MAXC)
      COMMON /MATCO / CMC1(MAXC),CMC2(MAXC),CMC3(MAXC)
      COMMON /INTPAR/ GA(MAXC,MAXC),GB(MAXC,MAXC)

      DIMENSION TCRIT(15),PCRIT(15),OMEGA(15),KIJ(15,15)

      N=NCOMP
      IPRT = 0
      C=ICEQ
      CALL CONST(C,C1,C2,CONA,CONB)
      GEMODEL=0
      MODEL7=GEMODEL
C
      DO 100 I=1,N
C--- connection critical parameters ------------------------
         TCR(I)=TCRIT(I)
         PCR(I)=PCRIT(I)/1.d5
         OMG(I)=OMEGA(I)
C-----------------------------------------------------------
         CMC1(I)=0
         CMC2(I)=0
         CMC3(I)=0
         NOTRY(I) = 1
         CPVAL(I,1) = 5.D0
         DO 50 J=2,4
           CPVAL(I,J) = 0.D0
 50      CONTINUE
         NCI=0
C----- Calculate parameters a and b, a=sqrt(a)/R, b=b/R
         TSQR(I) = 1./DSQRT(TCR(I))
         BC(I) = CONB*TCR(I)/PCR(I)
```

```
         BCL(I) = DLOG(BC(I))
         AC(I) = CONA*TCR(I)/DSQRT(PCR(I))
C-----ICEQ=0: Soave-Redlich-Kwong
         Q(I)=.48+OMG(I)*(1.574-.176*OMG(I))
C-----ICEQ=1: Peng-Robinson
         IF (ICEQ.EQ.1) THEN
            Q(I)=.37464+OMG(I)*(1.54226-.26992*OMG(I))
         ENDIF
C-----NCI=1:  Matheas-Copeman
         IF (NCI.EQ.1) THEN
            Q(I)= CMC1(I)
         ELSE
            CMC2(I)=0
            CMC3(I)=0
         ENDIF
 100  CONTINUE
C-----Input matrix of interaction energies, in reduced units (a/r)
      DO 110 J=1,N
        DO 110 I=1,N
          GA(I,J)=1.D0-KIJ(I,J)
 110  CONTINUE
      DO 130 J=1,N
        DO 130 I=1,N
          GB(I,J)=1.D0
 130  CONTINUE
      MODEL=0
      RETURN
      END

C-----------------------------------------------------------------------
C     Subroutine  : Const
C     Called from : PureData
C     Calls       : None
C     Input       : C
C     Output      : C1, C2, AC, BC
C     Remarks     : Accurate calculation of constants for generic EOS (SRK/PR)
C                   i.e. a = ac*(R^2 * Tc^2)/Pc
C                   and  b = bc*(R*Tc)/Pc
C                   SRK (c=0): ac = 0.42747,  bc = 0.08664
C                   PR  (c=1): ac = 0.45724,  bc = 0.07780
C-----------------------------------------------------------------------
      SUBROUTINE CONST(C,C1,C2,AC,BC)
      IMPLICIT REAL*8(A-H,O-Z)
      U = 1 + C
      W = -C
      D = U*U - 4*W
      C2 = (U+DSQRT(D))/2
      C1 = W/C2
      S1 = 1 + ((U+W)*(U+3)-W)/2
      S2 = 1 + U + W
      R = DSQRT(S1*S1-S2*S2*S2)
      A = (S1+R)**(1.D0/3)
      B = S2/A
      X = A + B + 1
      BC = 1./(3*X+U-1)
      AC = BC*DSQRT(X*(X*X-3*W)-U*W)
      RETURN
      END
```

