Development of Robust Trust-Region Algorithms for Multiphase Equilibrium Calculations

by

Lingfei Xu

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Petroleum Engineering

Department of Civil and Environmental Engineering
University of Alberta

ABSTRACT

Crude oil, as a major type of fossil fuels, is a mixture mainly made up of hydrocarbons. One important research topic relevant to the extraction, transportation, and storage of these hydrocarbon mixtures is the study of multiphase equilibria under specified specifications, especially the pressure-temperature (PT) specification. In the CO_2 injection process for enhanced oil recovery (EOR), the vapor-liquid-liquid (VLL) equilibria, consisting of a vapor phase (V), a solvent-rich liquid phase (L_2), and an oil-rich liquid phase (L_1), are frequently encountered. A robust PT equilibrium calculation algorithm, which quantitively determines the number of phases, phase fractions, and phase compositions, is a fundamental module that is required in the reliable compositional simulations of multiphase flows. However, the development of such an algorithm is challenging, because detecting the second liquid hydrocarbon phase can be a challenging task. To address this issue, Pan et al. (2019) proposed a robust multiphase (up to three phases) equilibrium calculation framework, where the state-of-the-art trust-region solvers developed by Petitfrere and Nichita (2014) were embedded into their framework. Such a framework consists of a series of implementation of two key subroutines, namely stability test (Michelsen, 1982a) and flash calculation (Michelsen, 1982b). The new trust-region-based algorithms (Petitfrere and Nichita, 2014) are successfully applied in both stability test and flash calculation to address the convergence difficulties that may be encountered when running the conventional Newton-based method.

Inspired by the trust-region method applied in the flash calculation algorithms (Petitfrere and Nichita, 2014; Pan et al., 2019), we build a new hybrid trust-region-based flash calculation algorithm based on a different formulation. In the new formulation, the logarithms of equilibrium ratios (lnK) and phase fractions are the independent variables that are solved simultaneously. Advantages of such a simultaneous solution strategy include that the explicit solution of the phase fractions (i.e., solving the Rachford-Rice equation; Rachford and Rice, 1952) is not needed in each iteration and the well-scaled variable lnK is treated as the independent variable. We build a new robust equilibrium calculation framework embedding the new flash calculation algorithm as a subroutine. Case studies indicate that the new hybrid flash calculation can generate digital multiphase phase diagrams in a more robust and efficient manner than the conventional approach.

Besides, considering that the conventional Newton-based method is still the major solution method for envelope constructions, we propose new trust-region-based phase envelope construction algorithms aiming to improve the computational efficiency and robustness. A trust-region-based algorithm is first proposed for two-phase envelope constructions. Example calculations show that the new construction algorithm is more cost-effective and robust than the conventional Newton-based algorithm. We then extend this base algorithm to three-phase envelope constructions. More specifically, the extended algorithm handles the calculation of multiphase envelopes involving two-phase envelopes, three-phase envelopes, and three-phase points (i.e., the intersection point of a two-phase envelope and a three-phase region). We successfully construct a total of 15 multiphase envelopes for 15 mixtures (including hydrocarbon-CO_2, hydrocarbon-water, hydrocarbon-asphaltene mixtures), without encountering a single failure. We also observe that the new trust-region-based algorithm significantly enhances the computational efficiency of multiphase envelope constructions.

PREFACE

A version of **CHAPTER 2** is submitted to *Industrial & Engineering Chemistry Research* for possible publication on Feb 23, 2023. Xu, L. is responsible for the concept formation, theoretical development, simulation results, analysis, and manuscript composition. Li, S. is responsible for hydrocarbon fluid property collection/generation and manuscript composition. Li, H. is the supervisory author and gets involved in theoretical development, analysis, and manuscript revision.

A version of **CHAPTER 3** is submitted to *Journal of Petroleum Science and Engineering* for possible publication on Jan 20, 2023. Xu, L. is responsible for the theoretical development, simulation results, analysis, and manuscript composition. Li, H. is the supervisory author and gets involved in the concept formation, theoretical development, analysis, and manuscript revision.

A version of **CHAPTER 4** is submitted to *Chemical Engineering Science* for possible publication on Feb 1, 2023. Xu, L. is responsible for the theoretical development, simulation results, analysis, and manuscript composition. Chen, Z. is responsible for hydrocarbon fluid property collection/generation and manuscript composition. Li, S. is responsible for the four-phase equilibrium calculations. Li, H. is the supervisory author and gets involved in the concept formation, theoretical development, analysis, and manuscript revision.

DEDICATION

To my parents, Dongsheng Xu and Xianfeng Yan,

and to Sirui Li.

ACKNOWLEDGMENTS

Words cannot express my gratitude to my supervisor Dr. Huazhou (Andy) Li. I would not have accomplished the present works without his help and guidance. I thank my committee members, Dr. Ergun Kuru, Dr. Nobuo Maeda, Dr. Hongbo Zeng, and Dr. Yong Li for their comments and suggestions on my PhD thesis. I am also grateful to Dr. Dan Nichita for serving as the external examiner and providing constructive comments and suggestions.

I also wish to thank the following individuals or organizations for supporting my PhD program:

- Natural Sciences and Engineering Research Council (NSERC) of Canada for a Discovery Grant to Dr. H. Li;
- The China Scholarship Council (CSC) for the financial support;
- The past and present group members in Dr. Li's research group; and
- All my friends for their help and friendship.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Research background

Three-phase equilibria, such as the vapor-liquid-liquid (VLL) equilibria for hydrocarbon-CO_2 systems, vapor-liquid-water (VLW) equilibria for hydrocarbon-water systems, and vapor-liquid-asphaltene (VLA) for hydrocarbon-asphaltene systems, are frequently encountered in petroleum and chemical engineering applications. Such three-phase equilibria may appear when injecting CO_2 for enhanced oil recovery (EOR) and carbon sequestration in low-temperature reservoirs. **Fig. 1-1.** illustrates the typical process of a water- alternating-CO_2 injection operation. In such a process, the multiple contacts of oil and CO_2 lead to the formation of one-phase, two-phase, and three-phase equilibria. In compositional simulation of such EOR process, phase equilibrium calculation serves as the workhorse to determine the correct number of phases, phase fractions, and phase compositions.

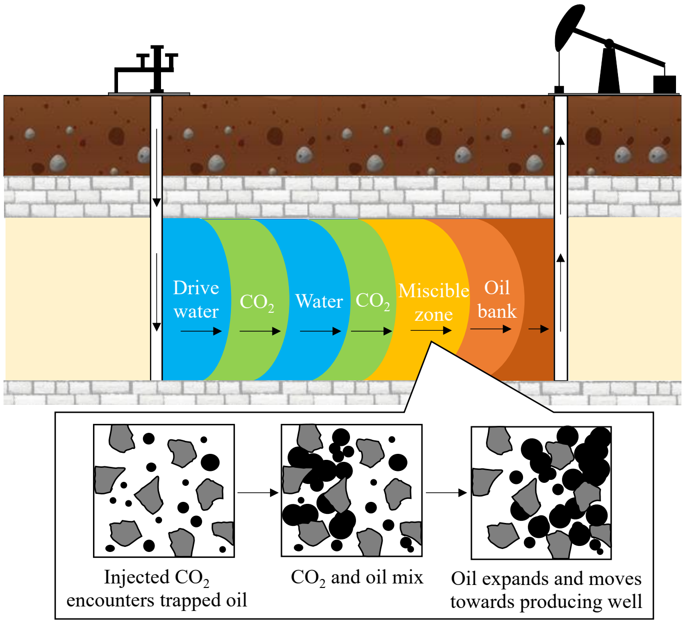
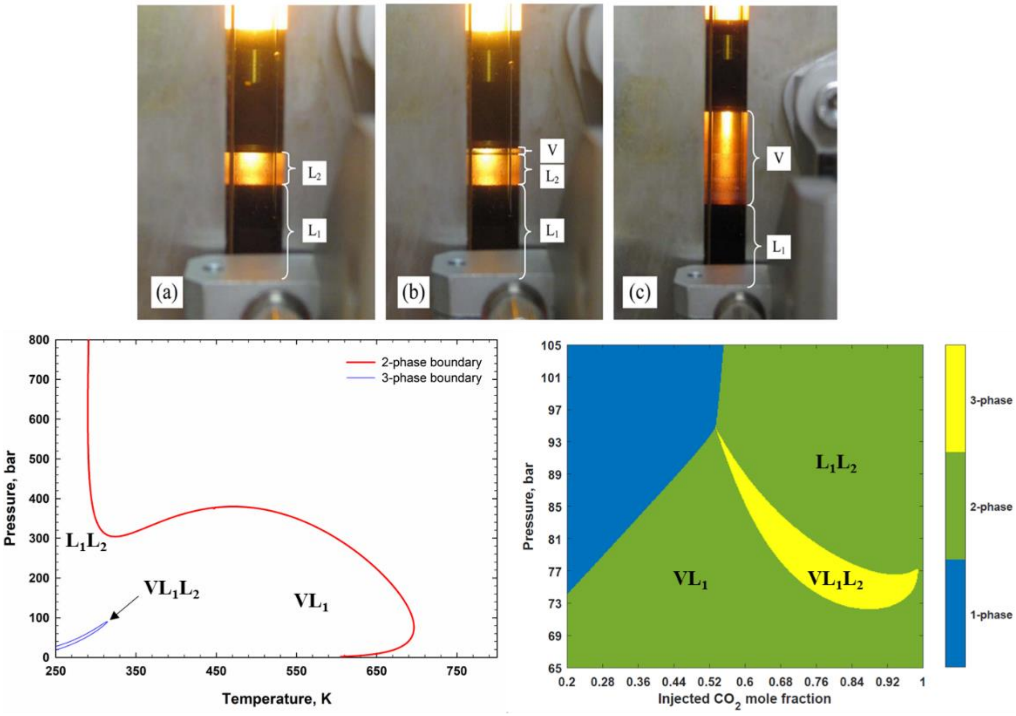


Fig. 1-1. Illustration of a CO_2 injection process for EOR (adapted from NETL (2010)).

Fig. 1-2 further illustrates the digital images of three typical equilibria for a CO_2-inclusive mixture. Specifically, Fig. 1-2a, b, and c illustrate the liquid-liquid (LL) two-phase equilibrium, VLL three-phase equilibrium, and vapor-liquid (VL) two-phase equilibrium, respectively. Then Fig. 1-2d and e show the pressure-temperature (PT) and pressure-composition (Px) diagrams which indicate possible conditions where three-phase VLL equilibria may appear. In summary, the VLL equilibria will be encountered in a variety of PT and Px conditions. From a computational aspect, however, the calculation of the VLL equilibria is challenging as detecting the second liquid hydrocarbon phase is difficult. Thus, a reliable multiphase (up to three phases) equilibrium calculation algorithm is required to capture the three-phase equilibria of such CO_2-inclusive mixtures.



(d) (e)

Fig. 1-2. The digital images of three possible equilibria for one CO_2-inclusive mixture captured by Li et al. (2013), a typical PT phase envelope diagram for a CO_2-inclusive oil mixture, and a pressure-composition diagram for a CO_2-inclusive oil mixture. (a), (b), and (c) show a LL two-phase equilibrium, a VLL three-phase equilibrium, and a VL two-phase equilibrium, respectively, for a C_3H_8-CO_2-heavy oil mixture with the composition (83.2 mol% CO_2, 11.8 mol% C_3H_8, and 5.0 mol% heavy oil); (d) is the PT phase envelope diagram for Oil G (Khan et al., 1992) mixed with 70 mol% CO_2; (e) is the PX diagram for Oil G (Khan et al., 1992) mixed with CO_2.

Based on Michelsen (1982a, b), the multiphase equilibrium calculation is a numerical algorithm containing stepwise executions of two algorithm subroutines, namely, stability test and flash calculation. For example, a three-phase PT equilibrium calculation starts with testing the stability of the given overall feed. If the feed composition is not stable, a two-phase flash is implemented. Then the stability of one of the two phases is tested. If instability is observed again, a three-phase flash calculation is performed, and the phase fractions and phase compositions in the three-phase equilibrium are determined.

The two-phase flash calculation, as a basic module in the numerical algorithm of multiphase equilibrium calculations, is briefly introduced as follows. At a specified pressure P and temperature T, one unit mole of feed (z) consisting of nc components is given. Assuming the feed z will split into a vapor phase and a liquid phase at the specified PT specification, we define the mole fractions of component i in the feed, in the liquid phase, and in the vapor phase as z_i, x_i, and y_i, respectively. Then we obtain the following relations (Firoozabadi, 2015):

1) Fugacity (*f*) equality:

$$f_{x,i}(T,P,x_1,\ldots,x_{nc}) = f_{y,i}(T,P,y_1,\ldots,y_{nc}),\ i = 1,2,\cdots,nc \tag{1-1}$$

where nc is the total number of components.

2) Material balance:

$$z_i = (1-\beta)x_i + \beta y_i \tag{1-2}$$

where β is the vapor phase fraction.

3) Physical constraints:

$$\sum_{i=1}^{nc} x_i = 1 \tag{1-3}$$

$$\sum_{i=1}^{nc} y_i = 1 \tag{1-4}$$

We then define a new variable called equilibrium ratio (K) as,

$$K_i = y_i/x_i \tag{1-5}$$

Eliminating y_i in **Eq. 1-2** with **Eq. 1-5**, we obtain,

$$x_i = \frac{z_i}{1+\beta(K_i-1)} \tag{1-6}$$

Applying Eq. 1-5 again, we obtain,

$$y_i = \frac{z_i K_i}{1+\beta(K_i-1)} \tag{1-7}$$

Combining **Eqs. 1-3**, **1-4**, **1-6**, and **1-7**, we obtain the Rachford-Rice (RR) equation (Rachford and Rice, 1952) as follows,

$$\sum_{i=1}^{nc} \frac{(K_i-1)z_i}{1+\beta(K_i-1)} = 0 \tag{1-8}$$

The two-phase flash calculation problem is solved for finding K and β that satisfy **Eqs. 1-1** and **1-8**. Conventionally, we handle K and β separately under the ideal-mixture assumption (Michelsen and Mollerup, 2007). The corresponding solution method is the successive substitution (SS) method, which is based on solving a set of nonlinear equations. In the SS method, an inner loop is used to solve the RR equation with constant K, while an outer loop is used for updating K. In the outer loop, the updating equation of K (based on the equivalent transformation of Eq. 1-1) is given by (Merha et al., 1983),

$$(K_i)^{new} = (K_i)^{old}(f_{x,i}/f_{y,i}),\ i = 1,2,\cdots,nc \tag{1-9}$$

The initial values of K can be generated using the Wilson's correlation (1969). Before 1980s, the SS method (with Newton-Raphson method for solving the RR equation) is the dominant solution method to solve the two-phase flash problem (Michelsen and Mollerup, 2007). The major advantages of the SS method include the guaranteed robustness and the derivative-free calculations. In the literature, there are advanced solution methods which follow the idea of equation solving, such as the accelerated SS method (Michelsen, 1982b; Heidemann, 1983; Mehra et al., 1983), lnK methods based on the direct solution of lnK using Newton-based methods (Michelsen, 1982b; Nghiem and Li, 1984; Ammar and Renon, 1987), and the improved lnK method where both lnK and β are treated as the independent variables (Haugen et al., 2011; Petitfrere and Nichita, 2016).

However, the equation-solving approach cannot guarantee the decrease of the Gibbs free energy as iteration proceeds (Okuno, 2009). Such an issue can be addressed by the minimization approach, where the dimensionless Gibbs free energy function is minimized with an available second-order method (Michelsen, 1982b). For two-phase equilibria, this function is given by (Pan et al., 2019),

$$G = \sum_{i=1}^{nc} (n_{x,i} \ln{(f_{x,i})} + n_{y,i} \ln{(f_{y,i})}) \tag{1-10}$$

where $n_{x,i}$ and $n_{y,i}$ are the molar amount of component i in the liquid phase and in the vapor phase, respectively. In **Eq. 1-10**, we define n_y as the independent variables. In this way, the dependent variables (i.e., n_x) are given by,

$$n_{x,i} = z_i - n_{y,i} \tag{1-11}$$

In fact, Michelsen (1982b) suggested applying several SS iterations first and then switching to the second-order Newton's method to reach tight convergence. Perschke (1988) implemented such an algorithm combining SS and Newton's methods. In the Newton's algorithm of Perschke (1988), the modified Cholesky decomposition method (Gill and Murray, 1974) guarantees that the Hessian matrix is positive definite. Ideally, such a Newton's algorithm, which deals with an unconstrained local minimization problem, can provide a fast convergence within several iterations. There are also global minimization methods documented in the literature, such as the simulated annealing method (Pan and Firoozabadi, 1998) and the tunnelling method (Nichita et al., 2002).

In the flash calculation problem, however, it is only assumed but not rigorously proved that the fluid at the specified conditions will split into two or more phases. Moreover, the initial values of the equilibrium ratios (K) are only generated by empirical correlations (i.e., the Wilson's correlation, 1969). Therefore, a flash calculation algorithm may converge to an unphysical or trivial solution. To address this problem, Michelsen (Michelsen, 1982a) suggested implementing the stability test to rigorously prove if a given feed will split at a specified condition. The solution methods for the stability test problem are similar to those for the flash calculation problem. That is, the stability test problem can be solved by the SS method/accelerated SS method (Michelsen, 1982a), Newton-based equation-solving methods (Nghiem and Li, 1984; Firoozabadi, 2015), or second-order minimization methods (Michelsen, 1982a; Perschke, 1988; Hoteit and Firoozabadi, 2006; Petitfrere and Nichita, 2014; Pan et al., 2019). By solving the stability test problem, we can obtain a trial phase to be fed into a flash calculation algorithm as one of the input variables. Michelsen (1982a, b) suggested applying a stability test algorithm and a flash calculation algorithm in a stepwise manner. This stepwise design is the key for building robust equilibrium calculation algorithms, especially multiphase ones.

Throughout the development of equilibrium calculation algorithms, the second-order Newton's method (for both the stability tests and flash calculations) remains an important solution method. However, the convergence problems encountered in the VLL equilibria calculations for low-temperature hydrocarbon-CO_2 mixtures still cannot be fully addressed until the development of the robust trust-region-based algorithms by Petitfrere and Nichita

(2014). Based on these algorithms, Pan et al. (2019) proposed one state-of-the-art multiphase equilibrium calculation framework, where a sequential workflow is well designed for both stability tests and flash calculations. The workflow is a consecutive execution of three numerical solution methods: the classical SS method gives an acceptable initial convergence; Newton's method provides a rapid convergence once the iteration point gets close to the solution; finally, the trust-region method manages special cases where Newton's method encounters infeasible conditions. Such framework is then tested by constructing high-resolution digital Px diagrams for a total of 9 CO_2-inclusive mixtures (Pan et al., 2019). No single failure is encountered during the calculations.

Inspired by the works by Petitfrere and Nichita (2014) and Pan et al. (2019), this study aims to apply the trust-region method in other types of equilibrium calculations where such a robust and efficient solution method is in urgent demand to improve the computational efficiency and robustness. To begin with, we would develop a new trust-region-based flash calculation algorithm for the improved lnK flash calculation approach (Haugen et al., 2011; Petitfrere and Nichita, 2016). Next, we would develop a robust and cost-effective trust-region-based two-phase envelope construction algorithm. Finally, we would extend the two-phase construction algorithm to handle three-phase equilibria.

1.2 Problem statement

Several technical problems related to the robustness of multiphase equilibrium calculations and phase envelope constructions are yet to be solved. Specifically, this thesis is focused on the following research problems:

- In several cases of VLL equilibria (such as in the near-critical region or on the phase boundary), the improved lnK flash calculation method (Haugen et al, 2011; Li and Firoozabadi, 2012; Petitfrere and Nichita, 2016), where both equilibrium ratios and phase fractions are independent variables, may not converge to the correct solutions. In other words, such algorithms may fail to find a three-phase solution with the conventional approach combining SS and Newton's methods. This requires us to introduce a robust trust-region algorithm to address such convergence difficulties.

- The algorithm used to construct a PT phase envelope is first developed by Michelsen (1980). Michelsen's algorithm has been further extended to different scenarios (Pederson et al., 1996; Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018), but the classical Newton's method has been followed without change for decades. Up to now, we are still lacking a new solution method that can provide more robust and efficient phase envelope calculations than Newton's method. Therefore, introducing the robust trust-region method will be promising to improve the computational efficiency and robustness of the envelope construction algorithms.
- Three-phase equilibria are not considered in Michelsen (1980). It might be a nontrivial task to locate a three-phase region. In fact, constructing such a multiphase envelope requires not only the formulations of two-phase envelope points, three-phase envelope points, and three-phase points, but also an integrated strategy to traverse all the two-phase and three-phase envelope branches (Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018; Cismondi, 2018). It is likely that, for constructing multiphase envelopes, especially three-phase envelopes, a trust-region-based algorithm will outperform the

conventional Newton-based algorithm in terms of robustness and efficiency.

1.3 Hypothesis

The central hypothesis of the thesis is that the trust region method, when properly combined with SS or Newton's methods, will lead to more robust and efficient multiphase equilibrium calculations and phase envelope calculations (including PT and Px envelope constructions).

1.4 Objectives

The objective of this research is to develop robust numerical algorithms for phase equilibrium calculations and phase envelope constructions. The proposed algorithms are expected to provide more robust and cost-effective computations for the two-phase and three-phase equilibria of reservoir fluids based on Peng-Robison equation of state (PR-EOS) (Peng and Robinson, 1976; Robinson and Peng, 1978). The short-term and long-term objectives are listed as below:

- Develop a trust-region-based flash calculation algorithm with the formulation that treats both equilibrium ratios and phase fractions as independent variables. Such an algorithm is expected to handle the challenging flash calculations where the combined use of SS and Newton's methods cannot converge.
- Develop a trust-region-based algorithm for two-phase envelope construction. We aim to introduce the trust-region method for envelope point calculations, which is expected to be more robust and cost-effective than the conventional Newton's method. Such a trust-region algorithm will be capable of constructing two-phase PT and Px envelopes.
- Develop a trust-region-based algorithm for three-phase envelope constructions.

It is expected that the new trust-region-based algorithm can build phase envelopes with two-phase and three-phase branches in a robust and efficient manner.

1.5 Thesis structure

This is a paper-based thesis. Five chapters are presented in this thesis and organized as follows:

CHAPTER 1 introduces the basic research background, the problem statement, and the major research objectives. In **CHAPTER 2**, a new hybrid flash calculation algorithm combining the trust-region and Newton-Raphson methods is developed to solve equilibrium ratios and phase fractions simultaneously. The new hybrid algorithm can provide robust and efficient multiphase flash calculations. In **CHAPTER 3**, a trust-region-based phase envelope construction algorithm is developed. The developed algorithm can further improve the computational efficiency and robustness of two-phase envelope constructions. In **CHAPTER 4**, a trust-region-based algorithm for three-phase envelope constructions is proposed. Such an algorithm is cost-effective and robust in constructing phase envelopes with two-phase and three-phase branches for hydrocarbon-water, hydrocarbon-CO_2, and hydrocarbon-asphaltene mixtures. **CHAPTER 5** summarizes the conclusions reached in the thesis and the recommendations for future works.

References

Agger, C.S., Sørensen, H., 2018. Algorithm for constructing complete asphaltene PT and Px phase diagrams. Ind. Eng. Chem. Res. 57, 392–400. https://doi.org/10.1021/acs.iecr.7b04246

Ammar, M.N., Renon, H., 1987. The isothermal flash problem: new methods for phase split calculations. AIChE J. 33, 926–939. https://doi.org/10.1002/aic.690330606

Cismondi, M., 2018. Phase envelopes for reservoir fluids with asphaltene onset lines: an integral computation strategy for complex combinations of two- and three-phase behaviors. Energy Fuels 32, 2742–2748. https://doi.org/10.1021/acs.energyfuels.7b02790

Firoozabadi, A., 2015. Thermodynamics and Applications of Hydrocarbons Energy Production, McGraw Hill Professional.

Gill, P.E., Murray, W., 1974. Newton-type methods for unconstrained and linearly constrained optimization. Math. Program. 7, 311–350. https://doi.org/10.1007/BF01585529

Haugen, K.B., Firoozabadi, A., Sun, L., 2011. Efficient and robust three-phase split computations. AIChE J. 57, 2555–2565. https://doi.org/10.1002/aic.12452

Heidemann, R.A., 1983. Computation of high pressure phase equilibria. Fluid Phase Equilib. 14, 55–78. https://doi.org/10.1016/0378-3812(83)80115-0

Hoteit, H., Firoozabadi, A., 2006. Simple phase stability-testing algorithm in the reduction method. AIChE J. 52, 2909–2920. https://doi.org/10.1002/aic.10908

Khan, S.A., Pope, G.A., Sepehrnoori, K., 1992. Fluid characterization of three-phase CO_2/oil mixtures. Presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, Oklahoma, USA, 22–24 April. https://doi.org/10.2118/24130-MS

Li, H., Yang, D., Li, X., 2013. Determination of three-phase boundaries of solvent(s)–CO_2–heavy oil systems under reservoir conditions. Energy Fuels 27, 145–153. https://doi.org/10.1021/ef301549a

Li, Z., Firoozabadi, A., 2012. General strategy for stability testing and phase-split calculation in two and three phases. SPE J. 17, 1096–1107. https://doi.org/10.2118/129844-PA

Lindeloff, N., Michelsen, M.L., 2003. Phase envelope calculations for hydrocarbon-water mixtures. SPE J. 8, 298–303. https://doi.org/10.2118/85971-PA

Mehra, R.K., Heidemann, R.A., Aziz, K., 1983. An accelerated successive substitution algorithm. Can. J. Chem. Eng. 61, 590–596. https://doi.org/10.1002/cjce.5450610414

Michelsen, M.L., 1980. Calculation of phase envelopes and critical points for multicomponent mixtures. Fluid Phase Equilib. 4, 1–10. https://doi.org/10.1016/0378-3812(80)80001-X

Michelsen, M.L., 1982a. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. https://doi.org/10.1016/0378-3812(82)85001-2

Michelsen, M.L., 1982b. The isothermal flash problem. Part II. Phase-split calculation. Fluid Phase Equilib. 9, 21–40. https://doi.org/10.1016/0378-3812(82)85002-4

Michelsen, M.L., Mollerup, J.M., 2007. Thermodynamic Models: Fundamentals & Computational Aspects, 2nd ed, Tie-Line Publications, Holte, Denmark.

NETL, 2010. Carbon Dioxide Enhanced Oil Recovery: Untapped Domestic Energy Supply and Long Term Carbon Storage Solution, US Department of Energy.

Nghiem, L.X., Li, Y.-K., 1984. Computation of multiphase equilibrium phenomena with an equation of state. Fluid Phase Equilib. 17, 77–95. https://doi.org/10.1016/0378-3812(84)80013-8

Nichita, D.V., Gomez, S., Luna, E., 2002. Multiphase equilibria calculation by direct minimization of Gibbs free energy with a global optimization method. Comput. Chem. Eng. 26, 1703–1724. https://doi.org/10.1016/S0098-1354(02)00144-8

Okuno, R., 2009. Modeling of Multiphase Behavior for Gas Flooding Simulation. Ph.D. thesis, University of Texas at Austin.

Pan, H., Connolly, M., Tchelepi, H., 2019. Multiphase equilibrium calculation framework for compositional simulation of CO_2 injection in low-temperature reservoirs. Ind. Eng. Chem. Res. 58, 2052–2070. https://doi.org/10.1021/acs.iecr.8b05229

Pan, H., Firoozabadi, A., 1998. Complex multiphase equilibrium calculations by direct minimization of gibbs free energy by use of simulated annealing. SPE Reserv. Eval. Eng. 1, 36–42. https://doi.org/10.2118/37689-PA

Pedersen, K.S., Michelsen, M.L., Fredheim, A.O., 1996. Phase equilibrium calculations for unprocessed well streams containing hydrate inhibitors. Fluid Phase Equilib. 126, 13–28. https://doi.org/10.1016/S0378-3812(96)03142-1

Peng, D.-Y., Robinson, D.B., 1976. A new two-constant equation of state. Ind. Eng. Chem. Fund. 15, 59–64. https://doi.org/10.1021/i160057a011

Perschke, D.R., 1988. Equation of State Phase Behavior Modeling for Compositional Simulatio. Ph.D. thesis, University of Texas at Austin.

Petitfrere, M., Nichita, D.V., 2014. Robust and efficient trust-region based stability analysis and multiphase flash calculations. Fluid Phase Equilib. 362, 51–68. https://doi.org/10.1016/j.fluid.2013.08.039

Petitfrere, M., Nichita, D.V., 2016. On a choice of independent variables in Newton iterations for multiphase flash calculations. Fluid Phase Equilib. 427, 147–151. https://doi.org/10.1016/j.fluid.2016.06.050

Rachford, H.H., Jr., Rice, J.D., 1952. Procedure for use of electronic digital computers in calculating flash vaporization hydrocarbon equilibrium. J. Pet. Tech. 4, 19–3. https://doi.org/10.2118/952327-G

Robinson, D.B., Peng, D., 1978. The Characterization of the Heptanes and Heavier Fractions for the GPA Peng-Robinson Programs, Gas Processors Association, Tulsa, Oklahoma.

Wilson, G.M., 1969. A modified Redlich-Kwong equation of state, application to general physical data calculations. Presented at the 65th National AIChE Meeting, Cleveland, OH, USA.

CHAPTER 2 A HYBRID MULTIPHASE FLASH CALCULATION ALGORITHM BASED ON TRUST-REGION AND NEWTON-RAPHSON METHODS FOR THE SIMULTANEOUS SOLUTION OF EQUILIBRIUM RATIOS AND PHASE FRACTIONS

A version of this chapter is submitted to *Industrial & Engineering Chemistry Research* for possible publication on Feb 23, 2023.

Abstract

A robust flash calculation algorithm can increase the possibility of correctly performing multiphase equilibrium calculations. This work develops a new hybrid multiphase flash calculation algorithm adopting both equilibrium ratios and phase fractions as iteration variables. Different from the standalone implementation of Newton-Raphson method documented in the literature, the hybrid algorithm properly hybridizes Newton-Raphson and trust-region methods to achieve a higher computational efficiency and a better robustness. We show the fast convergence behavior of the hybrid algorithm in some scenarios where the conventional Newton-Raphson method encounters convergence difficulties. We also demonstrate the good performance of the hybrid algorithm by constructing phase diagrams for several reservoir fluid mixtures.

2.1 Introduction

Phase equilibrium calculation framework, where the flash calculation code is one computational kernel, centers in plenty of petroleum engineering and chemical engineering applications. In a typical compositional simulation process, the fluid in each grid is assumed to be at local equilibrium instantaneously (Young, 2022). This means that the phase fractions and phase compositions in each grid at each time step should be determined prior to flow simulations. A significant number of equilibrium calculation calls will be made in such a simulation process. Thus, using a robust flash calculation algorithm in compositional simulations becomes extremely important. This will help ensure that the many compositional processes (such as CO_2 injection for enhanced oil recovery (EOR), multiphase flows in pipelines, and carbon sequestration in petroleum reservoirs) could be simulated in a reliable manner.

Michelsen (1982b) developed a robust and efficient flash calculation strategy with the use of both a lower-order solution method and a higher-order solution method. Such two methods are applied in sequence: the lower-order one is implemented first to generate good initial guesses, followed by the higher-order one to reach the final convergence. Conventionally, the lower-order method refers to the successive substitution (SS) method (Michelsen, 1982b) or its accelerated versions (Michelsen, 1982b; Heidemann, 1983; Mehra et al., 1983), while the higher-order methods differ greatly in terms of how the independent variables are solved. For example, a conventional higher-order solution method is to minimize the Gibbs free energy with mole numbers as the independent variables (Michelsen, 1982b; Perschke, 1988; Petitfrere and Nichita, 2014; Pan et al., 2019), while several other methods solve for the natural logarithm of the equilibrium ratios (lnK) (Michelsen, 1982b; Nghiem and Li, 1984; Ammar and Renon, 1987). In addition, two recent works further improve the lnK method by simultaneously solving lnK and phase fractions (θ) with Newton-Raphson (NR) method (Haugen et al., 2011; Petitfrere and Nichita, 2016).

The simultaneous solution of lnK and θ has its unique characteristics. Foremost, both the lower-order and higher-order algorithms adopt the equilibrium ratios (lnK) as the independent variables. Besides, lnK are better scaled than mole numbers that are used in the conventional higher-order methods. Furthermore, contrary to the original lnK method where the explicit solution of Rachford-Rice (RR) equations (Rachford and Rice, 1952) is required (Michelsen, 1982b; Ammar and Renon, 1987), such explicit solution of RR equations is not needed in the simultaneous solution approach (Petitfrere and Nichita, 2016). Nevertheless, the simultaneous solution approach suffers from two major drawbacks. First, the conventional NR method is applied in the simultaneous solution approach documented in the literature (Haugen et al., 2011; Petitfrere and Nichita, 2016). The NR method, however, as a classical Newton-type root-finding method, does not guarantee the decrease of the Gibbs free energy as iteration proceeds (Okuno, 2009). Second, in challenging cases where accurate initial guesses are not available, NR method itself is not robust. Specifically, NR method may fail in the flash calculations if the current temperature-pressure condition is located in near-critical regions, or low-quality trial phases are obtained from the stability tests (Petitfrere and Nichita, 2014; Pan et al., 2019). For example, in several difficult cases illustrated by Pan et al. (2019), the combined use of SS and Newton's method cannot lead to convergence with more than one thousand iterations.

To address the above issue, Petitfrere and Nichita (2014) introduced a robust trust-region (TR) method in flash calculations. The TR-based flash calculation algorithm is effective in reducing the number of iterations to fewer than 30 in the hard cases, as illustrated by Pan et al. (2019). However, such a TR method is only intended to minimize Gibbs free energy in terms of mole numbers. To our knowledge, we are lacking a robust solution method that is dedicated to the simultaneous approach. Therefore, the objective of this work is to develop a more robust and efficient flash calculation algorithm that minimizes the Gibbs free energy in terms of lnK and θ. We design a hybrid flash calculation algorithm which properly hybridizes NR and TR methods. The algorithm follows a general procedure as explained below. A NR step will be tested first. If the NR step is not suitable, a TR step will be then initiated and evaluated. If the current TR step is still not satisfactory, a final switch to an SS step will be executed.

This paper is structured as follows. The definition of the flash calculation problem, the solution method, and the numerical procedure are included in "Methodology". Case studies to show the performance of the new algorithm are illustrated in "Results and discussion". At last, we draw the major conclusions in the "Conclusions" section.

2.2 Methodology

2.2.1 NR method

Assuming the feed (z) containing nc components is split into np phases at given pressure and temperature, and the np^{th} phase is specified as the reference phase, we can have the following material balance equation,

$$\sum_{l=1}^{np-1} \theta_l x_{il} + \left(1 - \sum_{l=1}^{np-1} \theta_l\right) x_{i,np} = z_i \tag{2-1}$$

where x_{il} is the mole fraction of component i in phase l, and θ_l is the phase fraction of phase l. Note that the mole numbers satisfy $n_{il} = \theta_l x_{il}$ and $n_{i,np} = \left(1 - \sum_{l=1}^{np-1} \theta_l\right) x_{i,np}$, where n_{il} refers to the mole fraction of component i in phase l. Defining the equilibrium ratio as $K_{il} = x_{il}/x_{i,np}$, we can obtain the following equations:

$$x_{i,np} = \frac{z_i}{1+\sum_{l=1}^{np-1} \theta_l (K_{il}-1)} \tag{2-2}$$

$$x_{il} = \frac{K_{il} z_i}{1+\sum_{l=1}^{np-1} \theta_l (K_{il}-1)} \tag{2-3}$$

Considering that equilibrium ratios and phase fractions are the independent variables, we have (Petitfrere and Nichita, 2016),

$$\boldsymbol{Var} = \left[\boldsymbol{lnK}_1^T, \cdots, \boldsymbol{lnK}_{np-1}^T, \boldsymbol{\theta}^T\right]^T \tag{2-4}$$

where

$$\boldsymbol{lnK}_k = \left[lnK_{1k}, \cdots, lnK_{nc,k}\right]^T, k = 1, \cdots, np - 1 \tag{2-5}$$

$$\boldsymbol{\theta} = \left[\theta_1, \cdots, \theta_{np-1}\right]^T \tag{2-6}$$

Based on Haugen et al. (2011) and Petitfrere and Nichita (2016), $\boldsymbol{Var}$ defined by **Eq. 2-4** can be determined by solving a set of nonlinear equations given as,

$$g_{ik} = \frac{\partial G}{\partial n_{ik}} = lnK_{ik} + ln\varphi_{ik} - ln\varphi_{i,np} \tag{2-7}$$

$$R_k = \sum_{i=1}^{nc} \frac{z_i(K_{ik}-1)}{1+\sum_{l=1}^{np-1}\theta_l(K_{il}-1)} \tag{2-8}$$

where G is the dimensionless Gibbs free energy and g is its gradient in terms of the mole number, R refers to the RR equation, and φ_{ik} refers to the fugacity coefficient of component i in phase k. Then an NR step ($\boldsymbol{s}_{NR}$) is implemented as per,

$$J\boldsymbol{s}_{NR} = -\boldsymbol{f} \tag{2-9}$$

where

$$\boldsymbol{f} = \left[\boldsymbol{g}_1^T, \cdots, \boldsymbol{g}_{np-1}^T, R_1, \cdots, R_{np-1}\right]^T \tag{2-10}$$

where f represents the fugacity. The Jacobian matrix (J) in **Eq. 2-9** is a matrix with the dimension of $nc(np-1) \times nc(np-1)$. Appendix 2-A.1 lists the detailed Jacobian elements and the related partial derivatives.

2.2.2 Hybrid NR and TR method

The basic idea to implement the TR method is to solve the phase equilibrium problem using the variable vector $\boldsymbol{Var}$ as defined by Eq. 2-4. Based on **Eqs. 2-2** and **2-3**, we can have,

$$x_{i,np} = x_{i,np}\left(\boldsymbol{\theta}, lnK_{i1}, \cdots, lnK_{i,np-1}\right) \tag{2-11}$$

$$x_{il} = x_{il}\left(\boldsymbol{\theta}, lnK_{i1}, \cdots, lnK_{i,np-1}\right) \tag{2-12}$$

$$n_{i,np} = n_{i,np}(\boldsymbol{\theta}, lnK_{i1}, \cdots, lnK_{i,np-1}) \tag{2-13}$$

$$n_{il} = n_{il}(\boldsymbol{\theta}, lnK_{i1}, \cdots, lnK_{i,np-1}) \tag{2-14}$$

With known mole fraction x and mole number n, we can define the dimensionless Gibbs free energy as,

$$G = \sum_{k=1}^{np} \sum_{s=1}^{nc} n_{sk} lnf_{sk}\,(\boldsymbol{x}_k) \tag{2-15}$$

In **Eq. 2-15**, G is treated as a function in terms of the independent variables $\boldsymbol{Var}$. We can then write the elements of the gradient of G in terms of $\boldsymbol{Var}$ as,

$$\frac{\partial G}{\partial lnK_{jp}} = \sum_{k=1}^{np-1} \sum_{s=1}^{nc} \left(\frac{\partial n_{sk}}{\partial lnK_{jp}} lnf_{sk} + \frac{\partial lnf_{sk}}{\partial lnK_{jp}} lnf_{sk}\right) + \sum_{s=1}^{nc} \left(\frac{\partial n_{s,np}}{\partial lnK_{jp}} lnf_{s,np} + \frac{\partial lnf_{s,np}}{\partial lnK_{jp}} lnf_{s,np}\right) \tag{2-16}$$

$$\frac{\partial G}{\partial \theta_p} = \sum_{k=1}^{np-1} \sum_{s=1}^{nc} \left(\frac{\partial n_{sk}}{\partial \theta_p} lnf_{sk} + \frac{\partial lnf_{sk}}{\partial \theta_p} lnf_{sk}\right) + \sum_{s=1}^{nc} \left(\frac{\partial n_{s,np}}{\partial \theta_p} lnf_{s,np} + \frac{\partial lnf_{s,np}}{\partial \theta_p} lnf_{s,np}\right) \tag{2-17}$$

The partial derivatives involved in **Eqs. 2-16** and **2-17** are presented in Appendix 2-A.1. Considering the complexity of calculating the Hessian matrix of G in terms of the independent variables, we calculate the true gradient g, but approximate the Hessian B with the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method (Nocedal and Wright, 2006). The approximated BFGS Hessian matrix is updated by,

$$B^{n+1} = B^n + \frac{y^n (y^n)^T}{(y^n)^T p^n} - \frac{B^n p^n (p^n)^T B^n}{(p^n)^T B^n p^n} \tag{2-18}$$

where n is the iteration index, $B_1 = I$, and

$$y^n = g^n - g^{n-1}, n > 1 \tag{2-19}$$

$$p^n = Var^n - Var^{n-1}, n > 1 \tag{2-20}$$

Using the approximated Hessian matrix, the TR subproblem can be solved. The model function m in terms of the minimizer (i.e., the TR step) s is calculated by (Nocedal and Wright, 2006),

$$m^n(s^n) = (g^n)^T s^n(s_n) + 0.5(s^n)^T B^n s^n \tag{2-21}$$

The TR ratio ρ is determined by (Nocedal and Wright, 2006),

$$\rho^n = \frac{G(Var^n) - G(Var^n + s^n)}{m^n(0) - m^n(s^n)} \tag{2-22}$$

The TR algorithm developed in this study is based on the state-of-the-art TR algorithm adopting mole numbers as the independent variables (Petitfrere and Nichita, 2014; Pan et al., 2019). We closely follow the realistic exact subproblem solver (Conn et al., 2000) to calculate the TR step s. Detailed descriptions of the subproblem solver can be found in Conn et al. (2000) and Petitfrere and Nichita (2014).

As the Hessian matrix is approximated by the BFGS method, applying the TR method alone may compromise the convergence rate. Herein, we design a new hybrid trust-region algorithm that leverages the advantages of both NR and TR methods. **Fig. 2-1** illustrates the detailed numerical procedure of the hybrid algorithm. The general idea of the algorithm is to try an NR step first. If the NR step is infeasible, we try a TR step alternatively. If the TR step is still infeasible, we activate a final switch to an SS step. Such a switching strategy, as inspired by Petitfrere and Nichita (2014) and Pan et al. (2019), relies on an alternative method (i.e., TR or SS) to provide a safeguarded step if the current NR method does not work well. Before making a switch, we use relaxation to reduce the step size (Pan et al., 2021) and check the feasibility of the current step. In our implementation, one NR step will be accepted if both the Gibbs free energy and the residual error can be reduced, while a TR step will be taken if the TR ratio ρ is between 0 and 10. The residual error (*err*) is defined as:

$$err = \sqrt{\sum_{s=1}^{nc} \sum_{k=1}^{np-1} \left(f_{sk} - f_{s,np}\right)^2} \tag{2-23}$$

As NR and TR methods are unconstrained methods, updating the independent variables by a NR/TR step may lead to unphysical phase fraction values, especially in early iterations. To address this issue, two extra criteria should be satisfied before accepting a NR/TR step. First, we check if the updated phase fractions are strictly bounded between 0 and 1. Second, we substitute the updated K-values into RR equations to verify if the output phase fractions are bounded between 0 and 1. If any of the two criteria is violated, the NR/TR step is considered infeasible and the algorithm will turn to the next alternative method (i.e., TR or SS). We will further exemplify how the numerical procedure of the hybrid algorithm is implemented in the "Results and discussion" section.

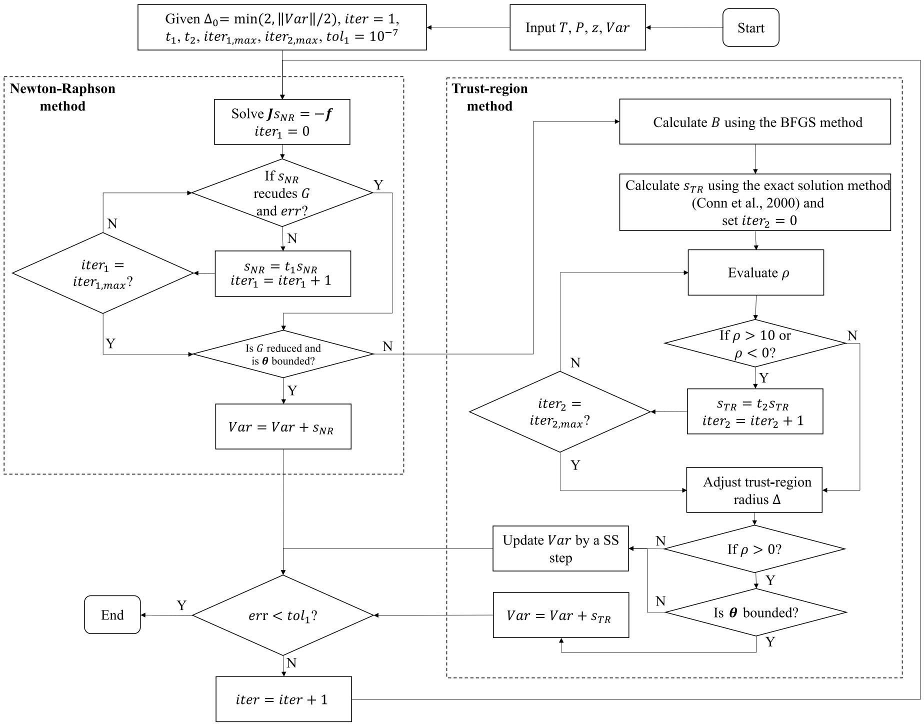


Fig. 2-1. Flowchart of the hybrid NR and TR flash calculation algorithm developed in this study.

We closely follow Michelsen's strategy (1982b) to combine the low-order SS method with a higher-order one (i.e., the developed hybrid algorithm) to build a robust flash calculation algorithm. Specifically, we follow the switching strategy implemented by Petitfrere and Nichita (2016) and Pan et al. (2019). **Fig. 2-2** presents the numerical procedure executed in the switching strategy. After inputting a trial phase from the stability test, we execute the conventional SS method first until the residual becomes smaller than 10^{-2}. The maximum-allowable number of iterations in the SS method is set to 20, as done by Pan et al. (2019). Afterwards, the hybrid NR and TR algorithm will be run until convergence is reached.

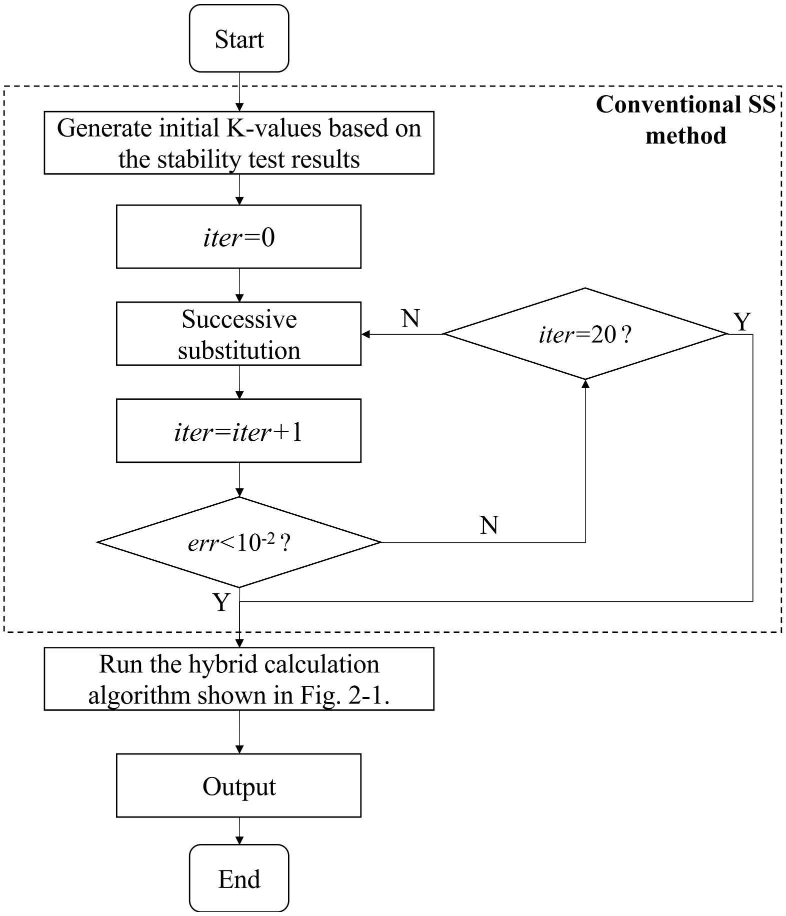


Fig. 2-2. Flowchart of the switching strategy used to switch from the SS method to the hybrid NR and TR method.

2.2.3 Workflow of multiphase equilibrium calculation

Based on Michelsen (1982a; 1982b), an equilibrium calculation workflow consists of a series of steps that implement stability tests and flash calculations. A conventional multiphase equilibrium calculation follows a stagewise workflow. The stability of the overall feed at given pressure and temperature is examined first. If not stable, a two-phase flash calculation is implemented, and another two-phase stability test ensues. If the two-phase mixture is still unstable, a three-phase flash calculation will be performed. This workflow continues until the maximum allowable phase number is reached. Pan et al. (2019) proposed a state-of-the-art workflow for performing general equilibrium calculations, especially for CO_2-inclusive mixtures. The important improvement embodied in the workflow by Pan et al. (2019) is that the stagewise workflow will be conducted one more time if the first three-phase flash calculation fails, and if there is at least one more trial phase available from the one-phase stability test. Such a feature allows the workflow to robustly detect the correct multiphase equilibrium. A total of 9 pressure-composition diagrams are successfully plotted by Pan et al. (2019) without encountering any convergence issues. In this paper, we apply Pan et al. workflow considering its superb performance in performing multiphase equilibrium calculations (Pan et al., 2019; Lu et al., 2021; Li, 2021). The developed hybrid flash calculation code is a subroutine embedded into the workflow developed by Pan et al. (2019).

In the stability test, we use a total of 7 initial guesses of equilibrium ratios. The initial guesses are given by (Michelsen, 1982a; Li and Firoozabadi, 2012; Pan et al., 2019),

$$K_i^{Wilson}, 1/K_i^{Wilson}, \sqrt[3]{K_i^{Wilson}}, 1/\sqrt[3]{K_i^{Wilson}}, K_i^{CO_2}, K_i^{H_2O}, K_i^{nc} \tag{2-24}$$

where

$$K_i^{Wilson} = P_{c,i}\frac{\exp\left[5.37(1+\omega_i)\left(1-\frac{T_{c,i}}{T}\right)\right]}{P} \tag{2-25}$$

and T_c, P_c, and ω represent critical temperature, critical pressure, and acentric factor, respectively, and

$$K_{CO_2}^{CO_2} = \frac{0.9}{z_{CO_2}},\ K_i^{CO_2} = \frac{0.1}{z_i(nc-1)},\ \ i \neq CO_2, i\ \in\ (1,2,\cdots,nc) \tag{2-26}$$

$$K_{H_2O}^{H_2O} = \frac{0.9}{z_{H_2O}},\ K_i^{H_2O} = \frac{0.1}{z_i(nc-1)}, i \neq H_2O, i\ \in\ (1,2,\cdots,nc) \tag{2-27}$$

$$K_{nc}^{nc} = \frac{0.9}{z_{nc}},\ K_i^{nc} = \frac{0.1}{z_i(nc-1)},\ i\ \in\ (1,2,\cdots,nc-1) \tag{2-28}$$

Note that K^{H_2O} is only required when the feed contains water.

2.3 Results and discussion

In this section, we test the performance of the hybrid NR and TR algorithm using a total of 6 benchmark fluid mixtures, including CO_2-inclusive, acid-gas-inclusive, and water-inclusive mixtures. Specifically, these mixtures are the JEMA oil (Khan et al., 1992), the Maljamar Reservoir Oil (MRO) (Orr et al., 1981; Haugen et al., 2011), the North Ward Este (NWE) oil (Khan et al., 1992), the Oil G (Khan et al., 1992), an acid gas mixture (Pan and Firoozabadi, 1998; Haugen et al., 2011), and the MY10 oil with the presence of CO_2 and water (Metcalfe and Yarborough, 1979; Johns and Orr, 1996; Imai et al., 2019). Appendix 2-A.2 lists the detailed properties of these fluid mixtures. Peng–Robinson equation of state (Robinson and Peng, 1978) has been used in all the calculations.

2.3.1 Comparison of the convergence behavior

Different strategies have been applied when incorporating the NR method in the flash calculation algorithms. In Haugen et al. (2011) and Li and Firoozabadi (2012), the general strategy is to run the SS method first and switch to the NR method if $err < 10^{-5}$. A permanent switch back to the SS method is activated when the NR method fails. We refer to this algorithm as SS+NR-H in the following discussion. In Petitfrere and Nichita (2016), the SS method switches to the NR method when $err < 10^{-2}$. In this NR solver, one SS step will replace the NR step if the NR step does not decrease the Gibbs free energy. This algorithm is referred to as SS+NR-PN. Our flash calculation strategy, which switches from the SS method to the hybrid NR and TR algorithm (HNRTR) when $err < 10^{-2}$, will be referred to as SS+HNRTR.

To make a fair comparative analysis, we build three parallel equilibrium calculation codes, each of which employs one flash calculation strategy. Following Petitfrere and Nichita (2014), we run the three codes to construct three digital pressure-composition (Px) phase diagrams for a given mixture, each involving a total of $400 \times 400 = 160000$ runs of equilibrium calculations.

Fig. 2-3 first shows the Px diagram generated by SS+HNRTR for the mixture (the JEMA oil mixed with CO_2). Note that SS+NR-H and SS+NR-PN yield the same diagrams, as shown in Fig. 2-3. **Fig. 2-4** compares the numbers of iterations consumed by the three flash calculation strategies over the Px diagram. Each point in the colormap in Fig. 2-4 represents the total number of iterations consumed by the flash calculations at this isobaric-isothermal point of the Px diagram. The total number of iterations here refers to the summed-up number of iterations consumed in both two-phase and three-phase flash calculations as defined by Petitfrere and Nichita (2014). As can be seen from Fig. 2-4, the convergence of the flash calculations becomes difficult in regions near the critical point and the three-phase point (i.e., the intersection point of the two-phase boundary and the three-phase boundary). In these regions, both SS+NR-H and SS+NR-PN require relatively large numbers of iterations to reach convergence, while SS+HNRTR requires much fewer number of iterations. **Table 2-1** further summarizes the average and maximum number of iterations consumed by the three flash strategies to construct the Px diagram. As listed in Table 2-1, the maximum numbers of iterations yielded by SS+NR-H and SS+NR-PN in the three-phase region are 1784 and 1003, respectively, while the number yielded by SS+HNRTR is only 74. Besides, SS+HNRTR shows the smallest average numbers of 19.55 and 38.77 in the two-phase and three-phase regions, respectively. These facts demonstrate a good ability of SS+HNRTR in further improving the computational efficiency of flash calculations.

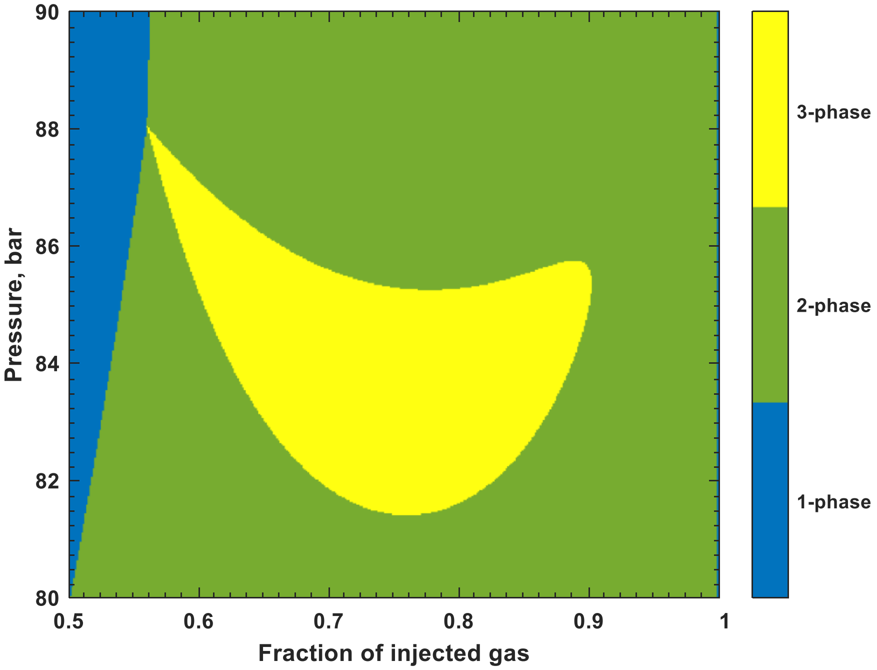
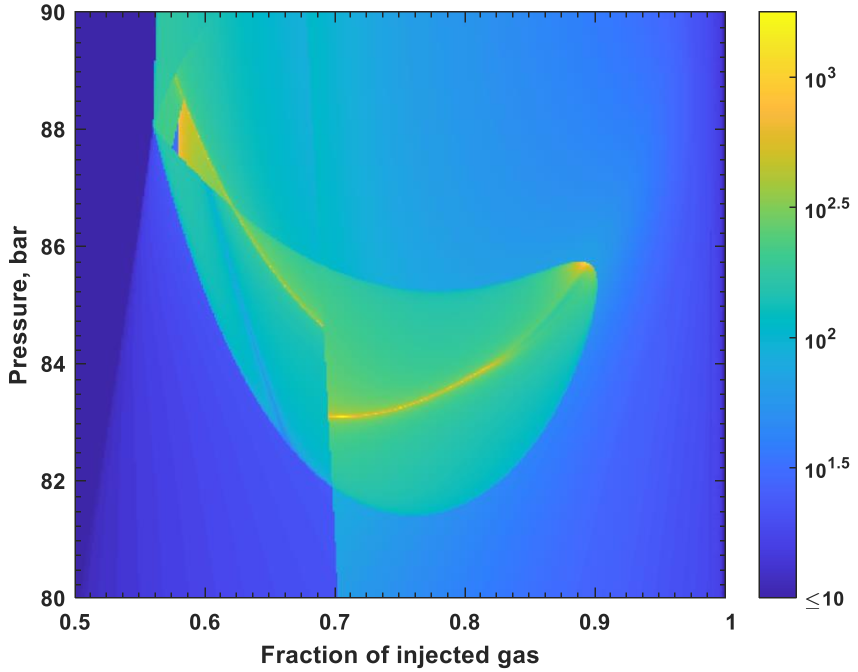
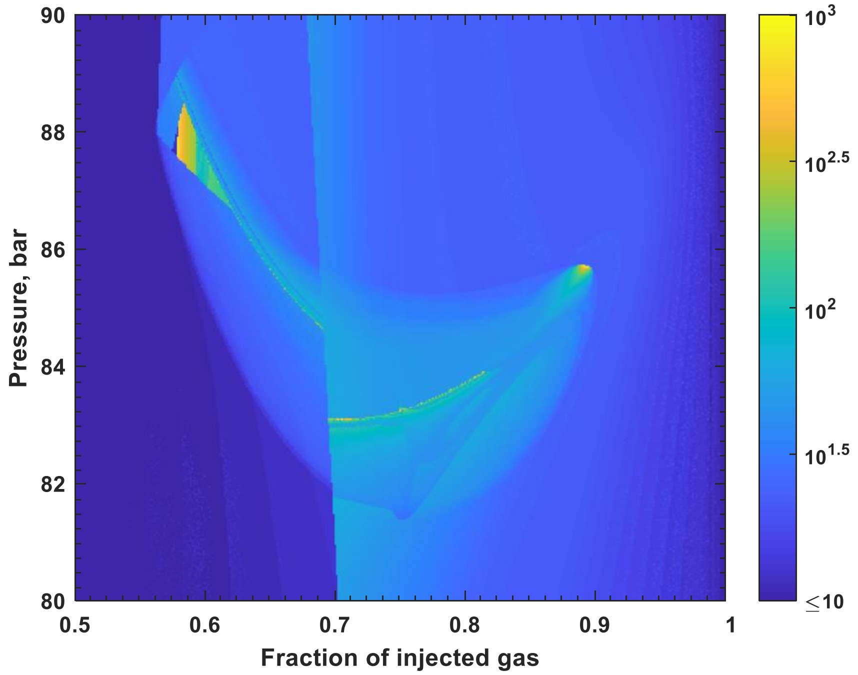


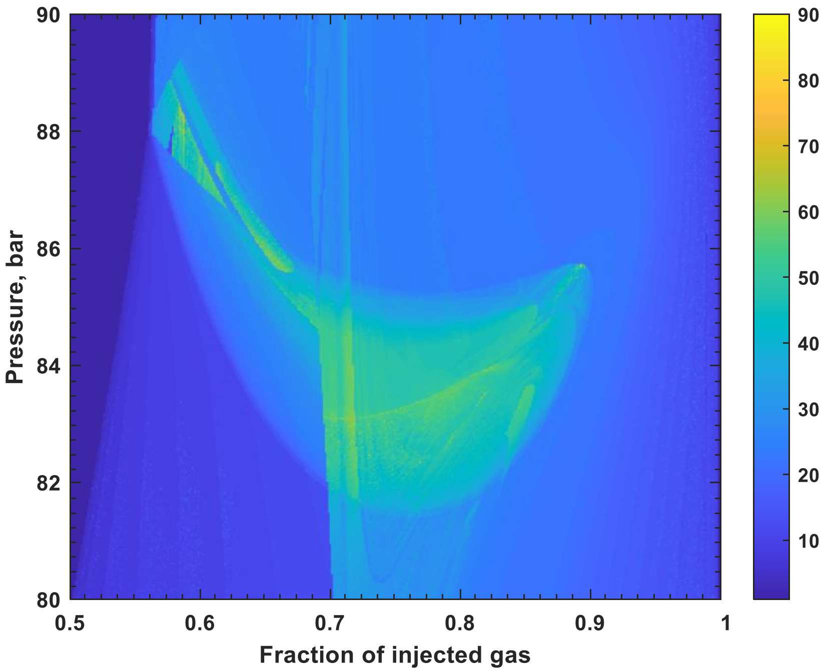
Fig. 2-3. Px phase diagrams generated by SS+HNRTR for the mixture (the JEMA oil mixed with CO_2). This diagram is constructed by running a total of $400 \times 400 = 160000$ equilibrium calculations.


90
88
86
84
82
80
Pressure, bar
0.5
0.6
0.7
0.8
0.9
1
Fraction of injected gas
10^3
$10^{2.5}$
10^2
$10^{1.5}$
≤10

(a)


90
88
86
84
82
80
Pressure, bar
0.5
0.6
0.7
0.8
0.9
1
Fraction of injected gas
10^3
$10^{2.5}$
10^2
$10^{1.5}$
≤10

(b)



(c)

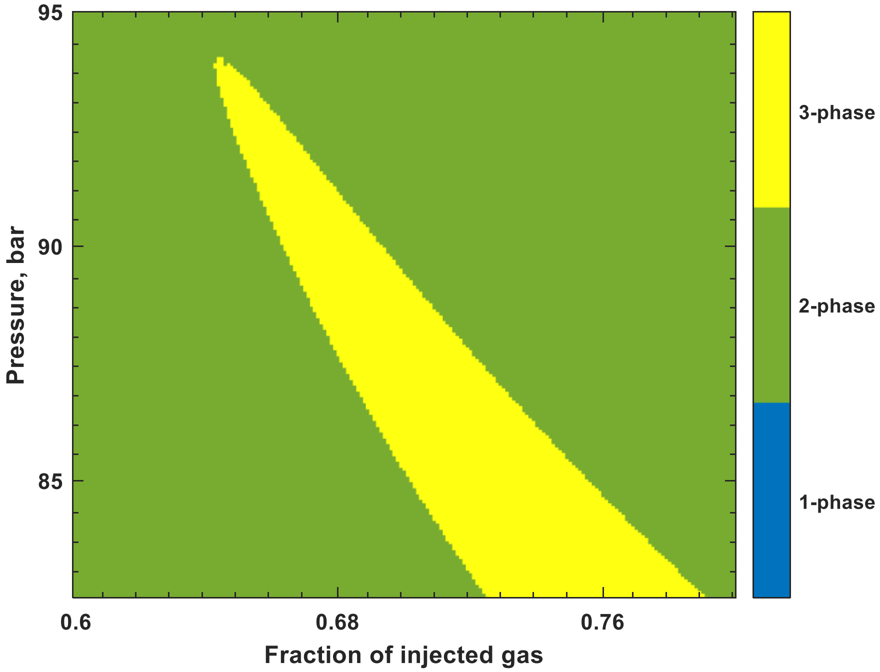
Fig. 2-4. Total number of iterations consumed by the flash calculations at each point of the Px phase diagram for the mixture (the JEMA oil mixed with CO_2): (a) SS+NR-H; (b) SS+NR-PN; (c) SS+HNRTR.

Table 2-1. Average and maximum numbers of iterations consumed by the flash calculation strategies to construct the Px diagram of the mixture (the JEMA oil mixed with CO_2).