```
**************************************************************************
*  DATA BASE FOR THE JOULE-THOMSON EXPANSION; DATA COLLECTED             *
*  FROM THE DIPPR DATA BASE; CP COEFF FROM REID ET AL                    *
**************************************************************************
```

COMP.	MW	TC (K)	PC(BAR)	OMEGA	COEFF CP1-CP4 IDEAL GAS			
N2	28.013	126.2	33.9D5	0.039	3.115D1	-1.357D-2	2.680D-5	-1.168D-8
CO	28.010	132.9	35.0D5	0.066	3.087D1	-1.285D-2	2.789D-5	-1.272D-8
CO2	44.010	304.1	73.8D5	0.239	1.980D1	7.344D-2	-5.602D-5	1.715D-8
C1	16.043	190.4	46.0D5	0.011	1.925D1	5.213D-2	1.197D-5	-1.132D-8
C2	30.070	305.4	48.8D5	0.099	5.409D0	1.781D-1	-6.938D-5	8.713D-9
C3	44.094	369.8	42.5D5	0.153	-4.224D0	3.063D-1	-1.586D-4	3.215D-8
NC4	58.124	425.2	38.0D5	0.199	9.487D0	3.313D-1	-1.108D-4	-2.822D-9
IC4	58.124	408.2	36.5D5	0.183	-1.390D0	3.847D-1	-1.846D-4	2.895D-8
NC5	72.151	469.7	33.7D5	0.251	-3.626D0	4.873D-1	-2.580D-4	5.305D-8
IC5	72.151	460.4	33.9D5	0.227	-9.525D0	5.066D-1	-2.729D-4	5.723D-8
NC6	86.178	507.5	30.1D5	0.299	-4.413D0	5.820D-1	-3.119D-4	6.494D-8
NC7	100.205	540.3	27.4D5	0.349	-5.146D0	6.762D-1	-3.651D-4	7.658D-8
NC8	114.232	568.8	24.9D5	0.398	-6.096D0	7.712D-1	-4.195D-4	8.855D-8
NC10	142.286	617.7	21.2D5	0.489	-7.931D0	9.609D-1	-5.288D-4	1.131D-7
NC12	170.340	658.2	18.2D5	0.575	-9.328D0	1.149D0	-6.347D-4	1.359D-7
NC15	212.421	707.0	15.2D5	0.706	-11.92D0	1.433D0	-7.972D-4	1.720D-7
NC20	298.555	767.0	11.1D5	0.907	-22.38D0	1.939D0	-1.117D-3	2.528D-7
NC24	338.659	818.0	9.048D5	0.942	-18.928	2.2838	-1.282D-3	2.744D-7

```
**************************************************************************
*   DATA BASE USED FOR THE JOULE-THOMSON EXPANSION OF GAS-CONDENSATES  *
*   CRITICAL CONSTANTS, ACENTRIC FACTORS AND IDEAL GAS HEAT CAPACITIES *
**************************************************************************

     TC (K)  PC(Pa)    OMEGA       CP COEFFICIENTS

N2   126.20  34.0D5    0.0377215  3.115D1  -1.357D-2 2.680D-5  -1.168D-8
CO2  304.21  73.83D5   0.2236210  1.980D1   7.344D-2  -5.602D-5 1.715D-8
C1   190.58  46.04D5   0.0107665  1.925D1   5.213D-2  1.197D-5  -1.132D-8
C2   305.42  48.8D5    0.098955   5.409D0   1.718D-1  -6.938D-5 8.713D-9
C3   369.82  42.49D5   0.1530710  -4.224D0  3.063D-1  -1.586D-4 3.215D-8
NC4  425.18  37.97D5   0.1992750  9.487D0   3.313D-1  -1.108D-4 -2.822D-9
IC4  408.14  36.48D5   0.1807710  -1.390D0  3.847D-1  -1.846D-4 2.895D-8
NC5  469.65  33.69D5   0.249638   -3.626D0  4.873D-1  -2.580D-4 5.305D-8
IC5  460.43  33.81D5   0.227461   -9.525D0  5.066D-1  -2.729D-4 5.723D-8
NC6  507.43  30.12D5   0.304586   -4.413D0  5.820D-1  -3.119D-4 6.494D-8

**********************************************************************
*   COMPOSITION (MOLE FRACTIONS) AND MOLECELUR WEIGHT (MW;G/MOLE)    *
*   AND DENSITIES (SG;G/CM3 AT 15 C AND 1 ATM.) OF THE PLUS          *
*   FRACTIONS FOR A TYPICAL GAS-CONDENSATE                           *
**********************************************************************

TOTAL NUMBER OF FRACTIONS:
14

      MOL FR.  MW   SG

N2    0.0071
CO2   0.0864
C1    0.7085
C2    0.0853
C3    0.0495
NC4   0.0126
IC4   0.0075
NC5   0.0040
IC5   0.0041
NC6   0.0046
C7    0.0061   91    0.820
C8    0.0071   105   0.777
C9    0.0039   120   0.803
C10   0.0133   222   0.853