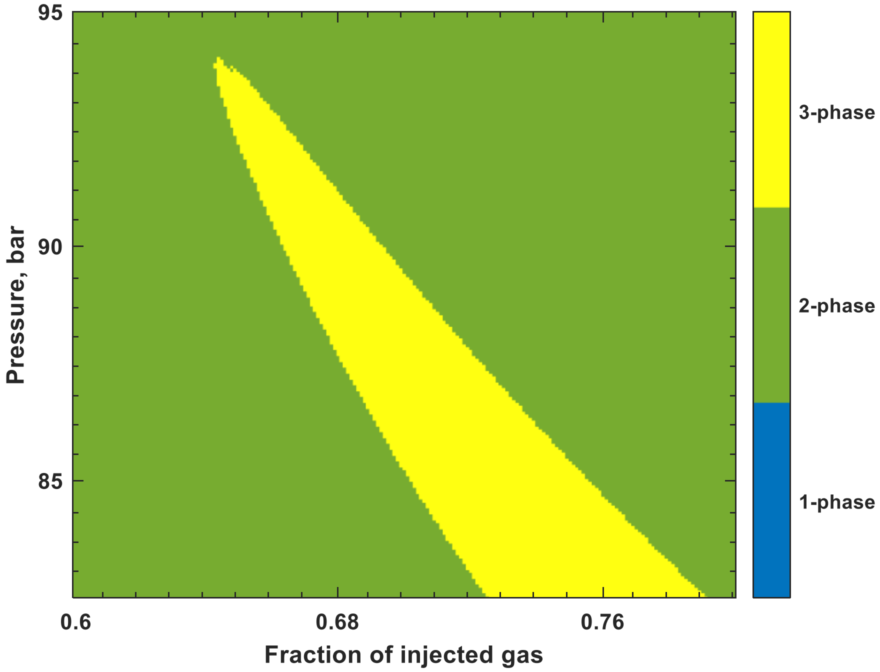
	SS+NR-H	SS+NR-PN	SS+ HNRTR
Average number of iterations consumed at points in the two-phase regions	51.94	21.72	19.55
Maximum number of iterations consumed at points in the two-phase regions	1327	854	76
Average number of iterations consumed at points in the three-phase regions	176.96	43.66	38.77
Maximum number of iterations consumed at points in the three-phase regions	1784	1003	74

We then test the three strategies by constructing the Px diagram for the mixture (the MRO oil mixed with CO_2). **Fig. 2-5** shows the Px diagrams yielded by the three flash strategies. Note that Fig. 2-5a and b are properly scaled to show the convergence issues encountered when running SS+NR-H and SS+NR-PN, respectively. **Fig. 2-6** compares the numbers-of-iteration colormaps generated by the three strategies. **Table 2-2** further summarizes the average and maximum number of iterations consumed by the flash calculation strategies to construct the phase diagram of the mixture (the MRO oil mixed with CO_2). In this case, SS+NR-PN becomes as efficient as SS+HNRTR in the two-phase region. However, the new SS+HNRTR remains the most cost-effective one among the three algorithms for flash calculations in the three-phase region. Furthermore, SS+HNRTR behaves robustly in the challenging three-phase cases near the bicritical point, where both SS+NR-H and SS+NR-PN encounter convergence issues to reach convergence.

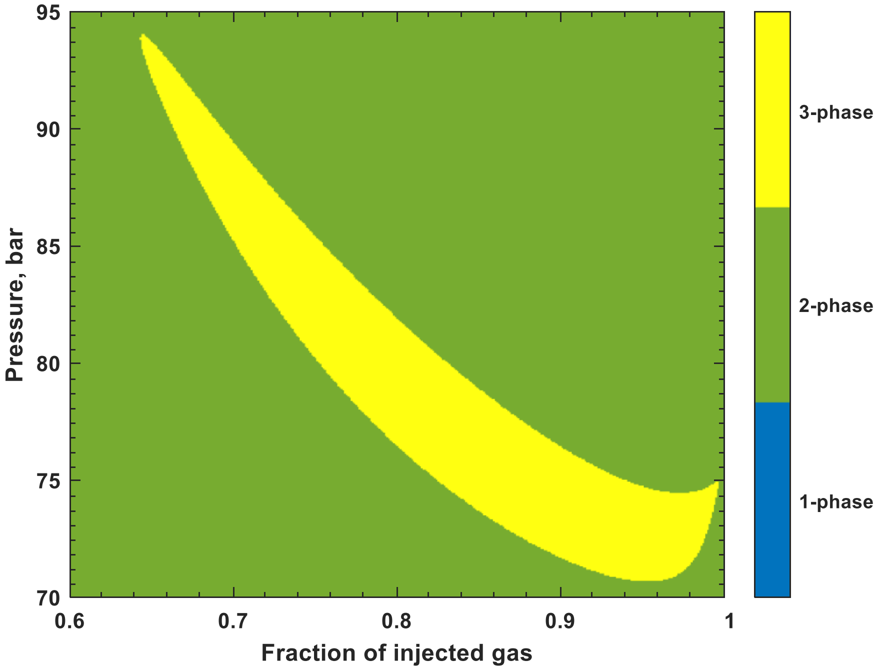
Such convergence difficulties/issues could be frequently encountered at conditions in the near-critical region, or where the flash calculations are poorly initialized for a CO_2-inclusive mixture (Petitfrere and Nichita, 2014; Pan et al., 2019). Here we select two test points from the aforementioned Px diagrams to show how the proposed hybrid algorithm can address this issue. **Fig. 2-7** compares the convergence behavior of the three different flash calculation algorithms at the two test points at which two-phase flash and three-phase flash exhibit convergence difficulties, respectively.


95
90
85
Pressure, bar
0.6
0.68
0.76
Fraction of injected gas
3-phase
2-phase
1-phase

(a)

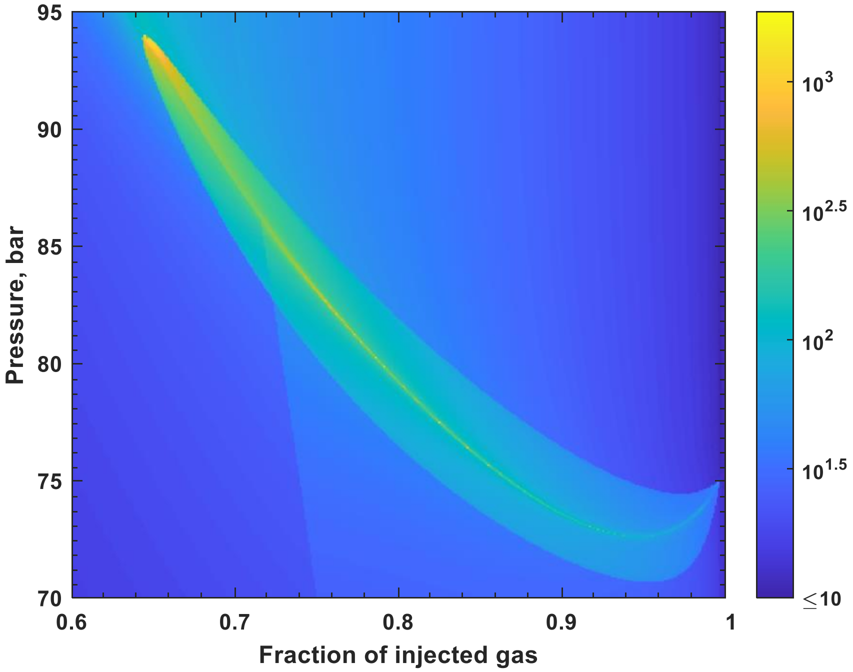

95
90
85
Pressure, bar
0.6
0.68
0.76
Fraction of injected gas
3-phase
2-phase
1-phase

(b)

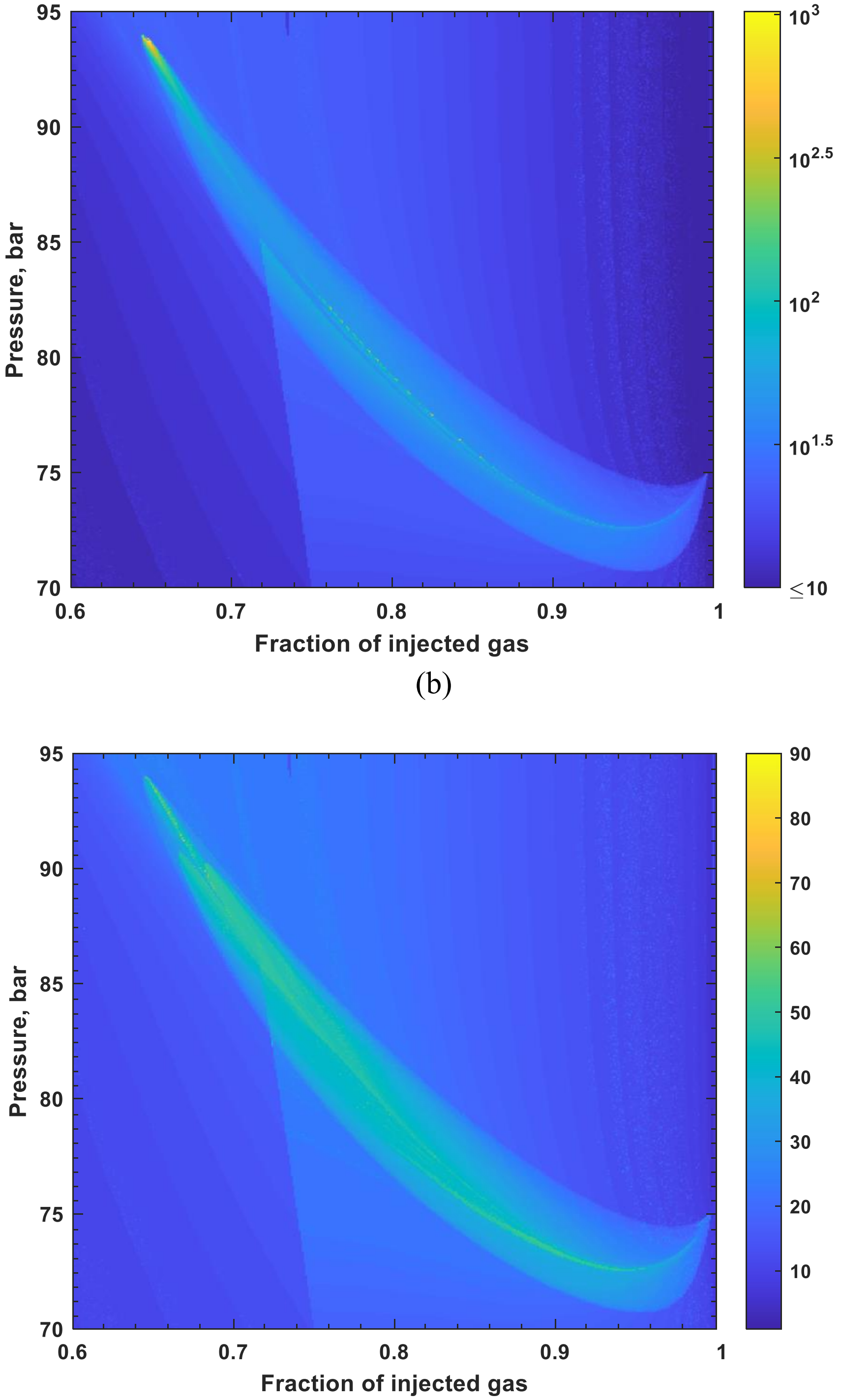


(c)

Fig. 2-5. Px phase diagrams generated by three flash calculation strategies for the mixture (the MRO oil mixed with CO_2): (a) SS+NR-H; (b) SS+NR-PN; (c) SS+HNRTR. Each diagram is constructed by running a total of $400 \times 400 = 160000$ equilibrium calculations. The first two diagrams are properly zoomed in to show the convergence difficulties.



(a)



(b)

(c)

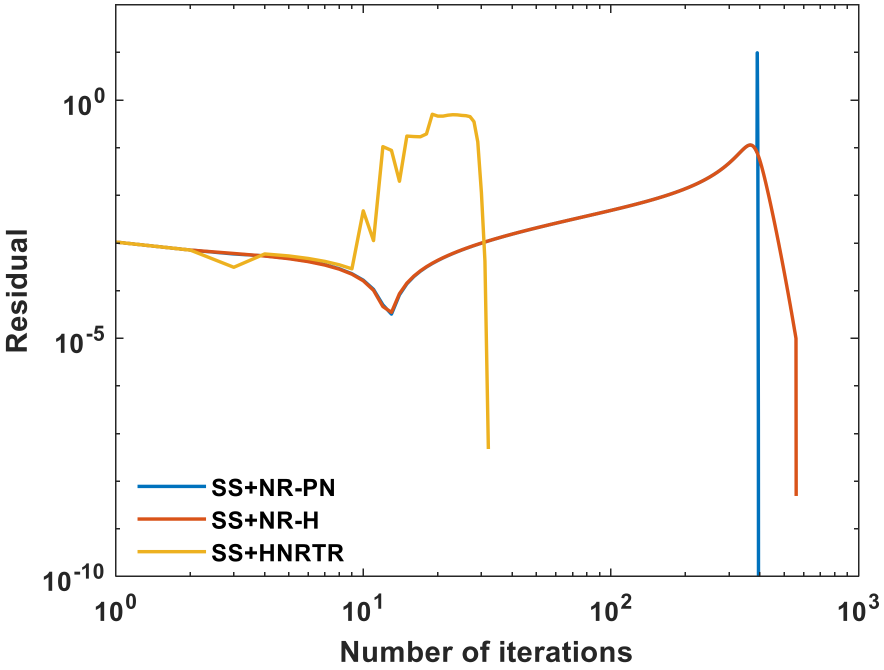
Fig. 2-6. Total number of iterations consumed by the flash calculations at each point of the Px phase diagram for the mixture (the MRO oil mixed with CO_2): (a) SS+NR-H; (b) SS+NR-PN; (c) SS+HNRTR.

Table 2-2. Average and maximum numbers of iterations consumed in the three flash calculation strategies to construct the Px diagram of the mixture (the MRO oil mixed with CO_2).

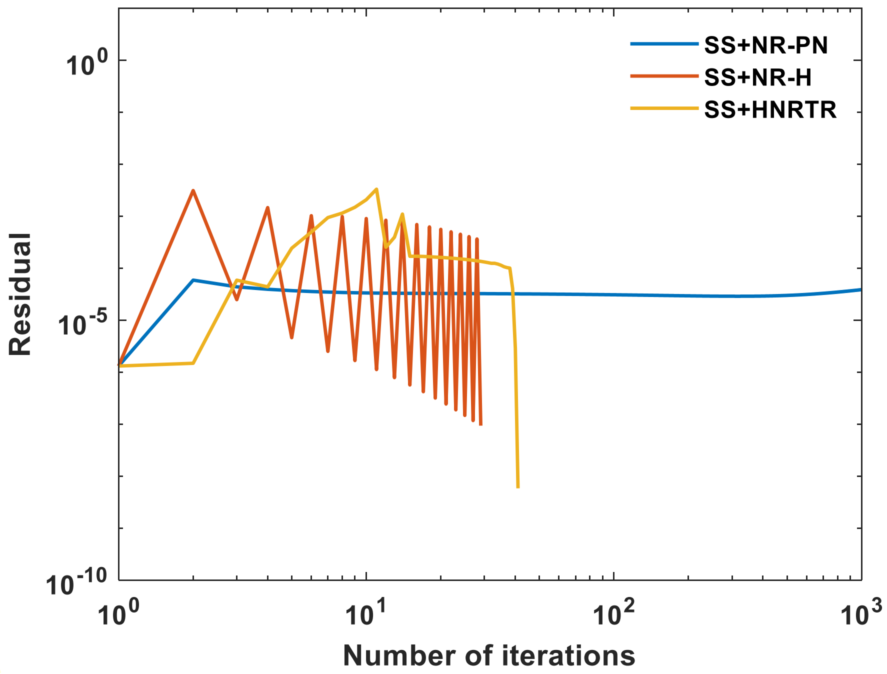
	SS+NR-H	SS+NR-PN	SS+ HNRTR
Average number of iterations consumed at points in the two-phase regions	30.17	16.05	16.07
Maximum number of iterations consumed at points in the two-phase regions	125	31	39
Average number of iterations consumed at points in the three-phase regions	126.4	39.22	35.75
Maximum number of iterations consumed at points in the three-phase regions	1867	1024	79

Fig. 2-7a corresponds to the test point at 87.625 bar and 316.48 K for the mixture (58.25 mol% CO_2 and 41.75 mol% JEMA oil). At this point, the two-phase flash calculation algorithm starts with a low-quality initial guess, causing a difficulty in converging the solution. As seen from Fig. 2-7a, SS+HNRTR converges with only 30 iterations. Compared with the fast convergence behavior of the hybrid algorithm, both SS+NR-H and SS+NR-PN require several hundreds of SS steps before a NR step can be successfully executed to reduce the residual significantly. SS+NR-H and SS+NR-PN consume 560 and 395 iterations before convergence, respectively.

Fig. 2-7b corresponds to the second test point at 93.9375 bar and 305.35 K for the mixture (64.60 mol% CO_2 and 35.4 mol% MRO). Such a test point is close to a bicritical point and requires a total number of 10624 SS steps alone to reach tight convergence in the three-phase flash calculation. In this case, both SS+HNRTR and SS+NR-H converge quickly with tens of iterations, while SS+NR-PN cannot converge with more than one thousand iterations. When we are running SS+NR-PN, we find that SS calls are consecutively made since the NR steps cannot reduce Gibbs free energy. After 1000 SS calls, we terminate the flash calculation. The residuals yielded by SS+NR-H exhibit drastic fluctuations during the iterations, resulting in a wrong solution with non-physical phase fractions. Based on Li and Firoozabadi (2012), a consecutive fluctuation of the residuals is a flag for switching to the SS method. Once we choose to switch back to the SS method, however, a significant computational cost may be incurred to reach convergence. In summary, in these two case studies, SS+HNRTR appears to be a more robust and efficient strategy than the other two counterparts.



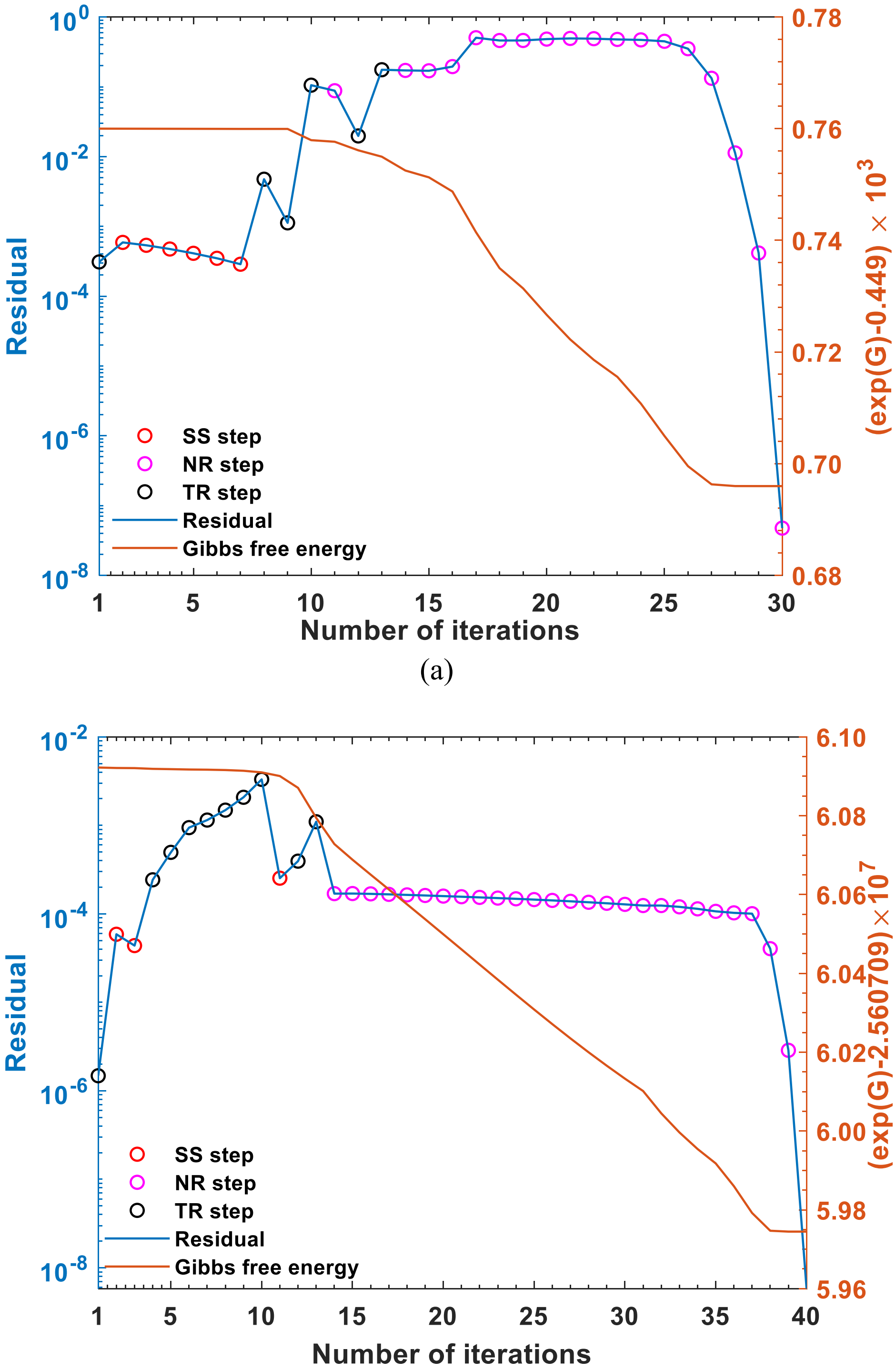
(a)



(b)

Fig. 2-7. Comparison of the convergence behavior of three flash calculation algorithms at two test points where convergence difficulty appears: (a) two-phase flash at 87.625 bar and 316.48 K for the mixture (58.25 mol% CO_2 and 41.75 mol% JEMA oil); (b) three-phase flash at 93.9375 bar and 305.35 K for the mixture (64.60 mol% CO_2 and 35.4 mol% MRO).

Fig. 2-8 shows the detailed variations in the residuals and Gibbs free energy yielded by the HNRTR algorithm at the aforementioned two test points. Note that we properly transform the Gibbs free energy values in order to present them clearly in Fig. 2-8. As illustrated in Fig. 2-8a and b, the TR and SS steps, especially the TR steps, are executed to find a descending direction of the Gibbs free energy in the early iterations. Once such a direction is successfully found, NR steps are applied to reach a fast convergence. We can see that the hybrid algorithm provides a robust approach to ensuring a decline in the Gibbs free energy as iteration proceeds, which is, however, not completely addressed by other algorithms (Haugen et al., 2011; Petitfrere and Nichita, 2016).



(a)

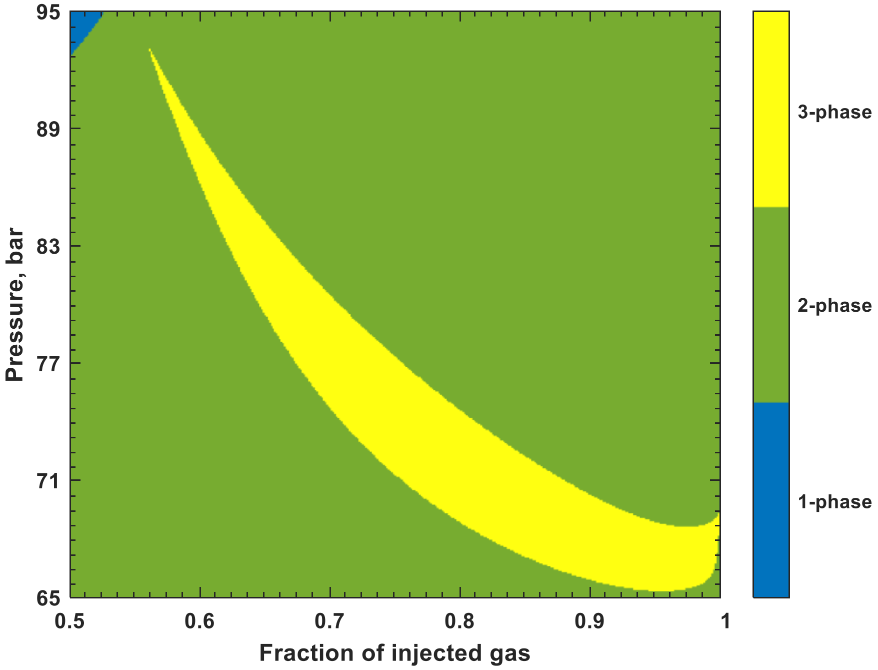
(b)

Fig. 2-8. Detailed variations in the residuals and Gibbs free energy yielded by the HNRTR algorithm at the two test points where convergence difficulties may appear: (a) two-phase flash at 87.6250 bar and 316.48 K for the mixture (58.25 mol% CO_2 and 41.75 mol% JEMA oil); (b) three-phase flash at 93.9375 bar and 305.35 K for the mixture (64.60 mol% CO_2 and 35.4 mol% MRO).

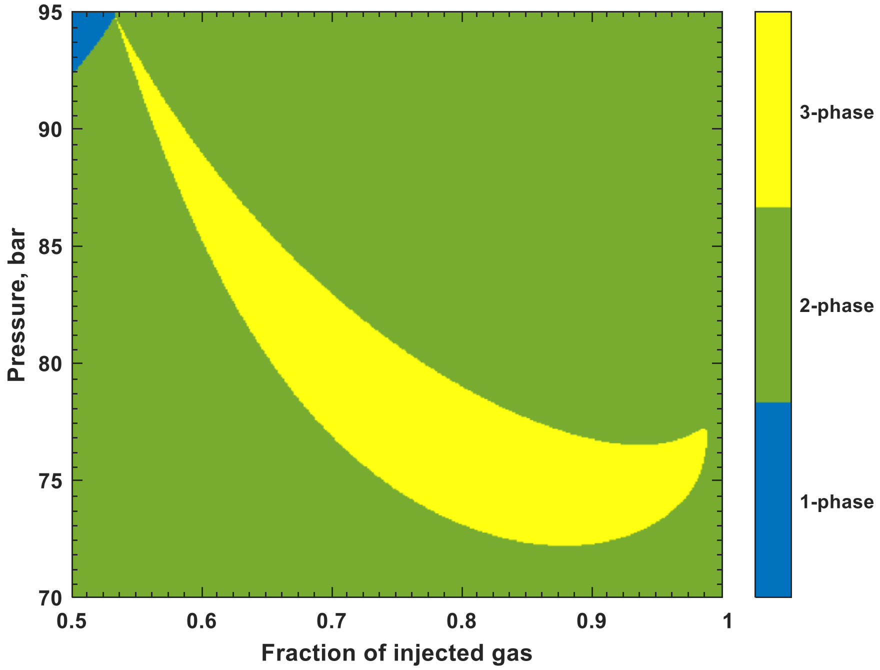
2.3.2 Phase diagrams generated using SS+HNRTR

We generate another 4 phase diagrams using the new flash calculation strategy, i.e., SS+HNRTR. These phase diagrams include 3 pressure-composition (Px) diagrams for the NWE oil, the Oil G, and the acid gas mixture, respectively, and 1 pressure-temperature diagram (PT) for the MY10-water mixture. **Fig. 2-9a-d** show the constructed Px and PT phase diagrams. Each diagram requires a total of $400 \times 400 = 160000$ runs of equilibrium calculations. No single convergence failure is encountered.

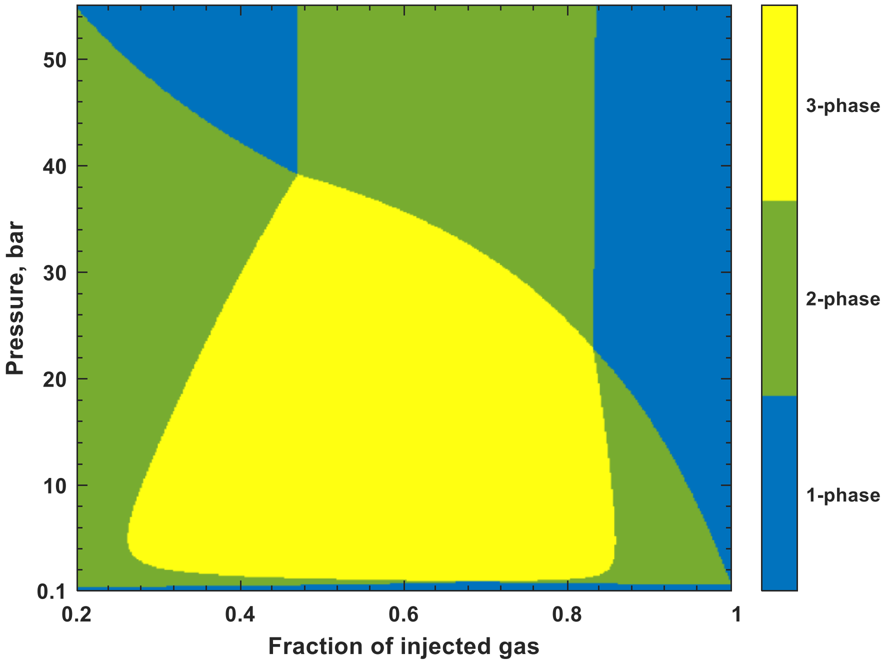
Fig. 2-10a-d illustrate the total number of iterations consumed by the flash calculations at each point in the phase diagrams. As shown in Fig. 2-10, the hybrid flash algorithm SS+HNRTR appears to be quite efficient since the maximum numbers of iterations consumed by flash calculations do not exceed 90.



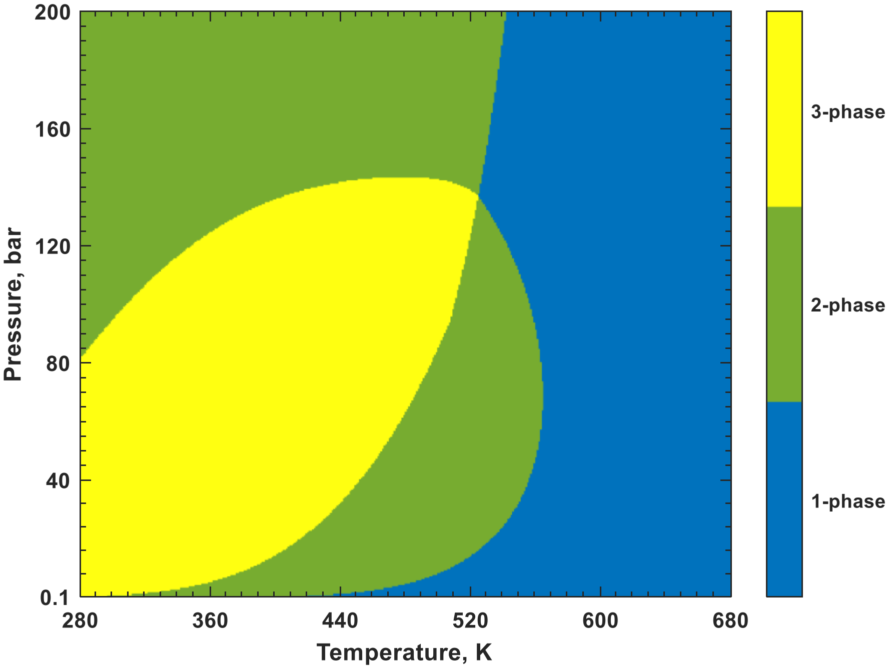
(a)


95
90
85
80
75
70
Pressure, bar
0.5
0.6
0.7
0.8
0.9
1
Fraction of injected gas
3-phase
2-phase
1-phase

(b)

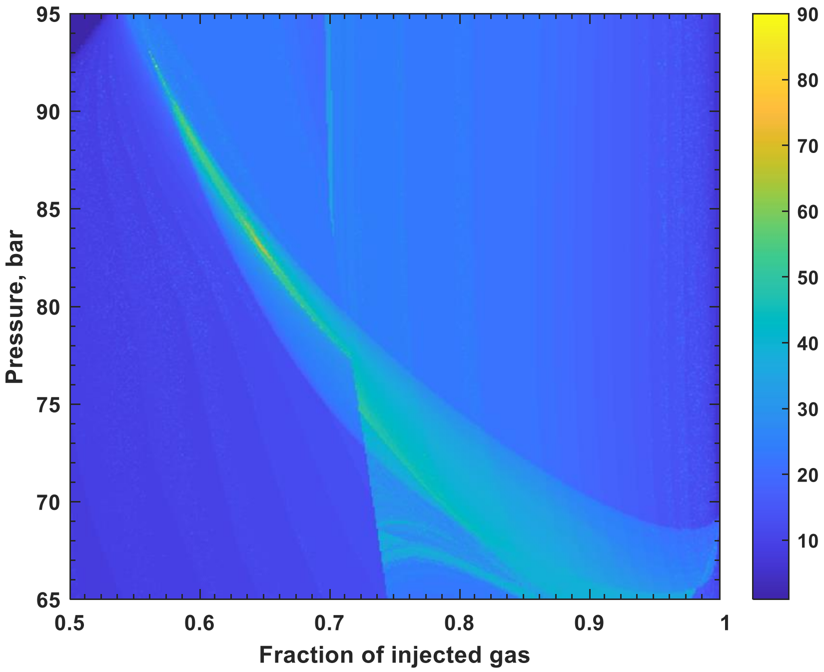

50
40
30
20
10
0.1
Pressure, bar
0.2
0.4
0.6
0.8
1
Fraction of injected gas
3-phase
2-phase
1-phase

(c)

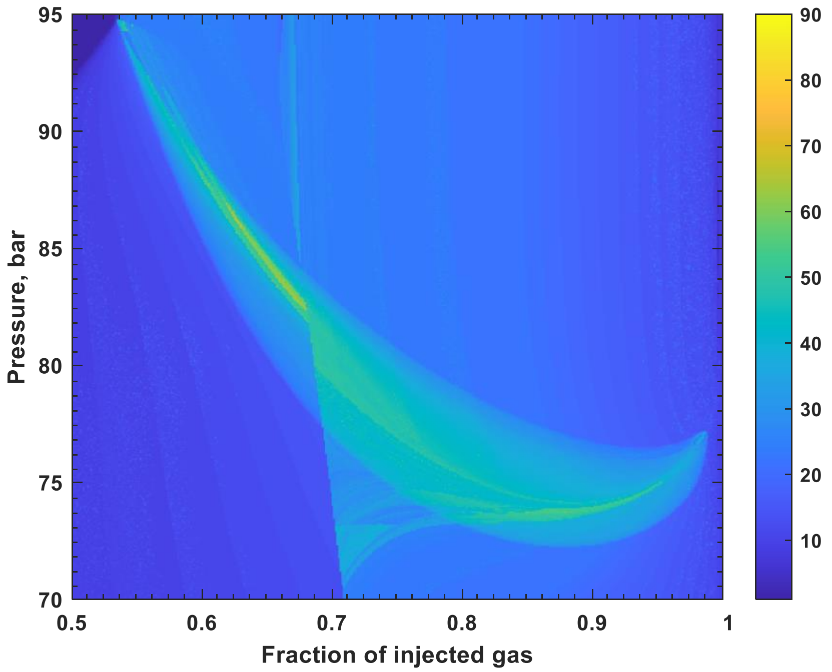


(d)

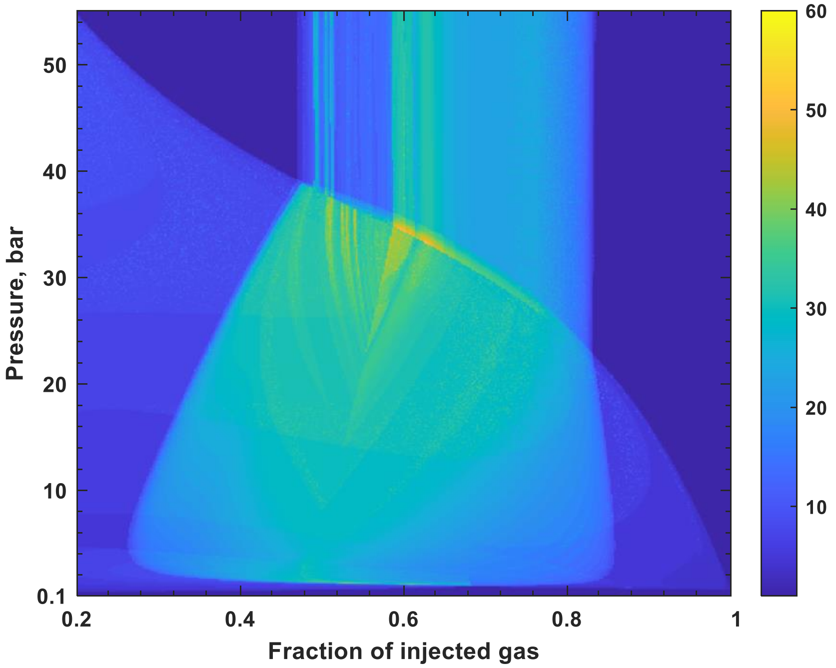
Fig. 2-9. Phase diagrams generated by SS+HNRTR: (a) Px diagram for the NWE oil; (b) Px diagram for the Oil G; (c) Px diagram for the acid gas mixture; (d) PT diagram of the MY10 oil and water mixture. Each diagram is constructed by running a total of $400 \times 400 = 160000$ equilibrium calculations.



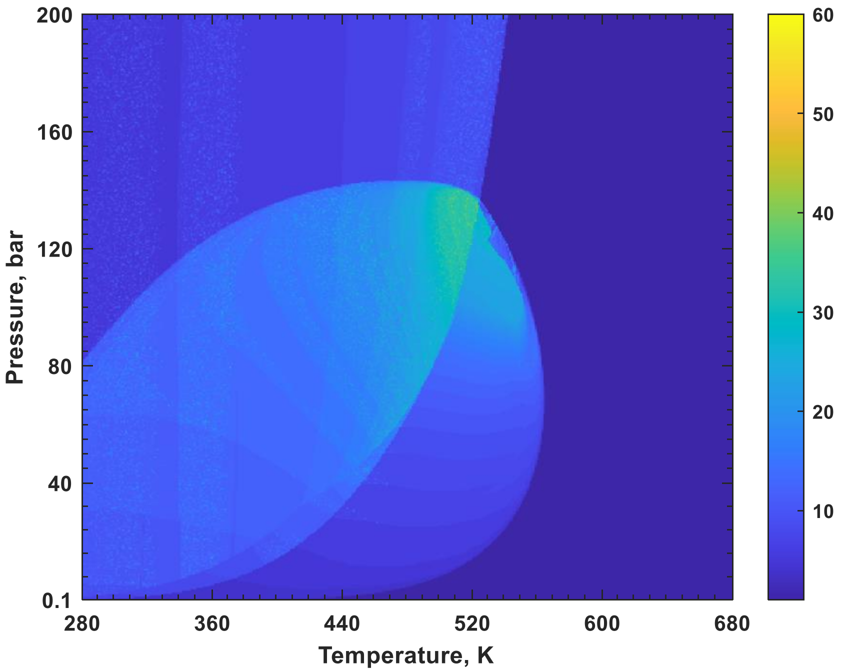
(a)


95
90
85
80
75
70
Pressure, bar
0.5
0.6
0.7
0.8
0.9
1
Fraction of injected gas
90
80
70
60
50
40
30
20
10

(b)


50
40
30
20
10
0.1
Pressure, bar
0.2
0.4
0.6
0.8
1
Fraction of injected gas
60
50
40
30
20
10

(c)



(d)

Fig. 2-10. Total number of iterations consumed by SS+HNRTR at each point of each phase diagram: (a) the NWE oil; (b) the Oil G; (c) the acid gas mixture; (d) the MY10 oil and water mixture.

Table 2-3 lists the average and maximum numbers of iterations consumed by the flash calculations to construct the 6 phase diagrams. As listed in Table 2-3, the maximum average numbers of iterations in the two-phase regions and in three-phase regions are no more than 20 and 40, respectively. These small average numbers indicate that SS+HNRTR is quite efficient in constructing the diagrams in the examined case studies.

Table 2-3. Average and maximum numbers of iterations consumed by the flash calculations to construct the 6 phase diagrams.

	JEMA	MRO	NWE	Oil G	Acid gas	MY10 oil and water mixture
Average number of iterations consumed at points in the two-phase regions	19.55	16.07	17.44	17.91	13.35	7.78
Maximum number of iterations consumed at points in the two-phase regions	76	39	47	56	40	37
Average number of iterations consumed at points in the three-phase regions	38.77	35.75	33.42	37.63	26.97	15.34
Maximum number of iterations consumed at points in the three-phase regions	74	79	87	71	52	40

2.4 Conclusions

In this work, we propose a new hybrid multiphase flash calculation algorithm adopting the well-scaled variables lnK and θ as the independent variables. The new algorithm properly hybridizes the NR and TR methods to improve the robustness and efficiency of flash calculations. The hybrid flash calculation algorithm is then successfully applied to construct a total of 6 phase diagrams for several reservoir mixtures. The new algorithm is shown to be more robust and efficient than the conventional algorithms documented in the literature. The current algorithm can be further extended to model four-phase equilibria.

References

Ammar, M.N., Renon, H., 1987. The isothermal flash problem: new methods for phase split calculations. AIChE J. 33, 926–939. https://doi.org/10.1002/aic.690330606

Conn, A.R., Gould, N.I.M., Toint, P.L., 2000. Trust Region Methods, MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898719857

Haugen, K.B., Firoozabadi, A., Sun, L., 2011. Efficient and robust three-phase split computations. AIChE J. 57, 2555–2565. https://doi.org/10.1002/aic.12452

Heidemann, R.A., 1983. Computation of high pressure phase equilibria. Fluid Phase Equilib. 14, 55–78. https://doi.org/10.1016/0378-3812(83)80115-0

Imai, M., Pan, H., Connolly, M., Tchelepi, H., Kurihara, M., 2019. Reduced variables method for four-phase equilibrium calculations of hydrocarbon-water-CO_2 mixtures at a low temperature. Fluid Phase Equilib. 497, 151–163. https://doi.org/10.1016/j.fluid.2019.06.002

Johns, R.T., Orr, F.M., Jr., 1996. Miscible gas displacement of multicomponent oils. SPE J. 1, 39–50. https://doi.org/10.2118/30798-PA

Khan, S.A., Pope, G.A., Sepehrnoori, K., 1992. Fluid characterization of three-phase CO_2/oil mixtures. Presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, Oklahoma, USA, 22–24 April. https://doi.org/10.2118/24130-MS

Li, H., 2021. Multiphase Equilibria of Complex Reservoir Fluids: An Equation of State Modeling Approach, Springer Nature.

Li, Z., Firoozabadi, A., 2012. General strategy for stability testing and phase-split calculation in two and three phases. SPE J. 17, 1096–1107. https://doi.org/10.2118/129844-PA

Lu, C., Jin, Z., Li, H., Xu, L., 2021. Simple and robust algorithm for multiphase equilibrium computations at temperature and volume specifications. SPE J. 26, 2397–2416. https://doi.org/10.2118/205499-PA

Mehra, R.K., Heidemann, R.A., Aziz, K., 1983. An accelerated successive substitution algorithm. Can. J. Chem. Eng. 61, 590–596. https://doi.org/10.1002/cjce.5450610414

Metcalfe, R.S., Yarborough, L., 1979. The effect of phase equilibria on the CO_2 displacement mechanism. SPE J. 19, 242–252. https://doi.org/10.2118/7061-PA

Michelsen, M.L., 1982a. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. https://doi.org/10.1016/0378-3812(82)85001-2

Michelsen, M.L., 1982b. The isothermal flash problem. Part II. Phase-split calculation. Fluid Phase Equilib. 9, 21–40. https://doi.org/10.1016/0378-3812(82)85002-4

Nghiem, L.X., Li, Y.-K., 1984. Computation of multiphase equilibrium phenomena with an equation of state. Fluid Phase Equilib. 17, 77–95. https://doi.org/10.1016/0378-3812(84)80013-8

Nocedal, J., Wright, S.J., 2006. Numerical Optimization, 2nd ed, Springer Series in Operations Research and Financial Engineering, Springer, New York. https://doi.org/10.1007/978-0-387-40065-5

Okuno, R., 2009. Modeling of Multiphase Behavior for Gas Flooding Simulation. Ph.D. thesis, University of Texas at Austin.

Orr, F.M., Jr., Yu, A.D., Lien, C.L., 1981. Phase behavior of CO_2 and crude oil in low-temperature reservoirs. SPE J. 21, 480–492. https://doi.org/10.2118/8813-PA

Pan, H., Connolly, M., Tchelepi, H., 2019. Multiphase equilibrium calculation framework for compositional simulation of CO_2 injection in low-temperature reservoirs. Ind. Eng. Chem. Res. 58, 2052–2070. https://doi.org/10.1021/acs.iecr.8b05229

Pan, H., Firoozabadi, A., 1998. Complex multiphase equilibrium calculations by direct minimization of gibbs free energy by use of simulated annealing. SPE Reserv. Eval. Eng. 1, 36–42. https://doi.org/10.2118/37689-PA

Pan, H., Imai, M., Connolly, M., Tchelepi, H., 2021. Solution of multiphase Rachford-Rice equations by trust region method in compositional and thermal simulations. J. Pet. Sci. Eng. 200, 108150. https://doi.org/10.1016/j.petrol.2020.108150

Perschke, D.R., 1988. Equation of State Phase Behavior Modeling for Compositional Simulatio. Ph.D. thesis, University of Texas at Austin.

Petitfrere, M., Nichita, D.V., 2014. Robust and efficient trust-region based stability analysis and multiphase flash calculations. Fluid Phase Equilib. 362, 51–68. https://doi.org/10.1016/j.fluid.2013.08.039

Petitfrere, M., Nichita, D.V., 2016. On a choice of independent variables in Newton iterations for multiphase flash calculations. Fluid Phase Equilib. 427, 147–151. https://doi.org/10.1016/j.fluid.2016.06.050

Rachford, H.H., Jr., Rice, J.D., 1952. Procedure for use of electronic digital computers in calculating flash vaporization hydrocarbon equilibrium. J. Pet. Tech. 4, 19–3. https://doi.org/10.2118/952327-G

Robinson, D.B., Peng, D., 1978. The Characterization of the Heptanes and Heavier Fractions for the GPA Peng-Robinson Programs, Gas Processors Association, Tulsa, Oklahoma.

Young, L.C., 2022. Compositional reservoir simulation: a review. SPE J. 1–47. https://doi.org/10.2118/208610-PA

Appendices of Chapter 2

2-A.1 Partial derivatives showing up in Eqs. 2-9, 2-16, and 2-17

The corresponding Jacobian matrix elements in Eq. 2-9 are given by,

$$\frac{\partial g_{ik}}{\partial lnK_{jp}} = \delta_{ij}\delta_{kp} + \sum_{s=1}^{nc} \frac{\partial ln\varphi_{ik}}{\partial x_{sk}} \frac{\partial x_{sk}}{\partial lnK_{jp}} - \sum_{s=1}^{nc} \frac{\partial ln\varphi_{i,np}}{\partial x_{s,np}} \frac{\partial x_{s,np}}{\partial lnK_{jp}} \tag{2-A1}$$

$$\frac{\partial g_{ik}}{\partial \theta_p} = \sum_{s=1}^{nc} \frac{\partial ln\varphi_{ik}}{\partial x_{sk}} \frac{\partial x_{sk}}{\partial \theta_p} - \sum_{s=1}^{nc} \frac{\partial ln\varphi_{i,np}}{\partial x_{s,np}} \frac{\partial x_{s,np}}{\partial \theta_p} \tag{2-A2}$$

$$\frac{\partial R_k}{\partial lnK_{jp}} = x_{jp}\left(\delta_{kp} - \theta_p \frac{x_{jk} - x_{j,np}}{z_j}\right) \tag{2-A3}$$

$$\frac{\partial R_k}{\partial \theta_p} = -\sum_{i=1}^{nc} \frac{(x_{ik} - x_{i,np})(x_{im} - x_{i,np})}{z_i} \tag{2-A4}$$

$$\frac{\partial x_{sk}}{\partial lnK_{jp}} = \delta_{sj} x_{sk}\left(\delta_{kp} - \frac{n_{sp}}{z_s}\right) \tag{2-A5}$$

$$\frac{\partial x_{s,np}}{\partial lnK_{jp}} = -\delta_{sj} x_{s,np} \frac{n_{sp}}{z_s} \tag{2-A6}$$

$$\frac{\partial x_{sk}}{\partial \theta_p} = -x_{sk} \frac{(x_{sp} - x_{s,np})}{z_s} \tag{2-A7}$$

$$\frac{\partial x_{s,np}}{\partial \theta_p} = -x_{s,np} \frac{(x_{sp} - x_{s,np})}{z_s} \tag{2-A8}$$

The partial derivatives showing up in the analytical gradients defined by Eqs. 2-16, and 2-17 are calculated as per the following equations,

$$\frac{\partial n_{sk}}{\partial lnK_{jp}} = \theta_k \frac{\partial x_{sk}}{\partial lnK_{jp}} \tag{2-A9}$$

$$\frac{\partial n_{s,np}}{\partial lnK_{jp}} = \left(1 - \sum_{l=1}^{np-1} \theta_l\right) \frac{\partial x_{s,np}}{\partial lnK_{jp}} \tag{2-A10}$$

$$\frac{\partial lnf_{sk}}{\partial lnK_{jp}} = \sum_{i=1}^{nc} \left(\frac{\partial ln\varphi_{sk}}{\partial x_{ik}} + \frac{\delta_{si}}{x_{sk}}\right) \frac{\partial x_{ik}}{\partial lnK_{jp}} \tag{2-A11}$$

$$\frac{\partial lnf_{s,np}}{\partial lnK_{jp}} = \sum_{i=1}^{nc} \left(\frac{\partial ln\varphi_{s,np}}{\partial x_{i,np}} + \frac{\delta_{si}}{x_{s,np}}\right) \frac{\partial x_{i,np}}{\partial lnK_{jp}} \tag{2-A12}$$

$$\frac{\partial n_{sk}}{\partial \theta_p} = \delta_{pk} x_{sk} + \frac{\partial x_{sk}}{\partial \theta_p} \theta_k \tag{2-A13}$$

$$\frac{\partial n_{s,np}}{\partial \theta_p} = -x_{s,np} + \frac{\partial x_{s,np}}{\partial \theta_p}\left(1 - \sum_{l=1}^{np-1} \theta_l\right) \tag{2-A14}$$

$$\frac{\partial lnf_{sk}}{\partial \theta_p} = \sum_{i=1}^{nc} \left(\frac{\partial ln\varphi_{sk}}{\partial x_{ik}} + \frac{\delta_{si}}{x_{sk}}\right) \frac{\partial x_{ik}}{\partial \theta_p} \tag{2-A15}$$

$$\frac{\partial lnf_{sk}}{\partial \theta_p} = \sum_{i=1}^{nc} \left(\frac{\partial ln\varphi_{s,np}}{\partial x_{i,np}} + \frac{\delta_{si}}{x_{s,np}}\right) \frac{\partial x_{i,np}}{\partial \theta_p} \tag{2-A16}$$

2-A.2 Fluid properties

Table 2-A1 to **Table 2-A6** list the properties of the reservoir fluids that are examined in this study.

Table 2-A1. Fluid properties of the JEMA oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.92	100	44.01	304.2	73.76	0.225	0
C_1	6.93	0	16.04	166.67	46	0.008	0.05
C_{2-3}	17.42	0	36.01	338.81	45.53	0.126	0.05
C_{4-6}	19.44	0	70.52	466.12	33.68	0.244	0.05
C_{7-16}	31.38	0	147.18	611.11	20.95	0.639	0.09
C_{17-29}	15.49	0	301.48	777.78	15.88	1	0.09
C_{30+}	7.42	0	562.81	972.22	15.84	1.281	0.09

Table 2-A2. Fluid properties of the MRO oil and the injected gas (Orr et al., 1981; Haugen et al., 2011).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with CH_4
CO_2	0	100	44	304.211	73.819	0.225	-	-
C_1	29.39	0	16	190.6	45.4	0.008	0.115	-
C_2	10.19	0	30.1	305.4	48.2	0.098	0.115	-
C_3	8.35	0	44.1	369.8	41.9	0.152	0.115	-
nC_4	3.31	0	58.1	425.2	37.5	0.193	0.115	-
C_{5-7}	12.04	0	89.9	516.667	28.82	0.2651	0.115	0.045
C_{8-10}	15.81	0	125.7	590	23.743	0.3644	0.115	0.055
C_{11-14}	8.23	0	174.4	668.611	18.589	0.4987	0.115	0.055
C_{15-20}	5.28	0	240.3	745.778	14.8	0.6606	0.115	0.06
C_{21-28}	2.76	0	336.1	812.667	11.954	0.8771	0.115	0.08
C_{29+}	4.64	0	536.7	914.889	8.523	1.2789	0.115	0.28

Table 2-A3. Fluid properties of the NWE oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	0.77	95	44.01	304.2	73.76	0.225	0
C_1	20.25	5	16.04	190.6	46	0.008	0.12
C_{2-3}	11.8	0	38.4	343.64	45.05	0.13	0.12
C_{4-6}	14.84	0	72.82	466.41	33.5	0.244	0.12
C_{7-14}	28.63	0	135.82	603.07	24.24	0.6	0.12
C_{15-24}	14.9	0	257.75	733.79	18.03	0.903	0.12
C_{25+}	8.81	0	479.95	923.2	17.26	1.229	0.12

Table 2-A4. Fluid properties of the Oil G and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.69	100	44.01	304.2	73.76	0.225	0
C_1	17.52	0	16.043	174.44	46	0.008	0.085
C_{2-3}	22.44	0	37.9086	347.26	44.69	0.1331	0.085
C_{4-6}	16.73	0	68.6715	459.74	34.18	0.2358	0.085
C_{7-14}	24.22	0	135.0933	595.14	21.87	0.5977	0.104
C_{15-25}	12.16	0	261.103	729.98	16.04	0.9118	0.104
C_{26+}	5.24	0	479.6983	910.18	15.21	1.2444	0.104

Table 2-A5. Fluid properties of the acid gas sample and the injected gas (Pan and Firoozabadi, 1998; Haugen et al., 2011).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with N_2	BIP with H_2S
CO_2	0	100	44	304.211	73.819	0.225	0	0	0
N_2	23.91	0	28	126.2	33.9	0.039	–0.02	0	0
H_2S	6.69	0	34.1	373.2	89.4	0.081	0.12	0.2	0
C_1	23.36	0	16	190.564	45.992	0.01141	0.125	0.031	0.1
C_2	35.94	0	30.1	305.322	48.718	0.10574	0.135	0.042	0.08
C_3	10.10	0	44.1	369.825	42.462	0.15813	0.15	0.091	0.08

Table 2-A6. Fluid properties of the MY10 and water mixture (Metcalfe and Yarborough, 1979; Johns and Orr, 1996; Imai et al., 2019).

Components	Composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with H_2O	BIP with CO_2	BIP with CH_4	BIP with C_2	BIP with C_3	BIP with C_4
H_2O	28.50	16	647	220.5	0.344	-	-	-	-	-	-
CO_2	5.00	44.01	304.04	73.84	0.225	0.095	-	-	-	-	-
CH_4	23.275	16.04	190.59	46.04	0.01	0.45	0.1	-	-	-	-
C_2	1.995	30.07	305.21	48.84	0.099	0.5	0.13	0	-	-	-
C_3	2.66	44.1	369.71	42.57	0.152	0.5	0.135	0	0	-	-
C_4	3.99	58.12	419.04	37.46	0.187	0.5	0.13	0	0	0	-
C_5	2.66	72.15	458.98	32.77	0.252	0.5	0.125	0	0	0	0
C_6	1.995	86.18	507.54	29.72	0.296	0.5	0.12	0.02	0.03	0.03	0.03
C_7	3.325	100.2	540.32	27.37	0.351	0.5	0.12	0.03	0.03	0.03	0.03
C_8	3.325	114.23	568.93	25.1	0.394	0.5	0.12	0.035	0.03	0.03	0.03
C_{10}	19.95	142.29	615.15	22.06	0.491	0.5	0.12	0.04	0.03	0.03	0.03
C_{14}	3.325	198.39	694.82	15.86	0.755	0.5	0.12	0.06	0.03	0.03	0.03

CHAPTER 3   A TRUST-REGION-BASED PHASE ENVELOPE CONSTRUCTION ALGORITHM

A version of this chapter has been submitted to *Journal of Petroleum Science and Engineering* for possible publication on Jan 20, 2023.

Abstract

Pressure-temperature (PT) and pressure-composition (Px) phase diagrams play an important role in various chemical and petroleum engineering applications. Lots of algorithms based on the Newton-Raphson method have been proposed to numerically construct PT/Px phase envelopes. No attempt is made to improve the computational efficiency of such Newton-Raphson solution method. This work develops a new trust-region-based algorithm to replace the Newton-Raphson method to provide more robust and efficient phase envelope construction calculations. We first convert the phase envelope construction problem into a minimization problem by defining a least-squares objective function. We then apply the trust-region optimization method with an exact subproblem solver to minimize the objective function. We follow the general strategy proposed by Michelsen (1980) to develop the revised phase envelope construction algorithm. We demonstrate the good performance of the trust-region-based algorithm by comparing it with the conventional Newton's method. The computational results indicate that the new trust-region-based algorithm is more robust and cost-effective than the Newton-based algorithm.

Keywords: Phase envelope; Trust-region method; Phase behavior; Compositional simulation

3.1 Introduction

Pressure-temperature (PT) and pressure-composition (Px) phase diagrams play an important role in guiding chemical and petroleum engineering in various aspects, such as experimental parameter estimations, simulation and process designs, and flow assurance. Practically, there are two approaches to numerically generating a PT/Px diagram. One approach is to build a high-resolution digital diagram, where each pixel represents the number of equilibrium phases at a specified PT/Px condition (Agger and Sørensen, 2018). This approach is computationally expensive since hundreds of thousands of PT equilibrium calculations are involved to build such a diagram with an acceptable resolution (Michelsen, 1982b; Pan et al., 2019). Compared to this approach, the other cost-effective approach, which only traces the phase envelopes with tens or hundreds of envelope point (EP) calculations (Michelsen, 1980), becomes more attractive.

The milestone work by Michelsen (1980) introduced an iterative algorithm for constructing a phase envelope numerically. This iterative algorithm contains an EP calculation solver as the computational kernel. To solve an EP calculation problem, Michelsen (1980) applied a full Newton (NT) method (i.e., the Newton-Raphson method for nonlinear equations) to solve a set of $nc + 2$ equations (where nc represents the number of compounds in the mixture); these equations include an important specification equation. The procedure of this numerical algorithm begins with working out a known low-pressure dew point as the first EP. Then a second EP is generated based on the first EP, and a subsequent EP calculation will be initialized based on the previous two EPs. The complete phase envelope can be generated in such an iterative manner (Michelsen, 1980).

Since its publication, the algorithm developed by Michelsen (1980) has been directly extended to different scenarios. An early attempt was to extend this algorithm to build Px, temperature-composition (Tx), and composition-composition envelopes (Li and Nghiem, 1982). In more recent studies, it was extended to construct three-phase envelopes containing an aqueous phase (Pedersen et al., 1996; Lindeloff and Michelsen, 2003) and three-phase envelopes involving asphaltene precipitation (Agger and Sørensen, 2018; Cismondi, 2018).

Other scholars tried to modify or change the formulations of the EP calculation problem for different computing needs. Kunz et al. (2007) proposed a volume-based formulation with volume, temperature, and mole numbers as independent variables. Cismondi and Michelsen (2007a; 2007b) developed a global algorithm to calculate critical loci, critical end points, and three-phase liquid-liquid-vapor equilibrium lines and another numerical algorithm to construct Pxy and Txy phase diagrams; both algorithms are dedicated to binary mixtures. Cismondi et al. (2008) further considered the calculation of the loci of homogeneous azeotropy of binary mixtures. Venkatarathnam (2014a, b) proposed a practical density-marching method based on a density-based specification equation. Nikolaidis et al. (2016) proposed a so-called bead-string method by replacing the specification equation with a derivative-related equation. Nichita (2018) developed a density-based formulation for phase envelope constructions based on the molar densities of individual components. Deiters and Bell (2019) incorporated ordinary differential equations to construct phase envelopes of binary mixtures. Medeiros et al. (2022) built phase envelopes for reactive systems with the RAND formulation. Another novel modification is to use different independent variables for constructing PT, Px, and Tx diagrams and find out the optimal ones (Nikolaidis et al., 2019; Nikolaidis et al., 2020).

In the aforementioned methods, the major innovation is the improvement of the formulations. Few modifications are made to the solution method of the EP calculation problem. Michelsen's full NT method (Michelsen, 1980) has been directly applied without change for decades. To our knowledge, we are lacking a new solution method that can provide more robust and efficient phase envelope calculations than the NT method. In fact, the trust-region (TR) method has been successfully applied to solve the stability test (Michelsen, 1982a) and flash calculation problems (Michelsen, 1982b). It is shown to significantly increase the robustness and efficiency of phase equilibrium calculations (Petitfrere and Nichita, 2014; Pan et al., 2019; Li, 2021). Therefore, it is worthwhile exploring the capability of applying the TR method to phase envelope constructions.

Our motivation is to introduce the robust TR method to solve the two-phase EP calculation problem defined by Michelsen (1980) and develop a general phase envelope construction algorithm based on the TR method. The paper is structured as follows. We first present the methodology of the TR-based envelope construction algorithm in the "Methodology" section. Then case studies involving the comparison between the TR method and the conventional NT method are shown in the "Results and discussion" section. At last, we draw major conclusions in the "Conclusions" section.

3.2 Methodology

3.2.1 Formulation

For a two-phase PT envelope, we work out each EP by solving the nonlinear equation set given by (Michelsen, 1980),

$$F_i(u) = ln(K_i) - ln\varphi_{x,i} + ln\varphi_{y,i}, i = 1,2,\cdots,nc, \tag{3-1}$$

$$F_{nc+1}(u) = \sum_{i=1}^{nc}(x_i - y_i), \tag{3-2}$$

$$F_{nc+2}(u) = S_{spec} - S, \tag{3-3}$$

In **Eq. 3-2**, x_i and y_i are, respectively, given by,

$$x_i = z_i \tag{3-4}$$

$$y_i = z_i K_i \tag{3-5}$$

Eq. 3-3 is called the specification equation (Michelsen, 1980). In Eq. 3-3, variable S is chosen to be one variable in vector u. That is,

$$S \in u = [lnK_1, \cdots, lnK_{nc}, lnT, lnP]^T \tag{3-6}$$

For a two-phase Px EP, the nonlinear equation set keeps unaltered, but the independent variables are changed to,

$$u_{px} = [lnK_1, \cdots, lnK_{nc}, r, lnP]^T \tag{3-7}$$

where r ($0 \leq r \leq 1$) is the injection gas fraction. Parameter r satisfies,

$$z_i(r) = (1 - r)z_{o,i} + rz_{g,i} \tag{3-8}$$

where z_o and z_g represent the oil and injection gas compositions. Then we obtain,

$$\frac{\partial z_i}{\partial r} = z_{g,i} - z_{o,i} \tag{3-9}$$

For a general independent variable vector u, the equation set to be solved can be summarized as (Michelsen, 1980),

$$F(u, S) = 0 \tag{3-10}$$

Appendix 3-A.1 summarizes the Jacobian matrix of **Eq. 3-10** for the PT specification. The Jacobian matrix for the Px specification can be easily worked out based on Appendix 3-A.1 and **Eq. 3-9**.

3.2.2 Trust-region method

The equation set defined by Eq. 3-10 can be directly solved by the NT method (Michelsen, 1980). A Newton step (s_{NT}) is yielded by solving,

$$Js_{NT} = -F \tag{3-11}$$

The numerical procedure of the NT method is covered in Appendix 3-A.3. Different from the simple form of the NT method, preprocessing is needed when applying the TR method to change an equation-solving problem to a minimization problem. Specifically, we define a new least-square merit function as the objective function to be minimized by the TR method,

$$M(u) = \sum_{i=1}^{nc+2} 0.5\|F_i\|^2 \tag{3-12}$$

The gradient g of M satisfies (Sun and Yuan, 2006),

$$g_i = \sum_{i=1}^{nc+2} F_i \nabla F_i \tag{3-13}$$

Letting J be the Jacobian of $F = (F_1, \cdots, F_{\mathrm{nc}+2})^T$,

$$J = \begin{pmatrix} \frac{\partial F_1}{\partial u_1} & \cdots & \frac{\partial F_1}{\partial u_{nc+2}} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_{nc+2}}{\partial u_1} & \cdots & \frac{\partial F_{nc+2}}{\partial u_{nc+2}} \end{pmatrix} \tag{3-14}$$

we can have (Sun and Yuan, 2006),

$$g = J^T F \tag{3-15}$$

The Hessian matrix B can be approximated as (Nocedal and Wright, 2006; Sun and Yuan, 2006),

$$B \approx J^T J \tag{3-16}$$

The TR method is a numerical algorithm to locate the minimum of the objective function based on the solution of the TR subproblem (Conn et al., 2000; Nocedal and Wright, 2006; Sun and Yuan, 2006). For the function $M(u)$, the method defines a "trusted" region (i.e., the region bounded by the TR radius Δ) where the function M can be represented by a quadratic model m (i.e., the TR subproblem). We then find a minimizer s (i.e., the TR step) for m and evaluate the TR ratio ρ.