**********************************************************************
*   INPUT OF BINARY INTERACTION COEFFICIENTS USED IN MIXING RULES    *
*   FOR HYDROCARBONS WITH NITROGEN AND CARBONDIOXIDE; HYDROCARBON-   *
*   HYDROCARBON INTERACTIONS AR ZERO; EOS USED: SRK & PR             *
**********************************************************************

SRK    N2       CO2

N2     0.0      -0.032
CO2    -0.032   0.0
C1     0.028    0.093
C2     0.041    0.136
C3     0.076    0.129
NC4    0.070    0.143
IC4    0.094    0.128
NC5    0.088    0.131
IC5    0.087    0.131
NC6    0.150    0.118
C7+    0.142    0.110

PR     N2       CO2

N2     0.0      -0.017
CO2    -0.017   0.0
C1     0.031    0.092
C2     0.052    0.132
C3     0.085    0.124
NC4    0.080    0.133
IC4    0.103    0.120
NC5    0.100    0.122
IC5    0.092    0.122
NC6    0.150    0.110
C7+    0.144    0.100
```

```
**************************************************************************
*   DATA BASE USED FOR THE JOULE-THOMSON EXPANSION OF GAS-CONDENSATES    *
*   CRITICAL CONSTANTS, ACENTRIC FACTORS AND IDEAL GAS HEAT CAPACITIES   *
**************************************************************************

TOTAL NUMBER OF FRACTIONS:
24
NUMBER OF PLUS FRACTIONS:
14

        MW       TC (K)  PC(Pa)   OMEGA    CP COEFFICIENTS

N2      28.013   126.2   33.9D5   0.039     3.115D1 -1.357D-2  2.680D-5 -1.168D-8
CO2     44.010   304.1   73.8D5   0.239     1.980D1  7.344D-2 -5.602D-5  1.715D-8
C1      16.043   190.4   46.0D5   0.011     1.925D1  5.213D-2  1.197D-5 -1.132D-8
C2      30.070   305.4   48.8D5   0.099     5.409D0  1.781D-1 -6.938D-5  8.713D-9
C3      44.094   369.8   42.5D5   0.153    -4.224D0  3.063D-1 -1.586D-4  3.215D-8
NC4     58.124   425.2   38.0D5   0.199     9.487D0  3.313D-1 -1.108D-4 -2.822D-9
IC4     58.124   408.2   36.5D5   0.183    -1.390D0  3.847D-1 -1.846D-4  2.895D-8
NC5     72.151   469.7   33.7D5   0.251    -3.626D0  4.873D-1 -2.580D-4  5.305D-8
IC5     72.151   460.4   33.9D5   0.227    -9.525D0  5.066D-1 -2.729D-4  5.723D-8
NC6     86.178   507.5   30.1D5   0.299    -4.413D0  5.820D-1 -3.119D-4  6.494D-8

**************************************************************************
*   COMPOSITION (MOLE FRACTIONS) AND MOLECELUR WEIGHT (MW;G/MOLE)        *
*   AND DENSITIES (SG;G/CM3 AT 15 C AND 1 ATM.) OF THE PLUS              *
*   FRACTIONS FOR A TYPICAL GAS-CONDENSATE                               *
**************************************************************************

        MOL FR.   MW     SG

N2      0.0012
CO2     0.0249
C1      0.7643
C2      0.0746
C3      0.0312
NC4     0.0121
IC4     0.0059
NC5     0.0059
IC5     0.0050
NC6     0.0079
C7      0.0095   95.0    0.726
C8      0.0108   106.0   0.747
C9      0.0078   116.0   0.769
C10     0.00592  133.0   0.781
C11     0.00467  152.0   0.778
C12     0.00345  164.0   0.785
C13     0.00375  179.0   0.802
C14     0.00304  193.0   0.815
C15     0.00237  209.0   0.817
C16     0.00208  218.0   0.824
C17     0.00220  239.0   0.825
C18     0.00169  250.0   0.831
C19     0.00140  264.0   0.841
C20+    0.00833  377.0   0.873

6
4 5
6 10
11 13
14 17
18 23
24 24
C2-C3
C4-C6
C7-C9
C10-C13
C14-C19
C20+
```