To obtain s at a certain iteration level n, we seek a solution to the TR subproblem defined as (Nocedal and Wright, 2006),

$$m^n(s^n) = 0.5\|F^n + J^n s^n\|^2 = M^n + s^n(J^n)^T F^n + 0.5 s^n (J^n)^T J^n s^n \tag{3-17}$$

The TR ratio ρ is determined by (Nocedal and Wright, 2006),

$$\rho^n = \frac{M(u^n) - M(u^n + s^n)}{m^n(0) - m^n(s^n)} \tag{3-18}$$

If $\rho > 0.75$, we obtain a good agreement between the actual function reduction and the predicted model reduction, indicating that we can expand the TR radius Δ. If $\rho < 0.25$, contrarily, the radius should be shrunk. If ρ is larger than a specified tolerance $\gamma \in [0,0.25)$, the TR step is acceptable, and the variables will be updated by $u = u + s$; otherwise, the step is rejected. In such an iterative manner, the TR method continues until the computational error is lower than the specified tolerance tol.

We closely follow the state-of-the-art TR method, which has been implemented in the multiphase equilibrium calculations (Petitfrere and Nichita, 2014; Pan et al., 2019), to develop the new TR-based EP calculation algorithm. To enhance the algorithm robustness, we switch to a successive substitution (SS) step when the objective function does not meet the requirement for a TR step at certain iteration level. Note that such a SS step is slightly different from the one being used in the flash calculations as only equilibrium ratios are updated. For the TR subproblem, we apply the realistic exact solver proposed by Conn et al. (2000). The numerical procedure of this TR subproblem solver is summarized in Appendix 3-A.2.

Fig. 3-1 summarizes the numerical procedure of the TR method for EP calculations. As discussed in Petitfrere and Nichita (2014), an important and sensitive parameter to be set up is the initial TR radius Δ_0. In our implementation, the initial TR radius is chosen to be $\Delta_0 = mean(|\boldsymbol{lnK}|)$. This choice of the initial TR radius provides generally robust EP calculations. But different values may be necessary for other types of applications such as the stability test and flash calculation (Petitfrere and Nichita, 2014). In this study, we directly use $\gamma = 0$ as we will calculate relatively good initial guesses for the TR method. The same setting can be found in Pan et al. (2019). The computational error is calculated by $err = \|F\|$. The calculations of the initial guesses based on extrapolation schemes will be introduced in the next section.

```
Algorithm 3-1: the trust-region method for the two-phase PT/Px EP calculation problem
Given $\Delta_0 > 0,\ tol > 0,\ \Delta \in [0, \Delta_0],\ n = 1,\ err^n = 1.$
While $err^n > tol$
  $\gamma \in [0, \Delta/4]$
  Calculate $M^n,\ g^n$
  Calculate Hessian matrix
  Calculate $\rho^n,\ s^n$, and $\lambda$ using the exact solution method for trust-region subproblems (Conn et al., 2000)
  If $\rho^n > 0.75$ and $\|s^n\| = \Delta$
    $\Delta = min(2\Delta, \Delta_0)$
  End if
  If $\rho^n < 0.25$
    $\Delta = \Delta/4$
  End if
  If $\rho^n > \gamma$
    Update $u^n$ by a trust-region step
  Else
    Update the equilibrium ratios in $u^n$ by a SS step
  End if
  Update $err^n$
  $n = n + 1$
End while
```

Fig. 3-1. A pseudo-code of the trust-region method tailored for the two-phase EP calculation problem.

3.2.3 Numerical algorithm for phase envelope constructions

3.2.3.1 Algorithm basics

The conventional numerical algorithm for phase envelope constructions is first proposed by Michelsen (1980). Since the first EP is known a priori, this algorithm starts working out the second EP. The calculation of the second EP will be initialized based on the first EP, and a subsequent EP will be initialized based on the previous two EPs.

There are two important parts to be initialized in the algorithm. The first is to determine the specification S based on the sensitivity vector at the previous EP (Michelsen and Mollerup, 2007). By differentiating Eq. 3-10 and applying the chain rule, we can obtain (Michelsen, 1980),

$$J(u)\frac{\partial(u)}{\partial S}+\frac{\partial(F)}{\partial S}=0 \qquad \text{(3-19)}$$

Assuming that the solution of Eq. 3-10 is u^*, the vector $\frac{\partial(u)}{\partial S}|_{u^*}$ can be determined easily, because

$$\frac{\partial(F)}{\partial S}=[0,\cdots,0,-1]^T \qquad \text{(3-20)}$$

The vector $\frac{\partial(u)}{\partial S}|_{u^*}$ is the so-called sensitivity vector, which reveals the gradient information of u with respect to the specification variable S. We define $d=\frac{\partial(u)}{\partial S}|_{u^*}$ for the sake of simplicity. Based on Michelsen and Mollerup (2007), the magnitude of all the elements of d will be limited to be equal to or less than 1. Thus, the index of S (i.e., the parameter id_{spec}) keeps unchanged if all the elements of $|d|$ are smaller than 1. If one or more elements of $|d|$ are larger than 1, S is changed to the variable associated with the largest element of $|d|$. Note that it is a convention to specify pressure ($ln\,(P)$) as the specification variable in the first EP calculation (Michelsen and Mollerup, 2007).

The second part is to generate the initial guesses of u using polynomial extrapolation schemes. Michelsen (1980) proposed the following linear extrapolation scheme,

$$u_{init1}^{k} = u^{k-1} + d^{k-1}\Delta S^{k-1} \tag{3-21}$$

where k is the EP index. In fact, there is another possible linear scheme as shown below,

$$u_{init2}^{k} = u^{k-1} + \frac{u^{k-1} - u^{k-2}}{u_{id_{spec}^{k}}^{k-1} - u_{id_{spec}^{k}}^{k-2}}\Delta S^{k-1} \tag{3-22}$$

The values of k are limited to be $k > 1$ and $k > 2$ in **Eqs. 3-21** and **3-22**, respectively. Furthermore, we can generate a third-order polynomial extrapolation scheme defined as (Michelsen, 1980; Michelsen and Mollerup, 2007; Agger and Sørensen, 2018),

$$u_{poly,i}^{k}(S) = C_{i,1} + C_{i,2}S + C_{i,3}S^2 + C_{i,4}S^3 \tag{3-23}$$

where $k > 2$. The coefficients C_i are determined by fitting the polynomial function with u^{k-1}, u^{k-2}, $d^{k-1}(S^{k-1}), and\ d^{k-2}(S^{k-2})$. After fitting the polynomial function, we can obtain,

$$u_{init3}^{k} = u_{poly,i}^{k}(S^{k-1} + \Delta S^{k-1}) \tag{3-24}$$

Eqs. 3-21, 3-22, and **3-24** define three different extrapolation schemes, any of which can be treated as the initial guess for solving Eq. 3-10 with a gradient-based method (i.e., the NT method or the TR method). The selections of the extrapolation schemes are further discussed in Appendix 3-A.3. Last but not least, the step size ΔS^k will be adjusted at each iteration level k based on the total number of iterations needed for converging the previous EP. Specifically, ΔS is updated by,

$$\Delta S^k = l\Delta S^{k-1} \tag{3-25}$$

The detailed values of l will be discussed in the following sections.

3.2.3.2 Numerical procedure

We follow Michelsen's general strategy (Michelsen, 1980; Michelsen and Mollerup, 2007) to construct the PT/Px phase envelopes but provide more numerical details based on our own experiences. **Fig. 3-2** shows the overall numerical procedure of the phase envelope construction algorithm. The construction algorithm starts with a pre-defined EP at $P = 1\ bar$ and an initial step size (such as $\Delta S^1 = ln(2) - ln(1)$) to proceed to the second EP. In Fig. 3-2, Algorithm 3-A1, the iterative sub-algorithm for the EP determination, employs the numerical TR/NT algorithm (i.e., Algorithm 3-1 or Algorithm 3-A2) to converge the EP calculations. The numerical procedure of Algorithm 3-A1 is summarized in Appendix 3-A.3. **Table 3-1** shows a set of recommended multipliers l inspired by the works by Michelsen and Mollerup (2007) and Agger and Sørensen (2018).

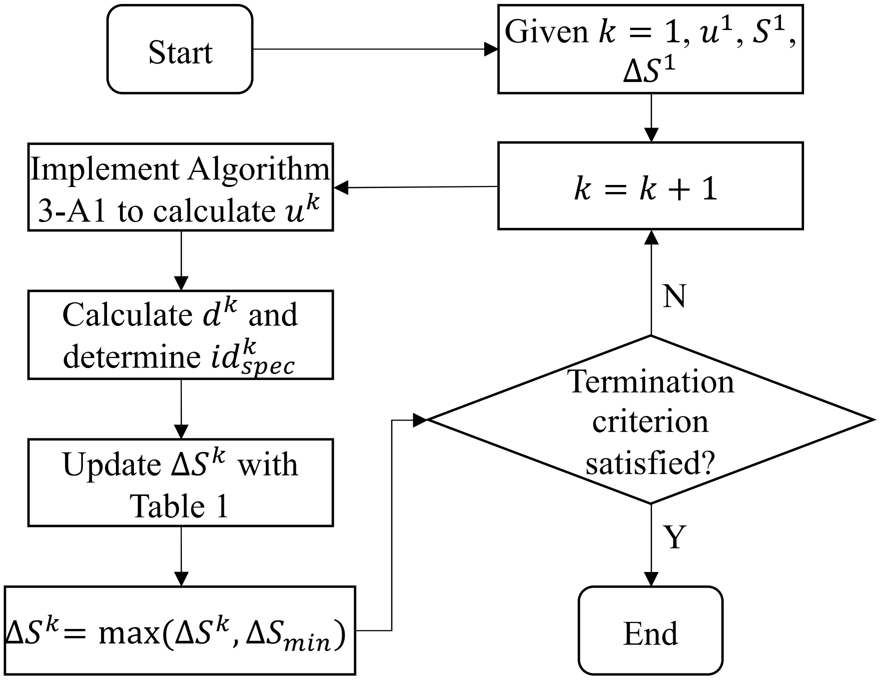


Fig. 3-2. Numerical procedure of the phase envelope construction algorithm.

Table 3-1. Values of the step-size multipliers used to update the step size.

$\Delta T/\Delta P$ condition in the PT specification		Number of iterations for the EP calculation in the PT specification				
$max(\Delta T^k, \Delta P^k) > 100$ (ΔT in K, ΔP in bar)	$max(\Delta T^k, \Delta P^k) < 2.5$ (ΔT in K, ΔP in bar)	1	2	3	4	≥5
0.5	1.5	2	1.5	1.2	0.9	0.5
$\Delta r/\Delta P$ condition in the Px specification		Number of iterations for the EP calculation in the Px specification				
$max\left(\Delta r^k, \frac{\Delta P^k}{500\ bar}\right) > 0.1$	$max\left(\Delta r^k, \frac{\Delta P^k}{500\ bar}\right) < 0.005$	1	2	3	4	≥5
0.5	1.5	2	1.5	1.2	0.9	0.5

As shown in Table 3-1, we apply additional step-size adjustment multipliers (based on the $\Delta T/\Delta P$ in the PT specification or $\Delta r/\Delta P$ in the Px specification between the current and last EPs) to prevent unfavorable step sizes, such as an excessively large step size. Moreover, we set a minimum threshold for the step size ΔS to avoid very small ΔS in a near-critical region. We design a parameter called critical-region flag (cf) to determine if an EP is inside the near-critical region. The critical-region flag is defined as,

$$cf^k = \frac{\max\left(\left\{K_i^k\right\}\right)}{min\left(\left\{K_i^k\right\}\right)} - 1 \tag{3-26}$$

In our implementation, $cf < 0.2$ signifies that the current EP is inside the near-critical region. Inspired by Agger and Sørensen (2018), we set the minimum ΔS to be the largest value of $|\boldsymbol{lnK}|$ at the last EP before entering the near-critical region.

3.3 Results and discussion

In this section, we calculate 15 two-phase PT envelopes, and 2 two-phase Px envelopes for a total of 17 oil-gas mixtures. These oil-gas mixtures include: n-decane mixed with pure nitrogen, 5-component hydrocarbon mixture (Michelsen, 1980), MY10 oil sample mixed with pure CO_2 (Metcalfe and Yarborough, 1979; John and Orr, 1996), Oil G mixed with pure CO_2 (Khan et al., 1992), NWE oil sample mixed with 95 mol% CO_2

and 5 mol % CH_4 (Khan et al., 1992), MRO oil sample mixed with pure CO_2 (Orr et al. 1981; Haugen et al., 2011), JEMA oil sample mixed with pure CO_2 (Khan et al. 1992), Zick-1 oil sample mixed with Zick-1 gas and Zick-2 oil sample mixed with Zick-2 gas (Zick, 1986; Jessen et al., 1998), slightly volatile oil sample (Høier-SVO) mixed with Høier-3 gas (Høier, 1997), and near-critical oil sample (NC-P) (Pedersen and Christensen, 2006). **Table 3-2** shows the basic information of the 17 case studies, while Appendix 3-A.4 shows the fluid properties of the 17 fluid mixtures as well as their phase envelopes calculated by the newly proposed algorithm. In the following calculations, we will compare the TR method with the NT method by constructing the same envelopes. Then the phase envelopes calculated by the TR method will be validated by the point-to-point multiphase equilibrium calculations. We adopt Peng–Robinson equation of state (Robinson and Peng, 1978) in all the calculations. The phase envelope construction code is run with MATLAB® (The MathWorks, Inc., Natick, Massachusetts, USA) using a laptop personal computer with a 3.1 GHz Intel® Core i7-6700HQ (Intel Corporation, Santa Clara, California, USA) central processing unit with 16 GB RAM.

Table 3-2. Basic information of the example fluid mixtures.

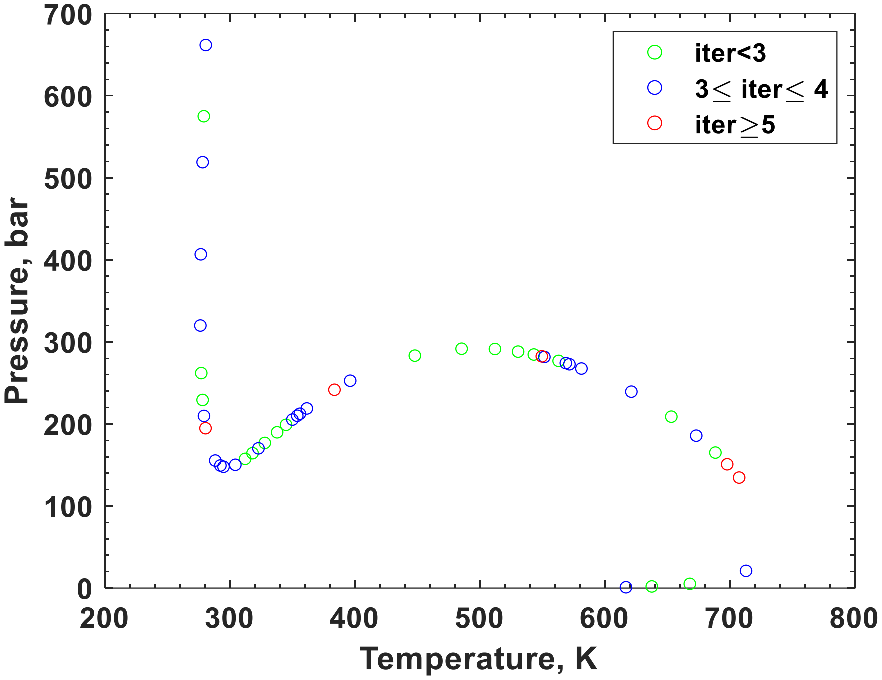
Case index	Oil type	Gas type	Gas mole fraction, mol%	Total number of components	Type of phase diagram	References
N2C10	C_{10}	N_2	50	2	PT	-
M5	M5	-	-	5	PT	Michelsen, 1980; Li and Firoozabadi, 2012
OilG0.7	Oil G	CO_2	70	7	PT	Khan et al., 1992
NWE0.7	NWE Oil	95 mol% CO_2 + 5 mol% CH_4	70	7	PT	Khan et al., 1992
JEMA0.6	JEMA Oil	CO_2	60	7	PT	Khan et al. 1992
JEMA0.75			75			
MRO0.55	MRO	CO_2	55	11	PT	Orr et al. 1981; Haugen et al., 2011
MRO0.7			70			

MY10	MY10	CO_2	0	11	PT	Metcalfe and Yarborough, 1979; John and Orr, 1996
MY10-0.2			20			
MY10-0.8			80			
NC-P	NC-P	-	-	16	PT	Pedersen and Christensen, 2006
Zick0.2	Zick-2	Zick-2	20	12	PT	Zick, 1986; Jessen et al., 1998
Zick-Px	Zick-1	Zick-1	-		Px	
Høier0.05	Høier-SVO	Høier-3	5	15	PT	Høier, 1997
Høier0.2			20		PT	
Høier-Px			-		Px	

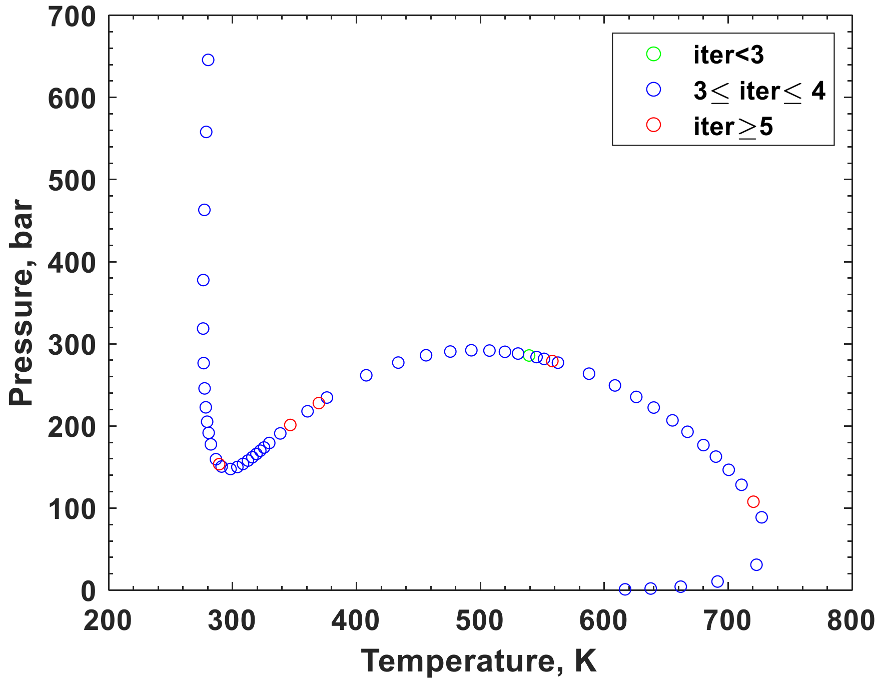
3.3.1 Computational efficiency and robustness

To make a fair comparison between the TR method and the NT method, we use the same numerical construction algorithm to construct the envelopes of the aforementioned fluids. Specifically, we use the numerical algorithm stated in Appendix 3-A.3 to implement both the TR and NT methods, and the basic algorithm settings keep unchanged. Other important settings that are not included in Appendix 3-A.3 include: (1) We use the step-size multipliers listed in Table 3-1; (2) The termination criterion of a PT envelope construction is $P > 600\ bar$ or $P < 1\ bar$, while the termination criterion of a Px envelope construction is $r > 0.99$; (3) The termination criterion of an EP calculation is set to $err = \|F\| < tol = 10^{-9}$.

We first compare the calculated EP sets by the TR and NT methods. **Fig. 3-3** and **Fig. 3-4** show the EP sets calculated by the two methods for cases MRO0.55 and Høier0.05, respectively. We can observe from these figures that the TR method requires generally fewer iterations for convergence, leading to larger step-size changes. As such, the TR method is able to consume less time for the phase envelope construction than the NT method. Quantitatively, the algorithm based on the TR method consumes 3.42 s in constructing the phase envelope (including 45 EP calculations) in Fig. 3-3a, while the NT method consumes 4.42 s in constructing the phase envelope (including 57 EP calculations) in Fig. 3-3b. Fig. 3-4 shows similar results, where the TR method consumes 4.2 s (including 39 EP calculations) and the NT method consumes 8.31 s (including 70 EP calculations). Note that the construction of the low-pressure bubble line for case Høier0.05 becomes more difficult as there are significant magnitude changes in the equilibrium ratios of heavy hydrocarbon components. Thus, the algorithm with a faster convergence rate, i.e., the TR algorithm in this case, is preferred.

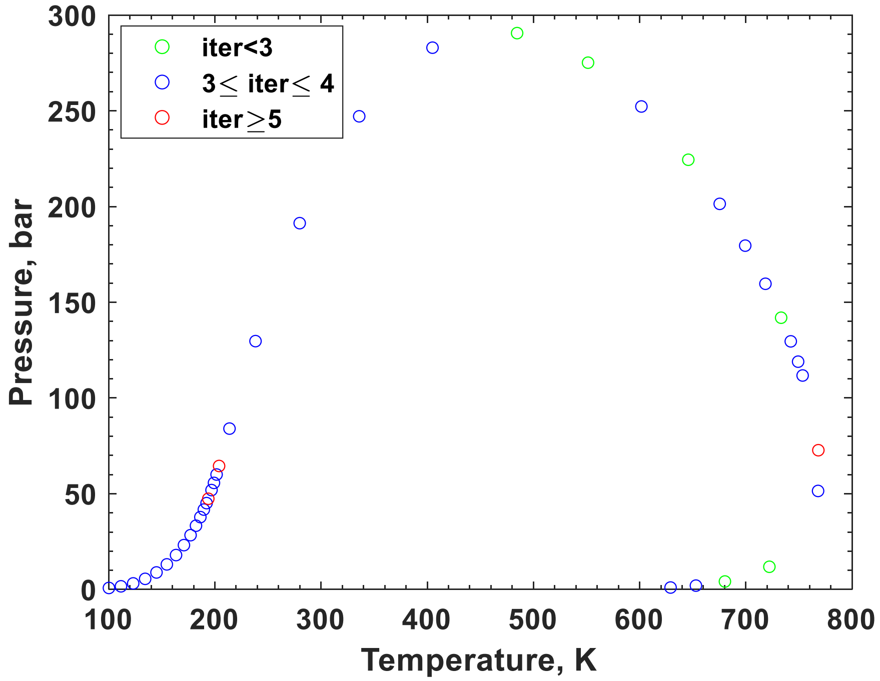


(a)

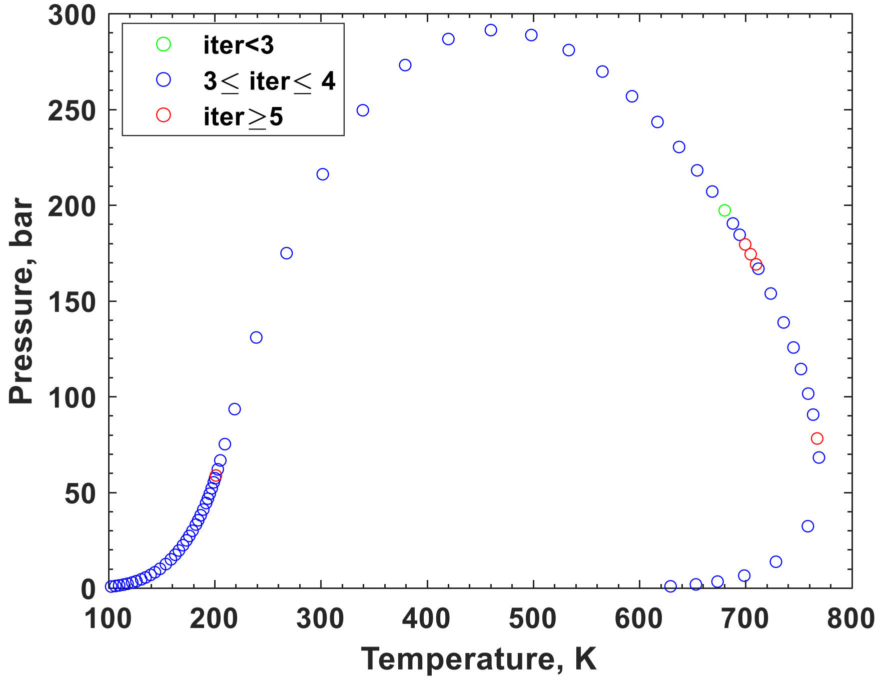


(b)

Fig. 3-3. Calculated EPs for case MRO0.55: (a): EPs calculated by the TR method; (b) EPs calculated by the NT method. Green, blue, and red circles represent EPs that require small (<3), medium (3 or 4), and large (≥5) iteration numbers to reach convergence, respectively.



(a)



(b)

Fig. 3-4. Calculated EPs for case Høier0.05: (a): EPs calculated by the TR method; (b) EPs calculated by the NT method. Green, blue, and red circles represent EPs that require small (<3), medium (3 or 4), and large (≥5) iteration numbers to reach convergence, respectively.

Table 3-3 summarizes the quantitative comparison of the two methods when they are used to construct the 17 phase envelopes. In general, the TR method is more cost-effective than the conventional NT method in constructing the same two-phase envelopes. Then we calculate the differences in the number of EPs, the total number of iterations, and the computational costs between the two methods. As shown in Table 3-3, the TR method can always reduce the total number of iterations. We observe that there is a positive correlation between the decrease in the number of EPs and the decrease in the computational time, indicating that the TR method becomes more timesaving if more EPs can be reduced. When the TR method cannot reduce the involved EP number (such as in cases M5 and MRO0.7), however, the computational time will become at most 0.02 seconds larger than the NT method. The primary cause is that Newton-Raphson method requires only the solution of Eq. 3-10, while the TR method solves both Eq. 3-10 and the TR subproblem in each iteration. Because of this, the average time per iteration of the TR method may become larger than that of the NT method, resulting in the two methods consuming almost the same time in these two cases. Even if we observe negative "reductions" as listed in Table 3-3, however, the absolute computational time differences in the two cases remain small. This indicates the TR method will hardly incur additional computational load in these cases. Moreover, the TR method still outperforms the NT method in other cases. On average, as compared to the NT method, the trust region method reduces 13.8 EPs, 66 iterations, and 27.43% computational time to construct a two-phase envelope.

Table 3-3. Quantitative comparison between the TR method and NT method that are used to construct the 17 phase envelopes with the multiplier values shown in Table 3-1.

Case index	Number of EPs		Number of solver calls		Total iteration number in the solver		Computational time, s		Decrease in the number of EPs	Decrease in the total iteration number	Percentage decrease in computational time, %
	NT	TR	NT	TR	NT	TR	NT	TR			
N2C10	27	9	29	9	92	28	0.61	0.48	18	64	21.97
M5	23	23	24	27	83	66	0.84	0.86	0	17	-2.62
OilG0.7	32	23	35	26	127	70	1.50	1.10	9	57	26.84
NWE0.7	31	22	33	25	113	71	1.33	1.09	9	42	18.39
JEMA0.6	54	27	58	31	193	83	2.45	1.27	27	110	48.30
JEMA0.75	28	19	30	20	106	60	1.28	0.96	9	46	24.69
MRO0.55	57	45	61	53	203	135	4.42	3.42	12	68	22.64
MRO0.7	29	27	32	37	124	110	2.05	2.05	2	14	0.10
MY10	38	24	40	24	132	69	2.49	1.58	14	63	36.58
MY10-0.2	41	26	42	28	141	74	2.93	1.80	15	67	38.40
MY10-0.8	57	39	62	45	220	131	4.52	2.96	18	89	34.51
NC-P	48	36	56	47	188	119	5.76	4.26	12	69	26.04
Zick0.2	71	35	72	37	246	106	6.29	2.82	36	140	55.22
Zick-Px	32	26	32	27	96	67	2.20	1.86	6	29	15.47
Høier0.05	70	39	72	41	253	121	8.31	4.20	31	132	49.38
Høier0.2	58	52	62	60	200	146	6.65	5.42	6	54	18.57
Høier-Px	43	32	44	32	134	67	4.32	2.94	11	67	31.91
								Average	13.82	66.35	27.43

We use MATLAB® Profiler (The MathWorks, Inc., Natick, Massachusetts, USA) to further analyze the detailed computational costs incurred by the subroutines of

Algorithm 3-A1. **Fig. 3-5** shows the time-cost percentages of different subroutines in the two algorithms implementing the TR method and the NT method. It is obvious that, for both algorithms, the extrapolation subroutine is the most time-consuming subroutine. Specifically, about 60% of the computational time is consumed to regress the 3rd-order polynomial extrapolation model in both algorithms. We can further reduce the computational time of the phase envelope construction algorithm if we can optimize the time spent on the extrapolation task.

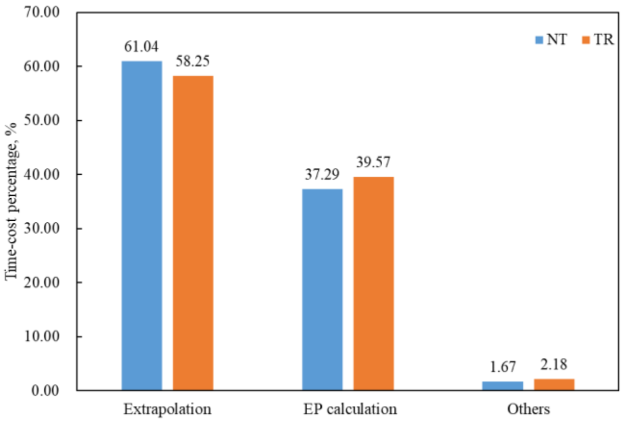


Fig. 3-5. Time-cost percentages of different subroutines in the algorithms applying the TR method and the NT method.

We conduct another batch of comparative analysis between the NT method and the TR method that use a new set of multipliers. **Table 3-4** lists the values of the new multipliers. The original five values related to the number of iterations are slightly reduced from 2, 1.5, 1.2, 0.9, 0.5 (as shown in Table 3-1) to 1.75, 1.25, 1.15, 0.75, 0.35 (as shown in Table 3-4), respectively. **Fig. 3-6** compares the EPs yielded by the original multiplier set and the modified set for case JEMA0.6. It shows that adopting the new set tends to increase the number of EPs, and thus increase the computational cost of phase envelope constructions. We then reconstruct all the phase envelopes involved in the 17 studied cases using the new multiplier set.

Table 3-4. An alternative set of step-size multipliers used to update the step size.

$\Delta T/\Delta P$ condition in the PT specification		Number of iterations for the EP calculations in the PT specification				
$max(\Delta T^k, \Delta P^k) > 100$ (ΔT in K, ΔP in bar)	$max(\Delta T^k, \Delta P^k) < 2.5$ (ΔT in K, ΔP in bar)	1	2	3	4	≥5
0.5	1.5	1.75	1.25	1.15	0.75	0.35
$\Delta T/\Delta P$ condition in the Px specification		Number of iterations for the EP calculations in the Px specification				
$max\left(\Delta r^k, \frac{\Delta P^k}{500\ bar}\right) > 0.1$	$max\left(\Delta r^k, \frac{\Delta P^k}{500\ bar}\right) < 0.005$	1	2	3	4	≥5
0.5	1.5	1.75	1.25	1.15	0.75	0.35

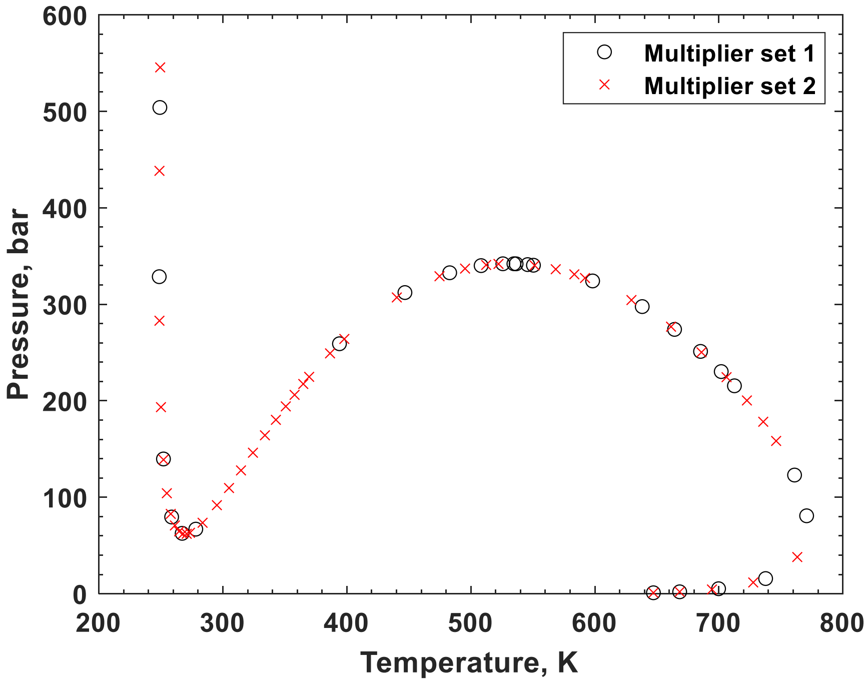


Fig. 3-6. EPs calculated by the TR method with the original set defined in Table 3-1 (marked by black circles) and the new multiplier set defined in Table 3-4 (marked by red crosses) for case JEMA0.6.

When reconstructing the phase envelopes, however, we find the NT algorithm fails in constructing the phase envelope of the case Zick-Px. **Fig. 3-7** shows the EPs yielded by the TR method and NT method for the case Zick-Px. As shown in Fig. 3-7, the EPs generated by the TR algorithm successfully move across the critical point, while the NT algorithm fails to do so.

Fig. 3-8 further compares the critical-region flags calculated with the equilibrium ratios from the TR and NT algorithms for the case Zick-Px. Each point in Fig. 3-8 corresponds to an EP in Fig. 3-7. The value of the critical-region flag reveals the "distance" between an EP and the critical point. As shown in Fig. 3-7 and Fig**. 3-8**, when the NT algorithm marches to the critical point, there is an obvious decrease in the distance between two consecutive EPs. Such a decrease is caused by a reduction of the step size in the NT algorithm. As a result of the step-size reduction, the NT algorithm fails to cross the critical point, yielding an incorrect phase envelope branch.

To examine if the step size is a key factor that leads to this failure, we compare three schemes where three different step-size multipliers are applied when passing the EP (0.732, 226.2 bar). Note that such multipliers are only applied once; the subsequent calculations will follow the original step-size multipliers as specified in Table 3-4. Fig. 3-9 shows the changes in the critical-region flags yielded by the three schemes. As shown in Fig. 3-9, the NT algorithm is not able to move across the critical point unless the step size becomes four times the original one. This result indicates that the step size plays a crucial role in constructing correct two-phase envelopes in the near-critical regions. Moreover, it is demonstrated that the higher convergence rate of the TR method helps yield larger step sizes, making the TR algorithm easier to jump over the critical point than the NT algorithm. That is, the TR method behaves more robustly than the NT method in constructing two-phase envelopes, especially when moving across the critical point.

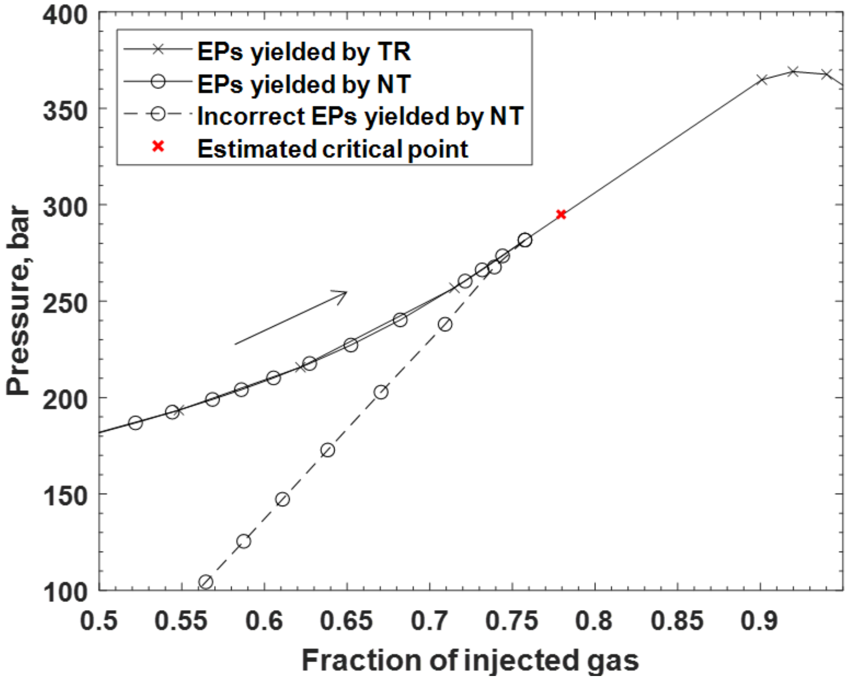


Fig. 3-7. EPs yielded by the TR method (marked by the solid line with crosses) and the NT method (marked by the solid line with circles) for case Zick-Px with the new multipliers as specified in Table 3-4. The arrow shows the marching direction. The dashed line with circles denotes the incorrect phase envelope yielded by the NT algorithm.

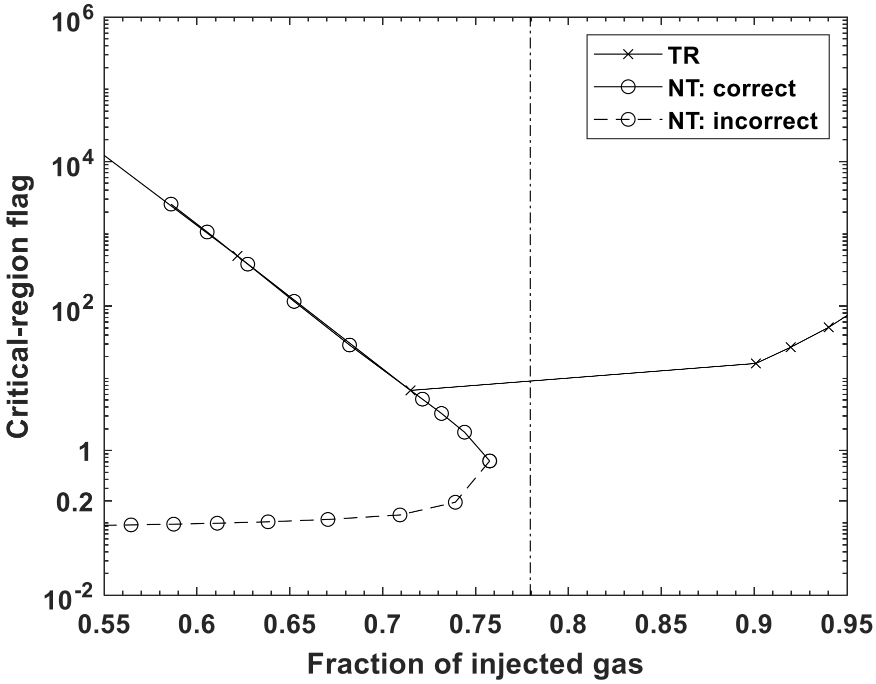


Fig. 3-8. Critical-region flags calculated with the equilibrium ratios yielded by the TR algorithm (marked by solid line with crosses) and the NT algorithm (marked by solid line with circles) for case Zick-Px. The new multipliers specified in Table 3-4 are used. The

dashed line with circles denotes the critical-region flags corresponding to the incorrect EPs yielded by the NT algorithm. The vertical dash-dotted line represents the gas fraction at the critical point.

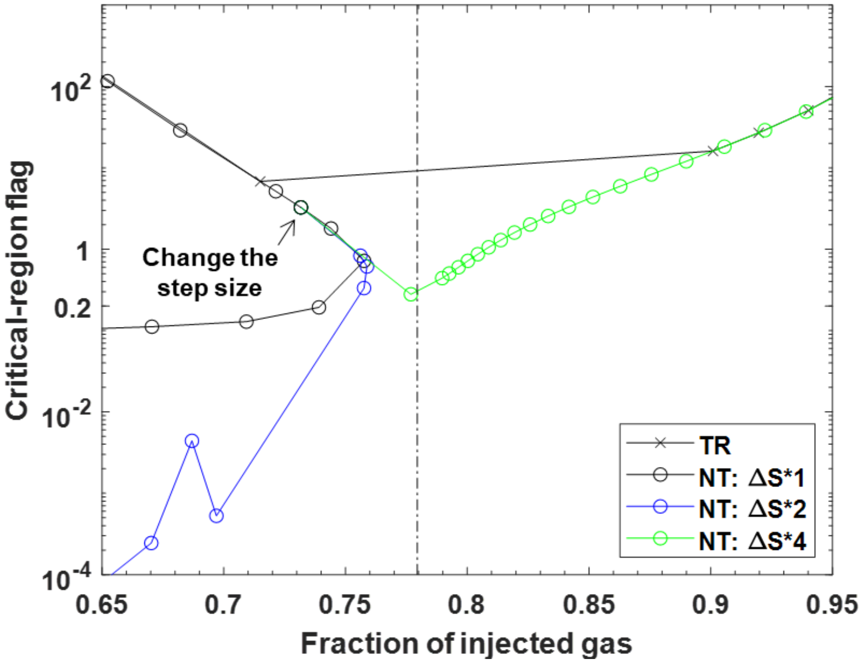


Fig. 3-9. Critical-region flags calculated with the equilibrium ratios yielded by the NT algorithm when passing the EP (0.732, 226.2 bar) in Fig. 3-7 (i.e., the point (0.732, 3.267) in this figure). The black circles represent the results yielded with the original step size *ΔS*. The blue circles represent the results yielded with 2*ΔS*. The green circles represent the results yielded with 4*ΔS*. The black crosses represent the critical-region flags yielded by the TR method. The vertical dash-dotted line represents the gas fraction at the critical point.

Table 3-5 further summarizes the quantitative comparison between the two algorithms when they are used with the new multipliers to construct the 17 phase envelopes. Compared against the NT algorithm, on average, the TR algorithm with the new multipliers could reduce 20.9 EPs, 93.4 iterations, and 34.0% computational time. These three values are all larger than the ones listed in Table 3-3. Besides, different from the results in Table 3-3, we can observe from Table 3-5 that the TR method outperforms the NT method in all the tested cases, including cases M5 and MRO0.7. All these facts, including the one that the NT algorithm fails in constructing the phase envelope for the case Zick-Px, demonstrate that when the envelope construction task becomes more challenging for the NT algorithm, the TR method will be a more competitive option than the NT method. In general, the TR-based algorithm is more computationally efficient and robust than the NT method to construct the 17 two-phase envelopes considered in this study.

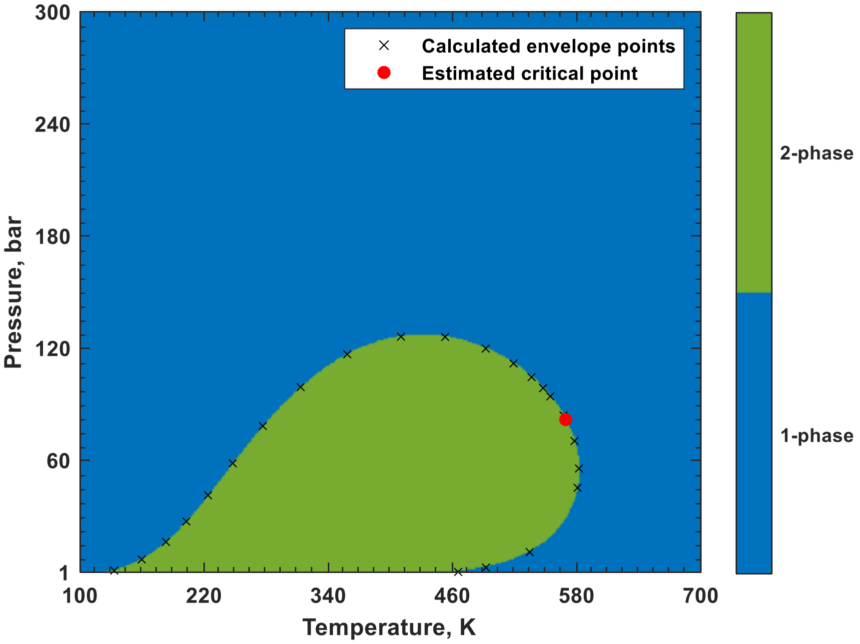
Table 3-5. Quantitative comparison between the TR method and NT method that are used to construct the 17 phase envelopes with the multiplier values shown in Table 3-4.

Case index	Number of EPs		Number of solver calls		Total iteration number in the solver		Computational time, s		Decrease in the number of EPs	Decrease in the total iteration number	Percentage decrease in computational time, %
	NT	TR	NT	TR	NT	TR	NT	TR			
N2C10	31	14	32	14	106	38	0.62	0.51	17	68	17.90
M5	33	25	35	28	113	70	1.05	0.91	8	43	13.81
OilG0.7	35	25	38	27	127	70	1.50	1.16	10	57	22.48
NWE0.7	45	30	46	34	148	83	1.91	1.39	15	65	27.33
JEMA0.6	64	46	58	31	214	128	2.99	2.11	18	86	29.60
JEMA0.75	37	21	39	22	131	60	1.68	1.04	16	71	37.99
MRO0.55	77	55	82	62	261	144	6.12	4.24	22	117	30.63
MRO0.7	37	23	40	32	149	105	2.66	1.78	14	44	33.34
MY10	64	29	70	29	205	74	4.46	1.92	35	131	56.85
MY10-0.2	56	35	57	39	184	90	4.24	2.49	21	94	41.26
MY10-0.8	56	45	59	53	205	144	4.38	3.38	11	61	22.91
NC-P	53	42	61	52	199	128	6.53	5.04	11	71	22.72
Zick0.2	91	42	92	43	303	115	8.28	3.49	49	188	57.86
Zick-Px	-	28	-	29	-	60	-	1.96	-	-	-
Høier0.05	96	51	96	56	328	139	11.46	5.70	45	189	50.22
Høier0.2	73	51	76	57	238	134	8.75	5.72	22	104	34.60
Høier-Px	54	34	55	34	169	64	5.60	3.12	20	105	44.36
								Average	20.88	93.38	33.99

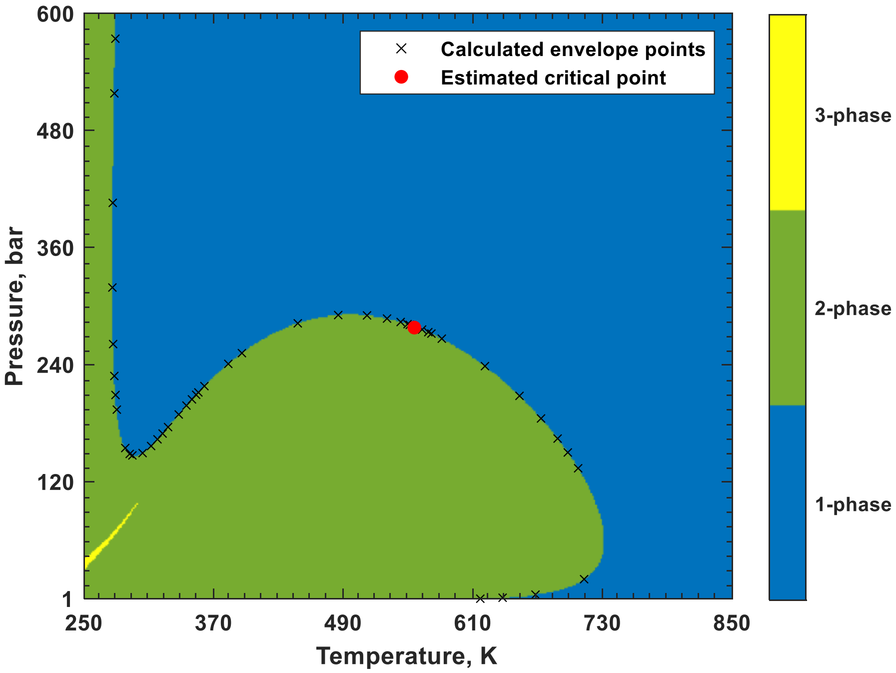
3.3.2 Validation by multiphase equilibrium calculations

The correctness of the calculated EPs can be validated qualitatively by comparing the locations of EPs in the PT/Px space against the phase boundaries demarcated by the multiphase equilibrium calculations (Agger and Sørensen, 2018).

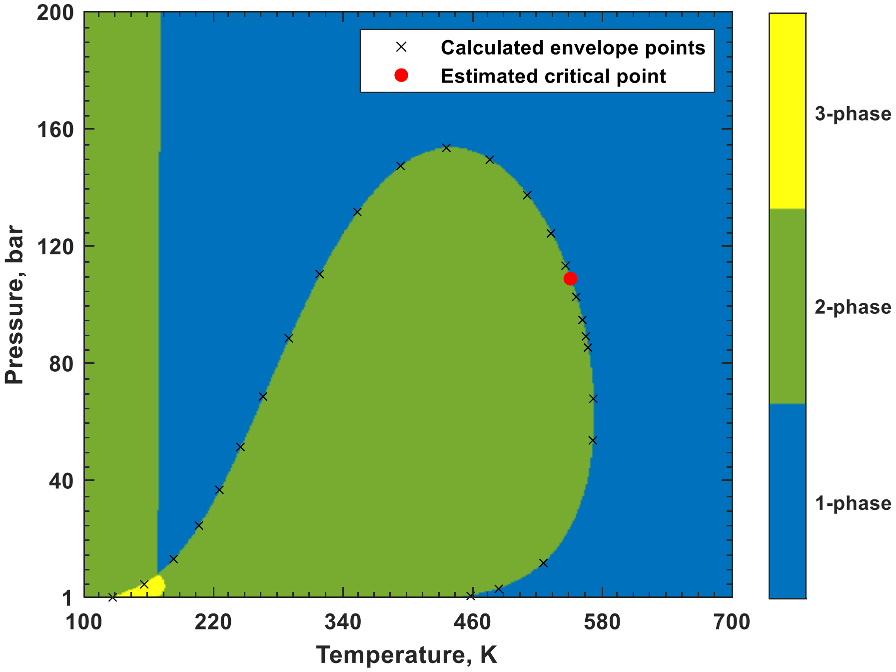
Specifically, we first generate high-resolution digital diagrams using a robust multiphase equilibrium calculation algorithm adopted in one of our previous publications (Lu et al., 2021). This equilibrium calculation algorithm employs the state-of-the-art TR algorithms to safeguard the robustness of stability tests and flash calculations (Petitfrere and Nichita, 2014; Pan et al., 2019). **Fig. 3-10a-e** presents five validation exercises conducted for cases MY10, MRO0.55, MY10-0.2, NC-P, and Zick-Px. Note that these examples are presented to show five representative phase diagrams: a conventional closed two-phase PT envelope, an open two-phase PT envelope with an immiscible phenomenon in the low-temperature region, a special open two-phase PT envelope, a special closed two-phase PT envelope, and a conventional two-phase Px envelope. The EPs points are generated with the multiplier set listed in Table 3-1. The validation results for the remaining cases are listed in Appendix 3-A.4.



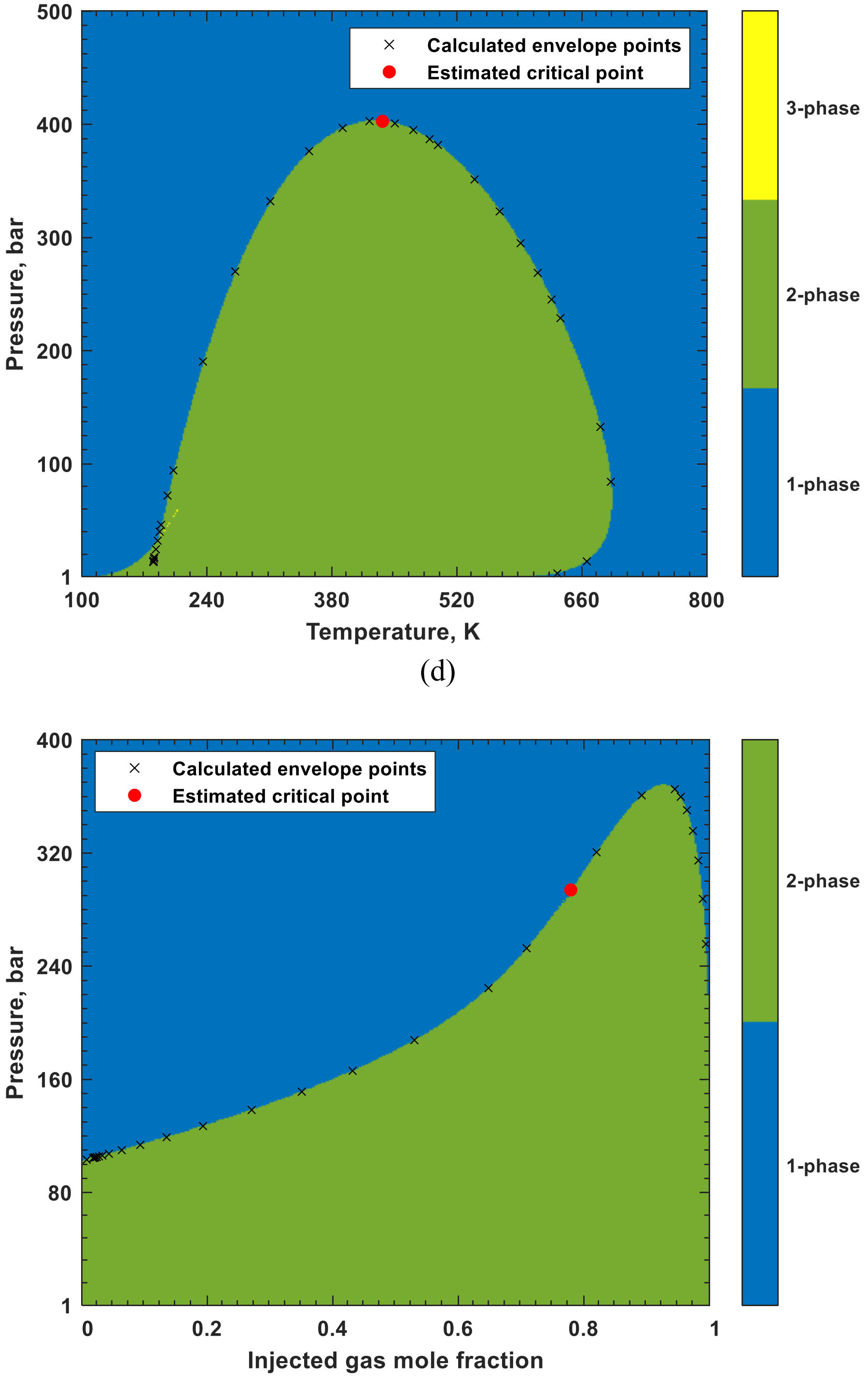
(a)


Calculated envelope points
Estimated critical point
Pressure, bar
Temperature, K
600
480
360
240
120
1
250
370
490
610
730
850
3-phase
2-phase
1-phase

(b)


Calculated envelope points
Estimated critical point
Pressure, bar
Temperature, K
200
160
120
80
40
1
100
220
340
460
580
700
3-phase
2-phase
1-phase

(c)



(d)

(e)

Fig. 3-10. Digital PT and Px phase diagrams, together with the calculated EPs, generated for five representative cases: (a) MY10; (b) MRO0.55; (c) MY10-0.2; (d) NC-P; (e) Zick-Px. Each digital phase diagram is generated by a total of $400 \times 400 = 160{,}000$ runs of multiphase equilibrium calculations. The EPs points are generated with the multiplier set listed in Table 3-1.

In Fig. 3-10a, b, and e, the calculated EPs are fully consistent with the phase boundaries resulted from the multiphase equilibrium calculations. In Fig. 3-10c and d, however, most of the EPs are consistent, while a few EPs do not follow the “true” two-phase envelope as a three-phase region directly contacts the two-phase envelope in the low-temperature region. In these cases, a three-phase point, at which a three-phase region intersects with the two-phase envelope, appears on the two-phase envelope (Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018; Cismondi, 2018). Because of the existence of the three-phase point, the equilibrium ratios on the two sides of this point become totally different. Thus, the current strategy is incapable of detecting the liquid-liquid boundary from the right-hand side of the three-phase point, as shown in Fig. 3-10c, as well as the liquid-vapor boundary from the right-hand side of the three-phase point, as shown in Fig. 3-10d. In Fig. 3-10c, after passing the three-phase point, the algorithm yields solutions that approximately follow the true three-phase bubble line. However, these solutions are considered incorrect as they are solved based on the two-phase formulations. A similar phenomenon appears in Fig. 3-10d, where the algorithm yields incorrect solutions after crossing the three-phase point. To address these issues, a multiphase envelope construction algorithm is required (Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018; Cismondi, 2018). Such an algorithm should be able to handle three-phase envelope constructions.

3.4 Conclusions

We develop a new TR-based algorithm for two-phase envelope constructions. We apply the TR method with an exact subproblem solver to solve the EP calculation problem. With the TR method as the computational kernel, we design an effective numerical procedure to construct two-phase PT and Px envelopes. The example calculations indicate that the newly developed TR-based algorithm is more robust and cost-effective in constructing two-phase PT and Px envelopes than the conventional NT-based algorithm. Although the current version of the TR algorithm can only construct simple two-phase envelopes without the existence of three-phase points, it can be extended to construct complex multiphase envelopes (i.e., phase envelopes containing two-phase envelopes, three-phase envelopes, and three-phase points). Such extension would require new formulations and modifications of the numerical construction algorithm.

Nomenclature

a, b = parameters in Peng-Robinson equation of state
A, B = parameters in Peng-Robinson equation of state
bip = binary interaction parameter
cf = critical-region flag defined by Eq. 3-26
$C, C1, C2, C3$ = coefficients for regression
d = sensitivity vector
err = computational error
F = the nonlinear equation set defined by Eq. 3-10
g = gradient vector
H = Hessian matrix
i, j = component index
id_{spec} = index of the specification variable in vector u
J = Jacobian matrix
k = envelope point index in the phase envelope construction
K = equilibrium ratio
$l, l1, l2$ = multipliers used to increase/decrease ΔS
m = the trust-region model function
M = merit/objective function
n = iteration index in the trust-region method
nc = number of components
P = pressure
r = injection gas fraction
R = gas constant
s = the vector for updating independent variables
S = specification variable
S_a = summation term defined by Eq. 3-A15
S_{a1} = summation term defined by Eq. 3-A14
S_b = summation term defined by Eq. 3-A16

tol = error tolerance
T = temperature
u = independent variables defined by Eq. 3-6
u_{poly} = third-order polynomial estimation of u
u_{px} = independent variables defined by Eq. 3-7
u^* = solution of the independent variables
w = vector solved in the trust-region subproblem
x, y = phase composition
z = feed
z_o = oil composition
z_g = gas composition
Z = compressibility factor in Peng-Robinson equation of state
α = alpha function in Peng-Robinson equation of state
γ = the tolerance for accepting a trust-region step
δ = delta function
Δ = trust-region radius
ΔS = the step size of the specification variable
ε = small positive quantity
λ = eigenvalue
ρ = trust-region ratio
φ = fugacity coefficient
∇ = differential operator

References

Agger, C.S., Sørensen, H., 2018. Algorithm for constructing complete asphaltene PT and Px phase diagrams. Ind. Eng. Chem. Res. 57, 392–400. https://doi.org/10.1021/acs.iecr.7b04246

Cismondi, M., 2018. Phase envelopes for reservoir fluids with asphaltene onset lines: an integral computation strategy for complex combinations of two- and three-phase behaviors. Energy Fuels 32, 2742–2748. https://doi.org/10.1021/acs.energyfuels.7b02790

Cismondi, M., Michelsen, M.L., 2007a. Global phase equilibrium calculations: Critical lines, critical end points and liquid–liquid–vapour equilibrium in binary mixtures. J. Supercrit. Fluids. 39, 287–295. https://doi.org/10.1016/j.supflu.2006.03.011

Cismondi, M., Michelsen, M., 2007b. Automated calculation of complete Pxy and Txy diagrams for binary systems. Fluid Phase Equilib. 259, 228–234. https://doi.org/10.1016/j.fluid.2007.07.019

Cismondi, M., Michelsen, M.L., Zabaloy, M.S., 2008. Automated generation of phase diagrams for binary systems with azeotropic behavior. Ind. Eng. Chem. Res. 47, 9728–9743. https://doi.org/10.1021/ie8002914

Conn, A.R., Gould, N.I.M., Toint, P.L., 2000. Trust Region Methods, MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898719857

Deiters, U.K., Bell, I.H., 2019. Calculation of phase envelopes of fluid mixtures through parametric marching. AIChE J. 65, e16730. https://doi.org/10.1002/aic.16730

Haugen, K.B., Firoozabadi, A., Sun, L., 2011. Efficient and robust three-phase split computations. AIChE J. 57, 2555–2565. https://doi.org/10.1002/aic.12452

Høier, L., 1997. Miscibility Variation in Compositionally Grading Reservoirs, Norwegian University of Science and Technology, Trondheim, Norway.

Jessen, K., Michelsen, M., Stenby, E.H., 1998. Global approach for calculating minimum miscibility pressure. Fluid Phase Equilib. 153, 251–263. https://doi.org/10.1016/S0378-3812(98)00414-2

Johns, R.T., Orr, F.M., Jr., 1996. Miscible gas displacement of multicomponent oils. SPE J. 1, 39–50. https://doi.org/10.2118/30798-PA

Khan, S.A., Pope, G.A., Sepehrnoori, K., 1992. Fluid characterization of three-phase CO_2/oil mixtures. Presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, Oklahoma, USA, 22–24 April. https://doi.org/10.2118/24130-MS

Kunz, O., Klimeck, R., Wagner, W., Jaeschke, M., 2007. The GERG-2004 Wide-Range Equation of State for Natural Gases and Other Mixtures, VDI Verlag GmbH, Verlag des Vereins Deutscher Ingenieure, Düsseldorf, Germany.

Li, H., 2021. Multiphase Equilibria of Complex Reservoir Fluids: An Equation of State

Modeling Approach, Springer Nature.

Li, Y.-K., Nghiem, L.X., 1982. The development of a general phase envelope construction algorithm for reservoir fluid studies. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 26–29 September. https://doi.org/10.2118/11198-MS

Li, Z., Firoozabadi, A., 2012. General strategy for stability testing and phase-split calculation in two and three phases. SPE J. 17, 1096–1107. https://doi.org/10.2118/129844-PA

Lindeloff, N., Michelsen, M. L., 2003. Phase envelope calculations for hydrocarbon-water mixtures. SPE J. 8, 298-303. https://doi.org/10.2118/85971-PA

Lu, C., Jin, Z., Li, H., Xu, L., 2021. Simple and robust algorithm for multiphase equilibrium computations at temperature and volume specifications. SPE J. 26, 2397–2416. https://doi.org/10.2118/205499-PA

Medeiros, F.d.A., Stenby, E.H., Yan, W., 2022. Saturation point and phase envelope calculation for reactive systems based on the RAND formulation. Chem. Eng. Sci. 247, 116911. https://doi.org/10.1016/j.ces.2021.116911

Metcalfe R.S., Yarborough, L., 1979. The effect of phase equilibria on the CO_2 displacement mechanism. SPE J., 19, 242-252. https://doi.org/10.2118/7061-PA

Michelsen, M.L., 1980. Calculation of phase envelopes and critical points for multicomponent mixtures. Fluid Phase Equilib. 4, 1-10. https://doi.org/10.1016/0378-3812(80)80001-X

Michelsen, M.L., 1982a. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. https://doi.org/10.1016/0378-3812(82)85001-2.

Michelsen, M.L., 1982b. The isothermal flash problem. Part II. Phase-split calculation, Fluid Phase Equilib. 9, 21-40. https://doi.org/10.1016/0378-3812(82)85002-4

Michelsen, M.L., Mollerup, J.M., 2007. Thermodynamic Models: Fundamentals & Computational Aspects, 2nd ed, Tie-Line Publications, Holte, Denmark.

Nichita, D.V., 2018. Density-based phase envelope construction. Fluid Phase Equilib. 478, 100–113. https://doi.org/10.1016/j.fluid.2018.09.007

Nikolaidis, I.K., Boulougouris, G.C., Peristeras, L.D., Economou, I.G., 2019. Efficient and robust methods for direct saturation point calculations, Fluid Phase Equilib. 500, 112242. https://doi.org/10.1016/j.fluid.2019.112242

Nikolaidis, I.K., Boulougouris, G.C., Peristeras, L.D., Economou, I.G., 2020. Construction of phase envelopes for binary and multicomponent mixtures with Euler-Newton predictor-corrector methods. Fluid Phase Equilib. 505, 112338. https://doi.org/10.1016/j.fluid.2019.112338

Nikolaidis, I.K., Economou, I.G., Boulougouris, G.C., Peristeras, L.D., 2016. Calculation of the phase envelope of multicomponent mixtures with the bead spring method.

AIChE J. 62, 868–879. https://doi.org/10.1002/aic.15064

Nocedal, J., Wright, S.J., 2006. Numerical Optimization, 2nd ed, Springer Series in Operations Research and Financial Engineering, Springer, New York. https://doi.org/10.1007/978-0-387-40065-5

Orr, F.M., Jr., Yu, A.D., Lien, C.L., 1981. Phase behavior of CO_2 and crude oil in low-temperature reservoirs. SPE J. 21, 480–492. https://doi.org/10.2118/8813-PA

Pan, H., Connolly, M., Tchelepi, H., 2019. Multiphase equilibrium calculation framework for compositional simulation of CO_2 injection in low-temperature reservoirs. Ind. Eng. Chem. Res. 58, 2052-2070. https://doi.org/10.1021/acs.iecr.8b05229

Pedersen, K., Christensen, P., 2006. Phase Behavior of Petroleum Reservoir Fluids, CRC Press. https://doi.org/10.1201/9781420018257

Pedersen, K.S., Michelsen, M.L., Fredheim, A.O., 1996. Phase equilibrium calculations for unprocessed well streams containing hydrate inhibitors. Fluid Phase Equilib. 126, 13-28. https://doi.org/10.1016/S0378-3812(96)03142-1

Petitfrere, M., Nichita, D.V., 2014. Robust and efficient trust-region based stability analysis and multiphase flash calculations, Fluid Phase Equilib. 362, 51-68. https://doi.org/10.1016/j.fluid.2013.08.039

Robinson, D.B., Peng, D.Y., 1978. The Characterization of the Heptanes and Heavier Fractions for the GPA Peng-Robinson Programs; Gas Processors Association: Tulsa, Oklahoma.

Sun, W., Yuan, Y.-X., 2006. Optimization Theory and Methods: Nonlinear Programming, Springer Science & Business Media.

Venkatarathnam G., 2014a. Density marching method for calculating phase envelopes, Ind. Eng. Chem. Res. 53, 3723-3730. https://doi.org/10.1021/ie403633d

Venkatarathnam G., 2014b. Density marching method for calculating phase envelopes. 2. Three-phase envelopes, Ind. Eng. Chem. Res. 53, 12122-12128. https://doi.org/10.1021/ie501838y

Zick, A.A., 1986. A combined condensing/vaporizing mechanism in the displacement of oil by enriched gases. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 5–8 October. https://doi.org/10.2118/15493-MS

Appendices of Chapter 3

3-A.1 The Jacobian matrix of Eq. 3-10

The Jacobian matrix of the nonlinear equation set (defined by Eq. 3-10) with respect to the independent variables $(lnK_1, \cdots, lnK_{nc}, lnT, lnP)$ is given by,

$$J = \begin{pmatrix} \frac{\partial F_1}{\partial \ \mathrm{ln}K_1} & \cdots & \frac{\partial F_1}{\partial \ \mathrm{ln}K_{nc}} & \frac{\partial F_1}{\partial \mathrm{ln}T} & \frac{\partial F_1}{\partial \mathrm{ln}P} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{\partial F_{nc}}{\partial \ \mathrm{ln}K_1} & \cdots & \frac{\partial F_{nc}}{\partial \ \mathrm{ln}K_{nc}} & \frac{\partial F_{nc}}{\partial \mathrm{ln}T} & \frac{\partial F_{nc}}{\partial \mathrm{ln}P} \\ \frac{\partial F_{nc+1}}{\partial \ \mathrm{ln}K_1} & \cdots & \frac{\partial F_{nc+1}}{\partial \ \mathrm{ln}K_{nc}} & 0 & 0 \\ 0 & \cdots & 0 & 0 & 1 \end{pmatrix} \quad (3\text{-A1})$$

where,

$$\frac{\partial F_i}{\partial \ \mathrm{ln}K_j} = \delta_{ij} + z_j K_j \frac{\partial ln\varphi_i(y)}{\partial y_j}, \ i,j \in \{1, \cdots, nc\} \quad (3\text{-A2})$$

$$\frac{\partial F_{nc+1}}{\partial \ \mathrm{ln}K_j} = -z_j K_j, \ j \in \{1, \cdots, nc\} \quad (3\text{-A3})$$

$$\frac{\partial F_i}{\partial lnT} = T(\frac{\partial ln\varphi_i(y)}{\partial T} - \frac{\partial ln\varphi_i(x)}{\partial T}), \ i \in \{1, \cdots, nc\} \quad (3\text{-A4})$$

$$\frac{\partial F_i}{\partial lnP} = P\left(\frac{\partial ln\varphi_i(y)}{\partial P} - \frac{\partial ln\varphi_i(x)}{\partial P}\right), \ i \in \{1, \cdots, nc\} \quad (3\text{-A5})$$

The above fugacity-coefficient-related partial derivatives are calculated based on Peng-Robison equation of state (Robinson and Peng, 1978) with Van der waals mixing rules. The expressions of two important terms, i.e., $\frac{\partial ln\varphi_i(x)}{\partial P}$ and $\frac{\partial ln\varphi_i(x)}{\partial T}$, are given by,

$$\frac{\partial ln\varphi_i(x)}{\partial P} = S_{b,i}\frac{\partial Z}{\partial P} - \frac{\frac{\partial Z}{\partial P} - \frac{\partial B}{\partial P}}{Z-B} - \frac{A}{2\sqrt{2}B}\left(\frac{1}{A}\frac{\partial A}{\partial P} - \frac{1}{B}\frac{\partial B}{\partial P}\right)\left(2S_{a1,i} - S_{b,i}\right)\ln\left(\frac{Z+\left(\sqrt{2}+1\right)B}{Z-\left(\sqrt{2}-1\right)B}\right) -$$

$$\frac{A}{B}\left(2S_{a1,i} - S_{b,i}\right)\frac{(Z\frac{\partial B}{\partial P} - B\frac{\partial Z}{\partial P})}{(Z^2+2ZB-B^2)}, \ i \in \{1, \cdots, nc\}. \quad (3\text{-A6})$$

$$\frac{\partial ln\varphi_i(x)}{\partial T} = S_{b,i}\frac{\partial Z}{\partial T} - \frac{\frac{\partial Z}{\partial T} - \frac{\partial B}{\partial T}}{Z-B} - \frac{A}{2\sqrt{2}B}\left(\frac{1}{A}\frac{\partial A}{\partial T} - \frac{1}{B}\frac{\partial B}{\partial T}\right)\left(2S_{a1,i} - S_{b,i}\right)\ln\left(\frac{Z+\left(\sqrt{2}+1\right)B}{Z-\left(\sqrt{2}-1\right)B}\right) -$$

$$\frac{A}{B}\left(2S_{a1,i} - S_{b,i}\right)\frac{\left(Z\frac{\partial B}{\partial T} - B\frac{\partial Z}{\partial T}\right)}{(Z^2+2ZB-B^2)} - \frac{A}{2\sqrt{2}B}\frac{(a\frac{dS_{a,i}}{dT} - S_i\frac{da}{dT})}{a^2}\ln\left(\frac{Z+\left(\sqrt{2}+1\right)B}{Z-\left(\sqrt{2}-1\right)B}\right), \ i \in \{1, \cdots, nc\} \quad (3\text{-A7})$$

where,

$$\frac{\partial Z}{\partial P} = \frac{\partial Z}{\partial A}\frac{\partial A}{\partial P} + \frac{\partial Z}{\partial B}\frac{\partial B}{\partial P} \tag{3-A8}$$

$$\frac{\partial Z}{\partial T} = \frac{\partial Z}{\partial A}\frac{\partial A}{\partial T} + \frac{\partial Z}{\partial B}\frac{\partial B}{\partial T} \tag{3-A9}$$

$$\frac{\partial Z}{\partial A} = \frac{(B-Z)}{3Z^2-2(1-B)Z+(A-2B-3B^2)} \tag{3-A10}$$

$$\frac{\partial Z}{\partial B} = \frac{(-Z^2+2Z(3B+1)+(A-2B-3B^2))}{3Z^2-2(1-B)Z+(A-2B-3B^2)} \tag{3-A11}$$

$$A = \frac{aP}{(RT)^2} \tag{3-A12}$$

$$B = \frac{bP}{RT} \tag{3-A13}$$

$$S_{a1,i} = \frac{S_{a,i}}{a},\ i \in \{1, \cdots, nc\} \tag{3-A14}$$

$$S_{a,i} = \sum_{j=1}^{nc} x_j\, a_{i,j} \tag{3-A15}$$

$$S_{b,i} = \frac{b_i}{b},\ i \in \{1, \cdots, nc\} \tag{3-A16}$$

and,

$$a(T) = \sum_{i=1}^{nc} \sum_{j=1}^{nc} x_i\, x_j\, a_{i,j}(T) \tag{3-A17}$$

$$a_{i,j}(T) = \left(1 - bip_{i,j}\right)\sqrt{a_i(T)a_j(T)},\ i,j \in \{1, \cdots, nc\} \tag{3-A18}$$

$$a_i(T) = a_i(T_c)\alpha(T),\ i \in \{1, \cdots, nc\} \tag{3-A19}$$

$$b = \sum_{i=1}^{nc} x_i\, b_i \tag{3-A20}$$

Other derivative/partial-derivative terms can be easily calculated using the above equations and will not be covered here.

3-A.2 The realistic exact trust-region subproblem solver

In this subproblem solver developed by Conn et al. (2000), the positive definiteness of the Hessian matrix must be satisfied. If not, a diagonal modification to the Hessian matrix is forced to make it positive definite. After that, a conventional Newton step will be calculated and evaluated first. Based on the Newton step calculation, one of three scenarios will be encountered and the trust-region minimizer s will be determined correspondingly. **Fig. 3-A1** illustrates the numerical procedure of the trust-region subproblem solver (Conn et al., 2000; Petitfrere and Nichita, 2014). One can turn to Conn et al. (2000) and Petitfrere and Nichita (2014) for more technical details.

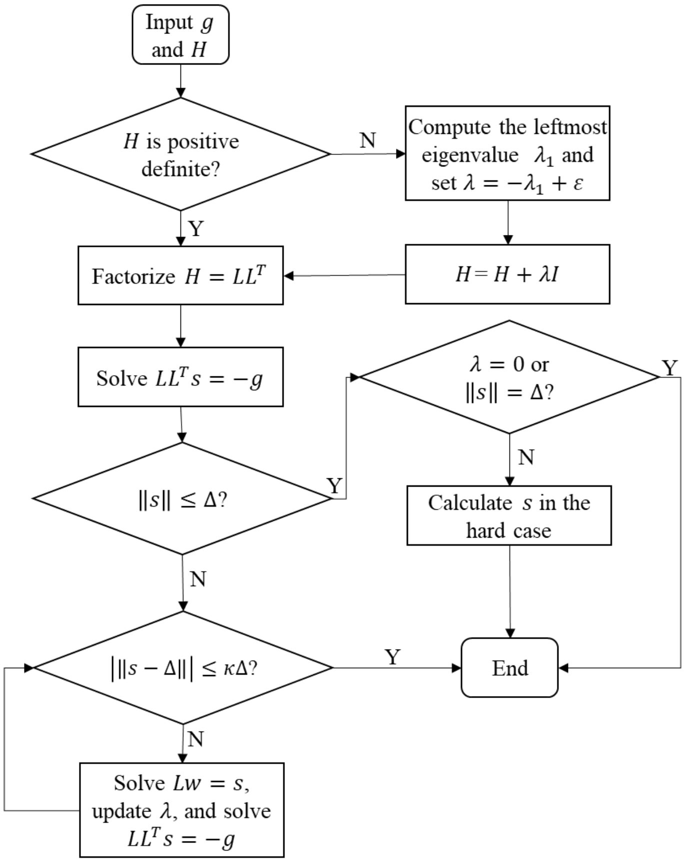


Fig. 3-A1. The numerical procedure of the trust-region subproblem solver proposed by Conn et al. (2000).

3-A.3 Iterative algorithm for the EP determination

To determine a new EP, an extrapolation is suggested (Michelsen, 1980; Michelsen and Mollerup, 2007) to generate the initial guess for a gradient-based method. In our extensive tests, we find that the 3rd order polynomial extrapolation may generate incorrect initial guesses. **Fig. 3-A2** illustrates such a situation for case MRO0.55.

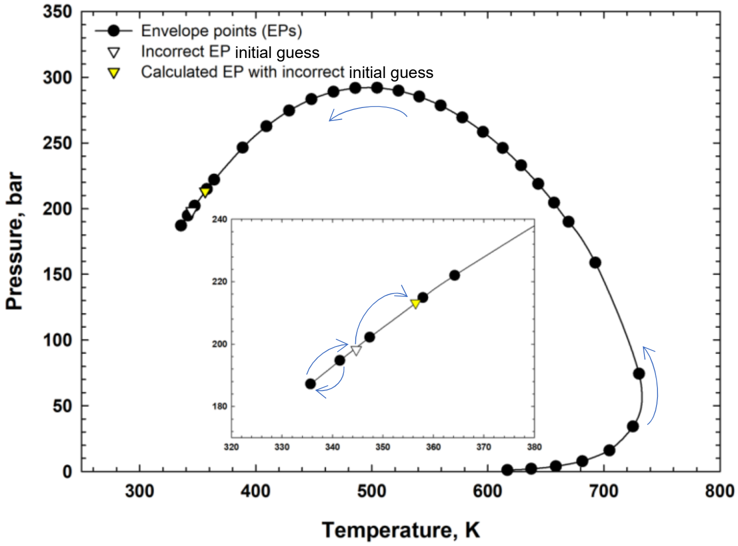


Fig. 3-A2. Calculated envelope points when incorrect extrapolation/convergence appears for MRO0.55. During the construction, an incorrect extrapolation appears, which converges to an EP moving backward on the envelope. A zoomed-in window is inserted in the blank part of the figure to better illustrate this issue. The blue arrows indicate the sequence to generate the points.

As shown in Fig. 3-A2, the construction starts at a low-pressure dewpoint, and the EPs are generated sequentially in a counterclockwise direction. However, we obtain an incorrect 3rd order polynomial extrapolation that goes backwards along the constructed envelope. What's worse, this initial guess converges to a "wrong" EP that moves backward farther along the envelope. If this EP is accepted, the algorithm will trace backwards along the already generated envelope until meeting the beginning point, thus resulting in an incomplete envelope. To avoid this problem, it is practical to replace a 3rd order polynomial extrapolation by a linear one when the above-mentioned problem is detected. Moreover, we apply another linear extrapolation scheme when passing a critical point, which was suggested by Agger and Sørensen (2018).

Except for the extrapolation schemes, the convergence difficulty of the gradient-based methods (e.g., the NT method and the TR method) may be encountered when adopting a large step size in a new EP calculation (Michelsen and Mollerup, 2007). Such a problem can be easily solved by shrinking the step size iteratively until a reasonable size is found. **Fig. 3-A3** illustrates the detailed numerical procedure together with the extrapolation schemes that are implemented to determine a new EP. **Fig. 3-A4** shows the numerical procedure of the NT method that is implemented in Algorithm 3-A1.

```
Algorithm 3-A1: the iterative algorithm for the EP determination
Given $k = 2,\ u^{k-1},\ \Delta S^{k-1},\ id^k_{spec},\ l1 = 0.5,\ l2 = 0.2$
$S^{k-1} = u^{k-1}_{id^k_{spec}}$
Regress a linear model with $u^{k-1}, d^{k-1}$, store the coefficient set $C1$
If $k > 2$
    $S^{k-2} = u^{k-2}_{id^k_{spec}}$
    Regress a linear model with $u^{k-1}, u^{k-2}$, store the coefficient set $C2$
    Regress a third-order polynomial model with $u^{k-1}, u^{k-2}$ and $d^{k-1}(S^{k-1}), d^{k-2}(S^{k-2})$, store the coefficient set $C3$
End If
While
    $S^k = S^{k-1} + \Delta S^{k-1}$; Calculate $cf^{k-1}$
    Calculate a linear extrapolation with $C1$ to generate $u^k_{init1}$
    If $k = 2$
        $u^k_{init}$=$u^k_{init1}$
    Else
        Calculate a linear extrapolation with $C2$ to generate $u^k_{init2}$
        Calculate a third-order polynomial extrapolation with $C3$ to generate $u^k_{init3}$
        $u^k_{init} = u^k_{init3}$
        If $u^k_{init3}$ indicates an EP moving backwards
            $u^k_{init} = u^k_{init1}$
        End If
        If $cf^{k-1} < 0.2$ and ($u^k_{init2}$ passes a critical point or $u^{k-1}$ passes a critical point)
            $u^k_{init} = u^k_{init2}$
        End If
    End If
    $flag = 1$
    If $u^k_{init}$ indicates incorrect orders of magnitude for K-values
        $flag = 0$
    End If
    If $flag = 1$
        Implement a gradient-based method with $u^k_{init}$ (i.e., Algorithm 3-1 or Algorithm 3-A2) to calculate $u^k$
    End If
    If the gradient-based method fails to converge
        $flag = 2$
    End If
    If $flag = 0$
        $\Delta S^{k-1} = \Delta S^{k-1} * l1$
    Else If $flag = 1$
        Break
    Else If $flag = 2$
        $\Delta S^{k-1} = \Delta S^{k-1} * l2$
    End If
End While
```

Fig. 3-A3. The detailed numerical procedure to determine a new EP.

```
Algorithm 3-A2: Newton's method for the two-phase PT/Px EP calculation problem
Given $tol > 0$, $n = 1$, $u^1$.
Calculate $err^1$
While $err^n > tol$
    Solve $Js_{NT} = -F$
    $u^{n+1} = u^n + s_{NT}$
    Update $err^n$
    $n = n + 1$
End while
```

Fig. 3-A4. A pseudo-code of the NT method for the two-phase EP calculation problem.

3-A.4 Example fluid mixtures considered in this study and their phase envelopes calculated by the newly proposed algorithm

Table 3-A1 through **Table 3-A10** list the properties of the hydrocarbon fluids in the 17 cases. **Fig. 3-A5** shows the phase envelopes that are calculated by the developed algorithm as well as the phase boundaries that are calculated by the multiphase equilibrium calculation algorithm.

Table 3-A1. Fluid properties of the N2C10 mixture.

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with C_{10}
N_2	0	100	28	126.2	33.5	0.04	-
C_{10}	100	0	142.285	617.7	20.9	0.488	0.11

Table 3-A2. Fluid properties of the M5 mixture (Michelsen, 1980; Li and Firoozabadi, 2012).

Components	Oil composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with N_2	BIP with H_2S
CO_2	4	44	304.211	73.819	0.225	0	0	0
H_2S	26	34.1	373.2	89.4	0.081	0.12	0.2	0
C_1	66	16	190.564	45.992	0.01141	0.125	0.031	0.1
C_2	3	30.1	305.322	48.718	0.10574	0.135	0.042	0.08
C_3	1	44.1	369.825	42.462	0.15813	0.15	0.091	0.08

Table 3-A3. Fluid properties of the OilG and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.69	100	44.01	304.2	73.76	0.225	0
C_1	17.52	0	16.043	174.44	46	0.008	0.085
C_{2-3}	22.44	0	37.9086	347.26	44.69	0.1331	0.085
C_{4-6}	16.73	0	68.6715	459.74	34.18	0.2358	0.085
C_{7-14}	24.22	0	135.0933	595.14	21.87	0.5977	0.104
C_{15-25}	12.16	0	261.103	729.98	16.04	0.9118	0.104
C_{26+}	5.24	0	479.6983	910.18	15.21	1.2444	0.104

Table 3-A4. Fluid properties of the NWE oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	0.77	95	44.01	304.2	73.76	0.225	0
C_1	20.25	5	16.04	190.6	46	0.008	0.12
C_{2-3}	11.8	0	38.4	343.64	45.05	0.13	0.12
C_{4-6}	14.84	0	72.82	466.41	33.5	0.244	0.12
C_{7-14}	28.63	0	135.82	603.07	24.24	0.6	0.12
C_{15-24}	14.9	0	257.75	733.79	18.03	0.903	0.12
C_{25+}	8.81	0	479.95	923.2	17.26	1.229	0.12

Table 3-A5. Fluid properties of the JEMA oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.92	100	44.01	304.2	73.76	0.225	0
C_1	6.93	0	16.04	166.67	46	0.008	0.05
C_{2-3}	17.42	0	36.01	338.81	45.53	0.126	0.05
C_{4-6}	19.44	0	70.52	466.12	33.68	0.244	0.05
C_{7-16}	31.38	0	147.18	611.11	20.95	0.639	0.09
C_{17-29}	15.49	0	301.48	777.78	15.88	1	0.09
C_{30+}	7.42	0	562.81	972.22	15.84	1.281	0.09

Table 3-A6. Fluid properties of the MRO and the injected gas (Orr et al. 1981; Haugen et al., 2011).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with CH_4
CO_2	0	100	44	304.211	73.819	0.225	-	-
C_1	29.39	0	16	190.6	45.4	0.008	0.115	-
C_2	10.19	0	30.1	305.4	48.2	0.098	0.115	-
C_3	8.35	0	44.1	369.8	41.9	0.152	0.115	-
nC_4	3.31	0	58.1	425.2	37.5	0.193	0.115	-
C_{5-7}	12.04	0	89.9	516.667	28.82	0.2651	0.115	0.045
C_{8-10}	15.81	0	125.7	590	23.743	0.3644	0.115	0.055
C_{11-14}	8.23	0	174.4	668.611	18.589	0.4987	0.115	0.055
C_{15-20}	5.28	0	240.3	745.778	14.8	0.6606	0.115	0.06
C_{21-28}	2.76	0	336.1	812.667	11.954	0.8771	0.115	0.08
C_{29+}	4.64	0	536.7	914.889	8.523	1.2789	0.115	0.28

Table 3-A7. Fluid properties of the MY10 oil and the injected gas (Metcalfe and Yarborough, 1979; John and Orr, 1996).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with CH_4	BIP with C_2	BIP with C_3	BIP with C_4
CO_2	0	100	44.01	304.04	73.84	0.225	0	-	-	-	-
CH_4	35	0	16.04	190.59	46.04	0.01	0.1	-	-	-	-
C_2	3	0	30.07	305.21	48.84	0.099	0.13	0	-	-	-
C_3	4	0	44.1	369.71	42.57	0.152	0.135	0	0	-	-
C_4	6	0	58.12	419.04	37.46	0.187	0.13	0	0	0	-
C_5	4	0	72.15	458.98	32.77	0.252	0.125	0	0	0	0
C_6	3	0	86.18	507.54	29.72	0.296	0.12	0.02	0.03	0.03	0.03
C_7	5	0	100.2	540.32	27.37	0.351	0.12	0.03	0.03	0.03	0.03
C_8	5	0	114.23	568.93	25.1	0.394	0.12	0.035	0.03	0.03	0.03
C_{10}	30	0	142.29	615.15	22.06	0.491	0.12	0.04	0.03	0.03	0.03
C_{14}	5	0	198.39	694.82	15.86	0.755	0.12	0.06	0.03	0.03	0.03

Table 3-A8. Fluid properties of the NC-P oil (Pedersen and Christensen, 2006).

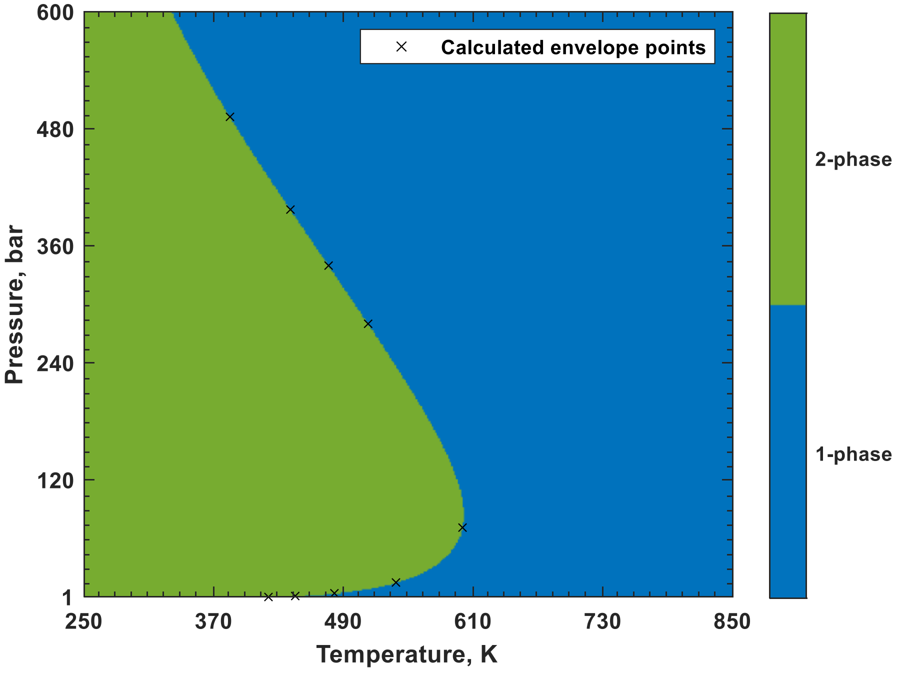
Components	Oil composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with N_2	BIP with CO_2
N_2	0.46	28	126.2	33.94	0.04	0.032	-
CO_2	3.36	44	304.3	73.76	0.225	0.028	0.12
CH_4	62.36	16	190.6	46	0.008	0.041	0.12
C_2	8.9	30.1	305.5	48.84	0.098	0.076	0.12
C_3	5.31	44.1	369.9	42.46	0.152	0.094	0.12
iC_4	0.92	58.1	408.2	36.48	0.176	0.07	0.12
C_4	2.08	58.1	425.3	38	0.193	0.087	0.12
iC_5	0.73	72.2	460.5	33.84	0.227	0.088	0.12
C_5	0.85	72.2	469.7	33.74	0.251	0.08	0.12
C_6	1.05	86.2	507.5	29.69	0.296	0.08	0.1
C_7	1.85	95	532.9	31.95	0.465	0.08	0.1
C_8	1.75	106	553.1	29.44	0.497	0.08	0.1
C_9	1.4	121	575.8	26.08	0.54	0.08	0.1
C_{10}-C_{15}	4.64	165.1	635.0	20.82	0.669	0.08	0.1
C_{16}-C_{25}	2.91	265.1	732.1	15.8	0.919	0.08	0.1
C_{26}-C_{80}	1.43	488.8	908.1	13.06	1.246	0.08	0.1

Table 3-A9. Fluid properties of Zick-1 oil and gas and Zick-2 oil and gas (Zick, 1986; Jessen et al., 1998).

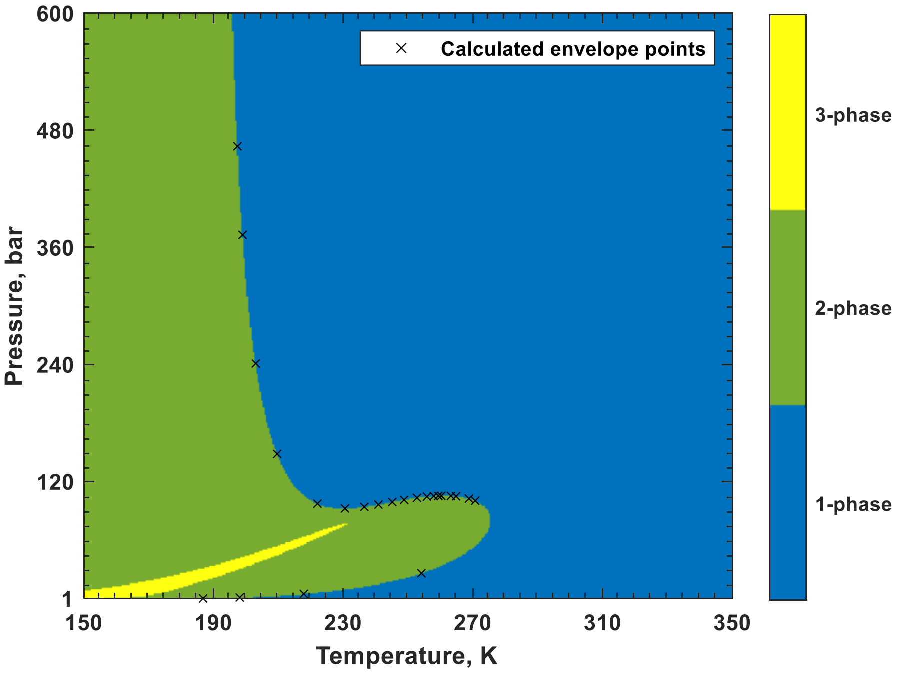
Components	Zick-1 oil composition (mol%)	Zick-1 gas composition (mol%)	Zick-2 oil composition (mol%)	Zick-2 gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	4.49	22.18	6.56	17.75	44.01	304.14	73.74	0.228	-
CH_4	20.71	23.49	37.11	38.78	16.043	190.6	45.92	0.008	0.12
C_2	4.81	23.5	5.38	18.8	30.07	305.4	48.75	0.098	0.15
C_3	4.09	27.45	3.73	21.96	44.097	369.8	42.38	0.152	0.15
C_4	3.23	3.38	2.61	2.71	58.123	425.2	37.93	0.193	0.15
C_5	2.47	0	1.87	0	72.15	469.6	33.68	0.251	0.15
C_6	2.98	0	2.18	0	86.177	507.4	29.64	0.296	0.15
$C_{7+}(1)$	25.25	0	17.91	0	118.3	616.2	28.83	0.454	0.15
$C_{7+}(2)$	12.85	0	9.10	0	172	698.9	19.32	0.787	0.15
$C_{7+}(3)$	8.55	0	6.05	0	236	770.4	16.59	1.048	0.15
$C_{7+}(4)$	6.31	0	4.47	0	338.8	853.1	15.27	1.276	0.15
$C_{7+}(5)$	4.27	0	3.02	0	451	1001.2	14.67	1.299	0.15

Table 3-A10. Fluid properties of the Høier-SVO and the Høier-3 gas (Høier, 1997).

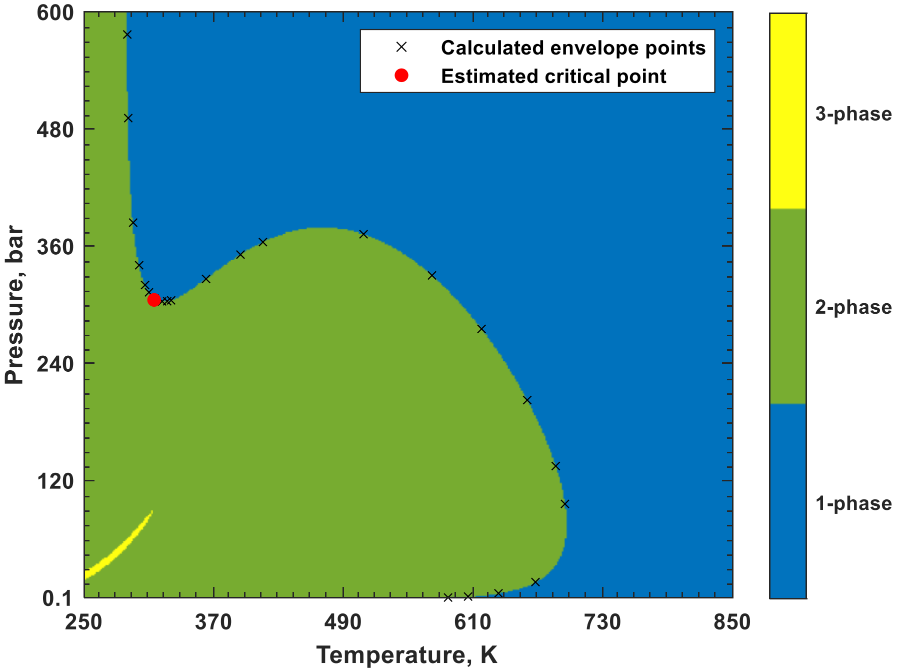
Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with N_2	BIP with CH_4
CO_2	0.265	0.59725	44.01	304.2	73.815	0.231	-	-	-
N_2	0.785	1.58583	28.01	126.3	33.991	0.045	0	-	-
CH_4	45.622	92.8772	16.04	190.6	46.043	0.0115	0.105	0.025	-
C_2	6.092	3.66376	30.07	305.4	48.801	0.0908	0.13	0.01	0
C_3	4.429	0.38875	44.1	369.8	42.492	0.1454	0.125	0.09	0
iC_4	0.865	0.33887	58.12	408.1	36.48	0.1756	0.12	0.095	0
C_4	2.26	0.08508	58.12	425.2	37.969	0.1928	0.115	0.095	0
iC_5	0.957	0.11535	72.15	460.4	33.812	0.2273	0.115	0.1	0
C_5	1.406	0.02264	72.15	469.7	33.688	0.251	0.115	0.11	0
C_6	2.097	0.11333	86.18	507.4	30.123	0.2957	0.115	0.11	0
$C_{7+}(1)$	4.902	0.12047	98.55	563.2	31.776	0.2753	0.115	0.11	0.02
$C_{7+}(2)$	9.274	0.0914	135.84	638.3	26.19	0.3761	0.115	0.11	0.028
$C_{7+}(3)$	9.88	0.0001	206.65	736.5	19.637	0.5552	0.115	0.11	0.04
$C_{7+}(4)$	7.362	0	319.83	837	14.519	0.8021	0.115	0.11	0.052
$C_{7+}(5)$	3.804	0	500	936.9	11.066	1.108	0.115	0.11	0.064



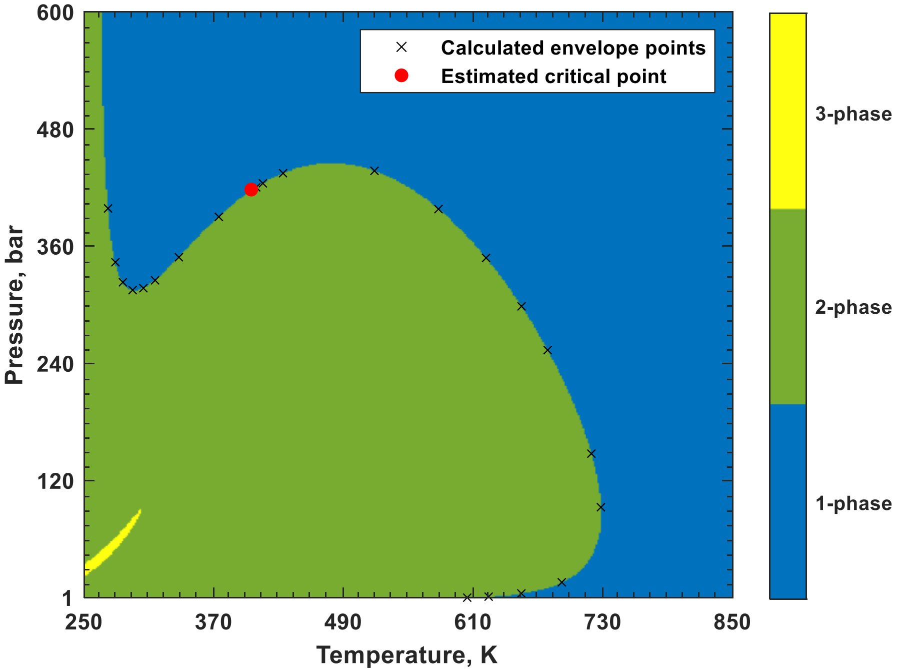
(a)


Calculated envelope points
3-phase
2-phase
1-phase
Pressure, bar
600
480
360
240
120
1
150
190
230
270
310
350
Temperature, K

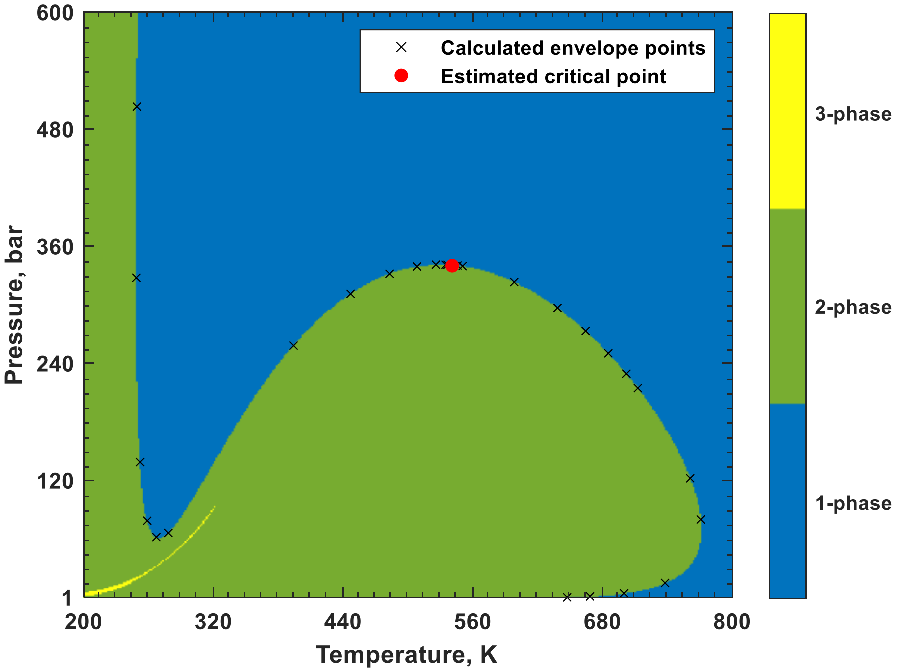
(b)


Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase
Pressure, bar
600
480
360
240
120
0.1
250
370
490
610
730
850
Temperature, K

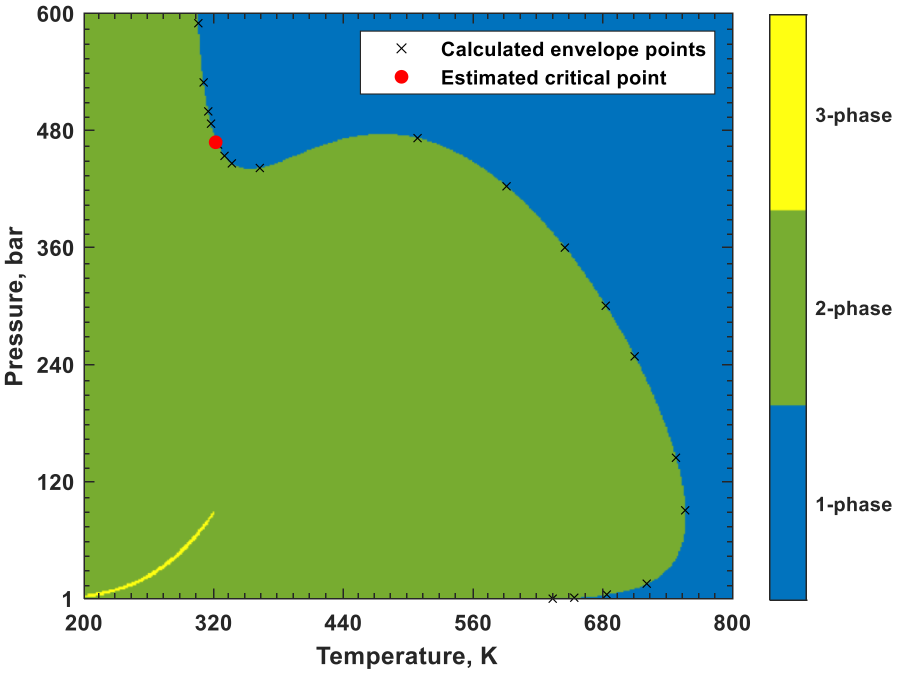
(c)


600
480
360
240
120
1
250
370
490
610
730
850
Pressure, bar
Temperature, K
Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase

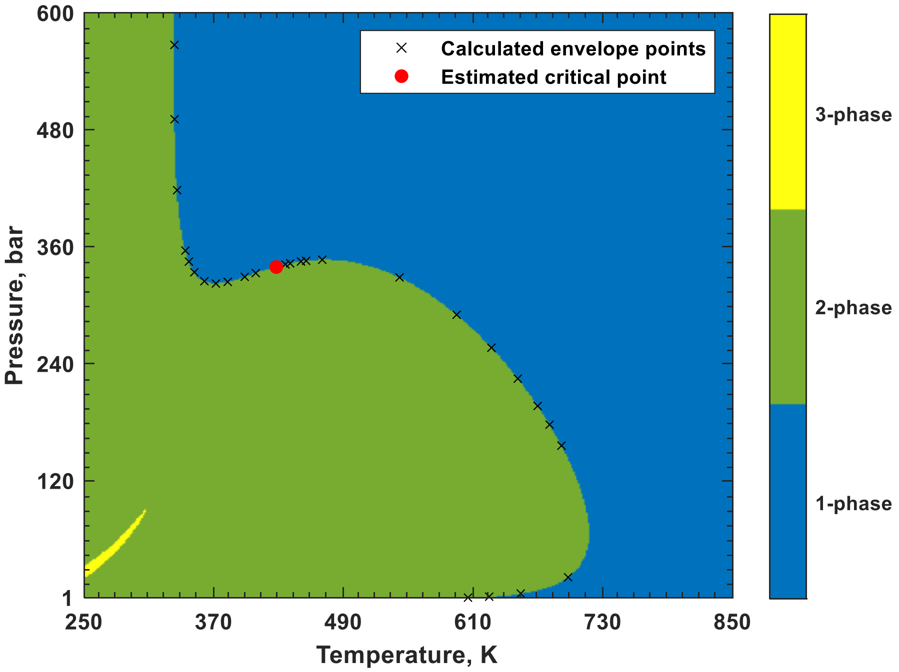
(d)


600
480
360
240
120
1
200
320
440
560
680
800
Pressure, bar
Temperature, K
Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase

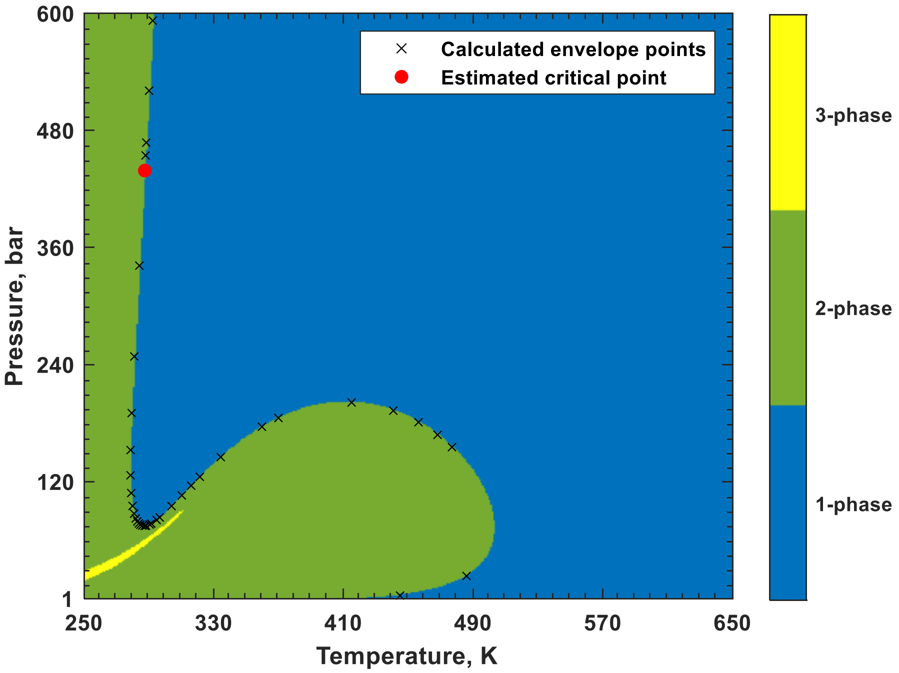
(e)


600
480
360
240
120
1
200
320
440
560
680
800
Pressure, bar
Temperature, K
Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase

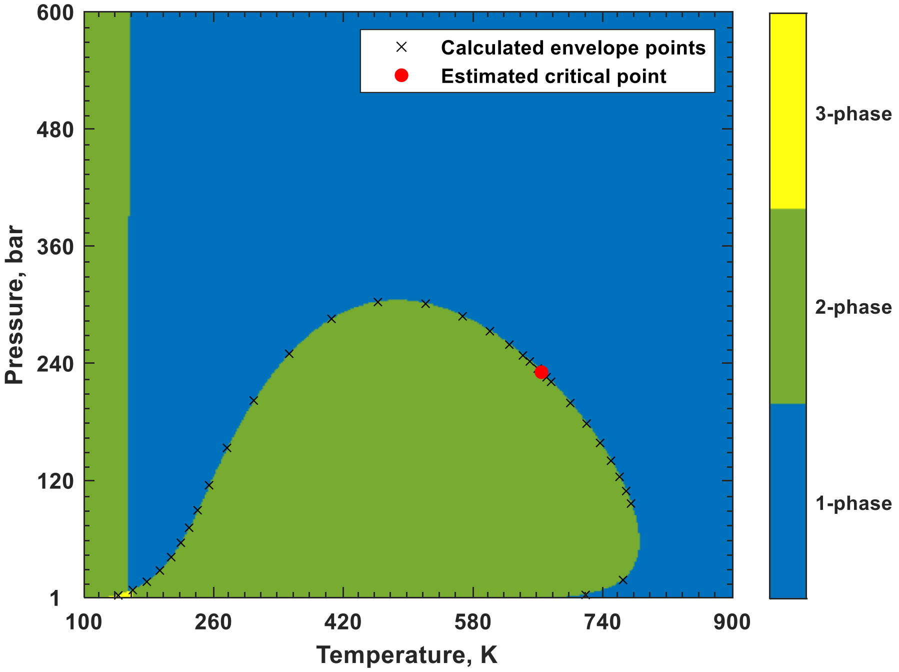
(f)


600
480
360
240
120
1
250
370
490
610
730
850
Pressure, bar
Temperature, K
Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase

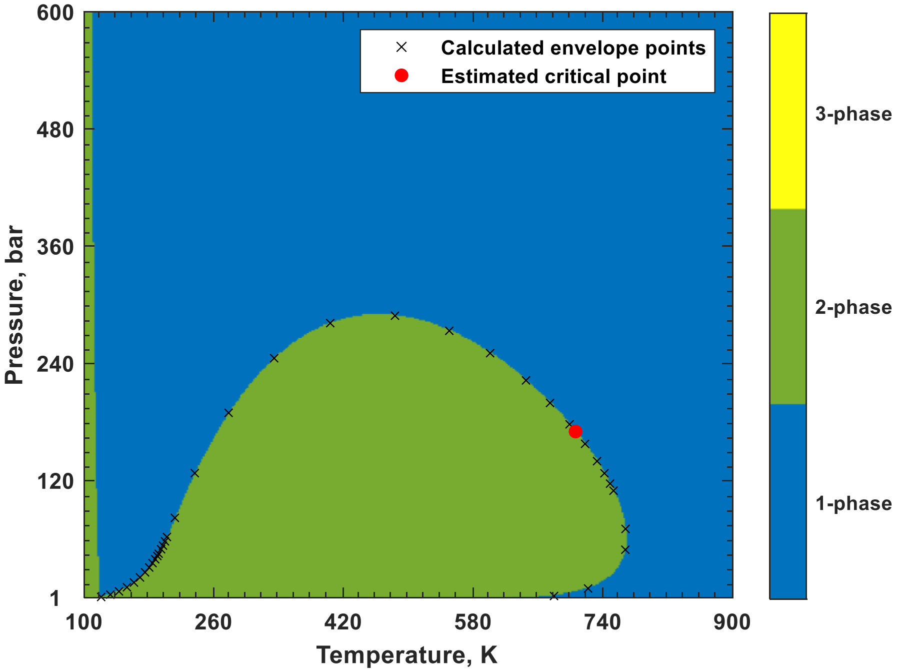
(g)


Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase
Pressure, bar
600
480
360
240
120
1
250
330
410
490
570
650
Temperature, K

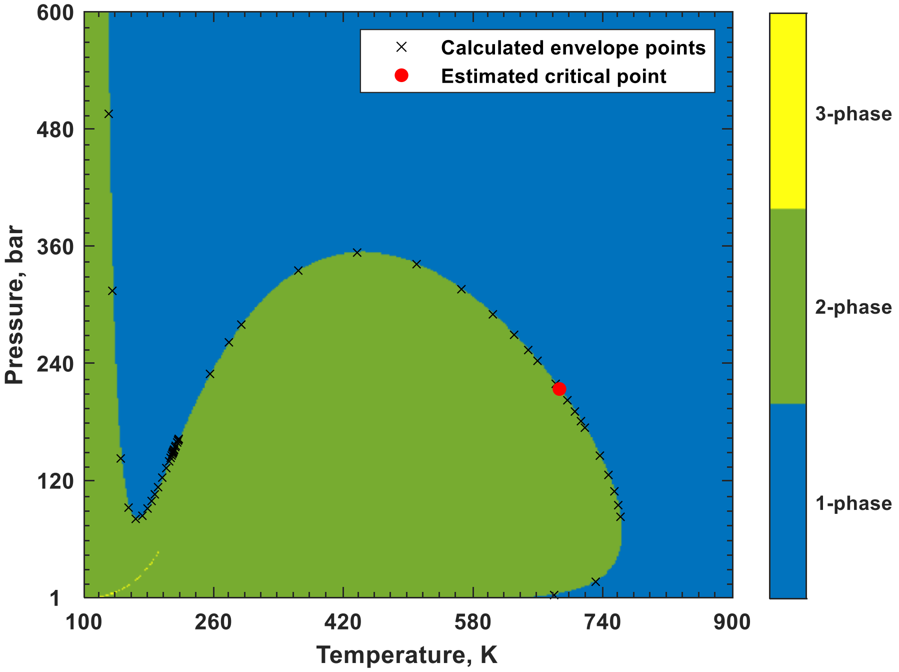
(h)


Calculated envelope points
Estimated critical point
3-phase
2-phase
1-phase
Pressure, bar
600
480
360
240
120
1
100
260
420
580
740
900
Temperature, K

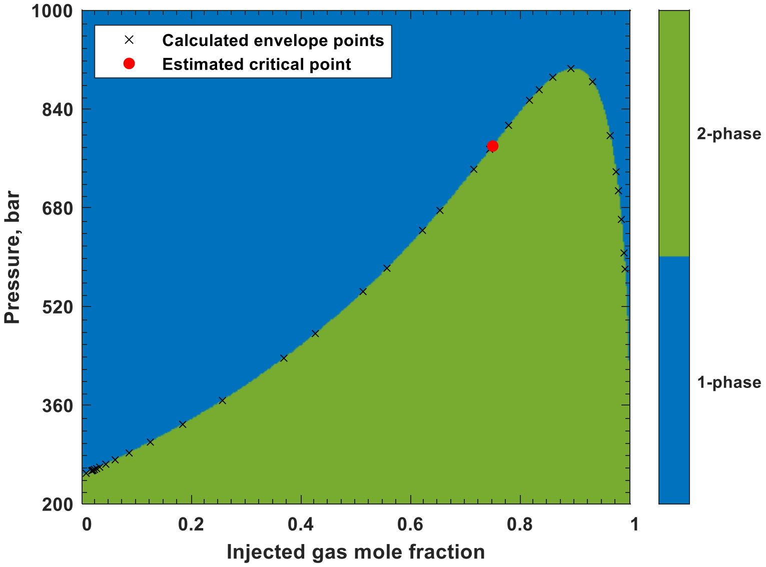
(i)


Calculated envelope points
Estimated critical point
Pressure, bar
600
480
360
240
120
1
100
260
420
580
740
900
Temperature, K
3-phase
2-phase
1-phase

(j)


Calculated envelope points
Estimated critical point
Pressure, bar
600
480
360
240
120
1
100
260
420
580
740
900
Temperature, K
3-phase
2-phase
1-phase

(k)



(l)

Fig. 3-A5. Digital PT and Px phase diagrams, together with the calculated EPs, generated for the following cases: (a) N2C10; (b) M5; (c) OilG0.7; (d) NWE0.7; (e) JEMA0.6; (f) JEMA0.75; (g) MRO0.7; (h) MY10-0.8; (i) Zick0.2; (j) Høier0.05; (k) Høier0.2; (l) Høier-Px. Each digital phase diagram is generated by a total of $400 \times 400 = 160{,}000$ runs of multiphase equilibrium calculations. The EPs points are generated with the multiplier set listed in Table 3-1.

CHAPTER 4 CONSTRUCTING THREE-PHASE ENVELOPES USING A TRUST-REGION-BASED ALGORITHM

A version of this chapter is submitted to *Chemical Engineering Science* for possible publication on Feb 1, 2023.

Abstract

We extend the trust-region-based algorithm, which is previously developed to construct two-phase envelopes (Xu and Li, 2023), to construct multiphase envelopes involving two-phase and three-phase branches. We test the performance of the trust-region-based algorithm by constructing multiphase envelopes for a total of 15 fluid mixtures that include hydrocarbons-CO_2, hydrocarbons-asphaltene, and hydrocarbons-water mixtures. Comparison against the conventional Newton-based algorithm indicates that the trust-region-based algorithm leads to a much higher computational efficiency and an enhanced robustness.

4.1 Introduction

The objective of the petroleum industry is to extract, separate, transport, and store different types of hydrocarbon mixtures. With the presence of CO_2, water, or asphaltenes, the hydrocarbon mixtures could exhibit three-phase equilibria, such as vapor-liquid-liquid (VLL), vapor-liquid-water (VLW), and vapor-liquid-asphaltene (VLA) equilibria (Li, 2021). Phase diagrams, under pressure-temperature (PT) or pressure-composition (Px) specifications, play an important role in understanding the in-situ phase behavior of hydrocarbon fluid mixtures. An efficient approach to plotting phase diagrams is to directly construct the phase envelopes (Michelsen, 1980).

Michelsen's algorithm (Michelsen, 1980) is an iterative approach to generating a string of two-phase PT envelope points (EPs), at which one of the phases becomes incipient. One starts with finding the first EP, i.e., a high-temperature and low-pressure dew point. A subsequent EP will be initialized based on the first EP, and the Newton-Raphson (NR) method is followed to reach tight convergence. The algorithm continues until one of the specified tolerances, such as the maximum allowable pressure/temperature, is crossed. Michelsen's method has been successfully extended to handle different scenarios, such as the Px envelope construction (Li and Nghiem, 1982), the three-phase PT envelope constructions (Michelsen, 1986), the water-inclusive three-phase envelope construction (Lindeloff and Michelsen, 2003), and the three-phase envelope construction involving asphaltene precipitations (Agger and Sørensen, 2018; Cismondi, 2018).

The construction of multiphase envelopes involving two- and three-phase envelopes requires not only the formulations configured to solve two-phase EPs, three-phase EPs, and three-phase points (i.e., the intersection point of a two-phase envelope and a three-phase region), but also an integrated strategy to delineate all the branches of the complete multiphase envelopes (Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018; Cismondi, 2018). The conventional strategy, developed by Lindeloff and Michelsen (2003) and followed by Agger and Sørensen (2018), applies the stability test (Michelsen, 1982) to detect three-phase points when constructing two-phase envelopes. The detected three-phase point will then serve as the starting point to construct the envelope of the corresponding three-phase region. As such, all the branches of the two-phase and three-phase envelopes can be traversed. However, appending a stability test to each two-phase EP calculation will further increase the total computational load. To address this issue, Cismondi (2018) proposed a new strategy without stability tests. Cismondi (2018) suggests constructing two different two-phase envelope branches, i.e., the vapor-liquid (VL) and liquid-liquid (LL) branches, and determining if the intersection points are three-phase points or critical points. On the one hand, such a strategy will save the computational cost consumed by the stability tests. On the other hand, however, it requires additional knowledge to locate the first EP on the LL branch and delete the unphysical parts of the two-phase envelopes. Considering its simplicity and robustness in constructing different types of multiphase envelopes, the strategy developed by Lindeloff and Michelsen (2003) is implemented in this study.

The NR method is the most used solution method for phase envelope constructions (Michelsen, 1980; Michelsen, 1986; Lindeloff and Michelsen, 2003; Agger and Sørensen, 2018; Cismondi, 2018) since the development of the general construction algorithm (Michelsen, 1980). However, the NR method has been followed without any change for decades. To our knowledge, no attempt has been made to develop a more cost-effective solution method for phase envelope constructions than the NR method. Moreover, we have observed that the NR-based algorithm may have difficulty in crossing the critical point during the two-phase envelope constructions (Xu and Li, 2023). Inspired by the implementation of the robust and efficient trust-region (TR) method in the stability tests and flash calculations (Petitfrere and Nichita, 2014; Pan et al., 2019), Xu and Li (2023) introduced a new TR-based algorithm into two-phase envelope constructions. As shown in Xu and Li (2023), the TR-based algorithm is proven to be generally more robust and efficient than the conventional NR algorithm. This work extends such a TR-based algorithm to construct multiphase PT and Px envelopes (i.e., envelopes containing both two-phase and three-phase branches).

In “Methodology”, we first introduce the formulations of the two-phase and three-phase EPs, the formulations of the TR method, and the strategy for multiphase envelope constructions. Case studies to show the performance of the new TR-based algorithm are illustrated in "Results and discussion". At last, we draw the major conclusions in "Conclusions".

4.2 Methodology

4.2.1 Formulations of two-phase and three-phase EPs under PT and Px conditions

To construct a two-phase envelope, the positions of EPs should be determined. Specifically, the position of each EP in the PT/Px diagrams can be determined by solving the $(nc + 2)$-dimension nonlinear equation set given by (Michelsen, 1980),

$$F_i(u_{2p}) = ln(K_i) - ln\varphi_{x,i} + ln\varphi_{y,i}, i = 1,2,\cdots,nc \tag{4-1}$$

$$F_{nc+1}(u_{2p}) = \sum_{i=1}^{nc}(x_i - y_i) \tag{4-2}$$

$$F_{nc+2}(u_{2p}) = S_{spec} - S \tag{4-3}$$

In our implementation, fugacity coefficients are calculated by Peng-Robinson equation of state (1978). In **Eq. 4-2**, x_i and y_i are, respectively, defined as,

$$x_i = z_i \tag{4-4}$$

$$y_i = z_i K_i \tag{4-5}$$

Eq. 4-3 is the specification equation as defined by Michelsen (1980). In Eq. 4-3, variable S is the specification variable that is chosen to be one variable in the independent variable vector u_{2p}:

$$S \in u_{2p} = [lnK_1, \cdots, lnK_{nc}, lnT, lnP] \tag{4-6}$$

For a two-phase Px EP, the nonlinear equation set keeps unaltered, but the independent variables are changed to,

$$u'_{2p} = [lnK_1, \cdots, lnK_{nc}, r, lnP] \tag{4-7}$$

where r ($0 \leq r \leq 1$) is the injection gas fraction that satisfies,

$$z_i(r) = (1-r)z_{o,i} + rz_{g,i} \tag{4-8}$$

where z_o represents the oil composition and z_g represents the gas composition. The differentiation of **Eq. 4-8** leads to,

$$\frac{\partial z_i}{\partial r} = z_{g,i} - z_{o,i} \tag{4-9}$$

Each three-phase PT EP satisfies a total of $2nc + 3$ equations, including $2nc$ fugacity equalities, two material balance equations, and one specification equation (Agger and Sørensen, 2018):

$$F_i(u_{3p}) = ln\big(K_{i,1}\big) - ln\varphi_{w,i} + ln\varphi_{x,i}, i = 1,2,\cdots,nc \tag{4-10}$$

$$F_{nc+i}(u_{3p}) = ln\big(K_{i,2}\big) - ln\varphi_{w,i} + ln\varphi_{y,i}, i = 1,2,\cdots,nc \tag{4-11}$$

$$F_{2nc+1}\big(u_{3p}\big) = \sum_{i=1}^{nc}(x_i - y_i) \tag{4-12}$$

$$F_{2nc+2}\big(u_{3p}\big) = \sum_{i=1}^{nc} w_i - 1 \tag{4-13}$$

$$F_{2nc+3}(u_{3p}) = S_{spec} - S \tag{4-14}$$

where w represents the incipient phase, and,

$$K_{i,1} = \frac{x_i}{w_i} \tag{4-15}$$

$$K_{i,2} = \frac{y_i}{w_i} \tag{4-16}$$

$$w_i = \frac{z_i}{\beta K_{i,1} + (1-\beta)K_{i,2}} \tag{4-17}$$

$$u_{3p} = [lnK_{1,1},\cdots,lnK_{nc,1},K_{1,2},\cdots,lnK_{nc,2},\beta,lnT,lnP] \tag{4-18}$$

For the calculation of one three-phase Px EP, similarly, we replace lnT in the independent variable vector u_{3p} by r.

To sum up, for a general independent variable vector u, we can write the nonlinear equation set to be solved as (Michelsen, 1980),

$$F(u,S) = 0,(S \in u) \tag{4-19}$$

Assuming that the solution of **Eq. 4-19** is u^*, the vector $\frac{\partial(u)}{\partial S}|_{u^*}$ is the so-called sensitivity vector (Michelsen, 1980) which reveals the gradient information of u with respect to the specification variable S. We define $d = \frac{\partial(u)}{\partial S}|_{u^*}$ as the sensitivity vector for the sake of simplicity.

4.2.2 Formulation of three-phase points

A three-phase point is a special point along a two-phase envelope at which two different incipient phases appear. Graphically, it is the intersection point of a two-phase envelope and a three-phase region in a phase diagram. There are $2nc + 2$ equations to be satisfied for a three-phase point, which are given by (Lindeloff and Michelsen, 2003),

$$F_i\ (u_{3pp}) = ln(w_i) + ln\varphi_{w,i} - ln(z_i) - ln\varphi_{z,i}, i = 1,2,\cdots,nc \tag{4-20}$$

$$F_{nc+i}(u_{3pp}) = ln(x_i) + ln\varphi_{x,i} - ln(z_i) - ln\varphi_{z,i}, i = 1,2,\cdots,nc \tag{4-21}$$

$$F_{2nc+1}(u_{3pp}) = \sum_{i=1}^{nc} w_i - 1 \tag{4-22}$$

$$F_{2nc+2}(u_{3pp}) = \sum_{i=1}^{nc} x_i - 1 \tag{4-23}$$

where,

$$u_{3pp} = [ln(w_1), \cdots, ln(w_{nc}), ln(x_1), \cdots, ln(x_{nc}), lnT, lnP] \tag{4-24}$$

To solve for a three-phase point in the Px diagram, we replace lnT by r in u_{3pp}.

The initial guesses of u_{3pp}, including the compositions of the incipient phases w and x, are required for the three-phase point calculations. To determine an initial u_{3pp}, a stability test is performed on each converged two-phase EP. When instability is detected, the recorded trial phases and the PT/Px condition will be used to generate the initial guesses of u_{3pp} (Lindeloff and Michelsen, 2003). In the stability test, we use the following six initial estimates of K-values (Michelsen, 1982; Li and Firoozabadi, 2012; Pan et al., 2019),

$$K_i^{Wilson}, 1/K_i^{Wilson}, \sqrt[3]{K_i^{Wilson}}, 1/\sqrt[3]{K_i^{Wilson}}, K_i^1, K_i^{nc} \tag{4-25}$$

where,

$$K_i^{Wilson} = P_{c,i}\frac{exp\left[5.37(1+\omega_i)\left(1-\frac{T_{c,i}}{T}\right)\right]}{P},\ i = 1,2,\cdots,nc \tag{4-26}$$

$$K_1^1 = \frac{0.9}{z_1}, K_i^1 = \frac{0.1}{z_i(nc-1)},\ i = 2,\cdots,nc \tag{4-27}$$

$$K_{nc}^{nc} = \frac{0.9}{z_{nc}}, K_i^{nc} = \frac{0.1}{z_i(nc-1)},\ i = 1,2,\cdots,nc-1 \tag{4-28}$$

4.2.3 Trust-region method for EP determinations

To solve the general nonlinear system defined by Eq. 4-19 using the TR method, we define a least-square function as follows (Xu and Li, 2023),

$$M(u) = \sum_{i=1}^{nc+2} 0.5\|F_i\|^2 \tag{4-29}$$

The gradient g and the Hessian matrix H can be expressed as (Sun and Yuan, 2006),

$$g = J^T F \tag{4-30}$$

$$H \approx J^T J \tag{4-31}$$

where J is the Jacobian matrix of the non-linear equation set F. In the TR method for the least-squares problem, the model function $m^n(s^n)$ is given by (Nocedal and Wright, 2006),

$$m^n(s^n) = 0.5\|F^n + J^n s^n\|^2 = M^n + s^n(J^n)^T F^n + 0.5 s^n (J^n)^T J^n s^n \tag{4-32}$$

where n is the iteration index. The TR ratio ρ is determined by (Nocedal and Wright, 2006),

$$\rho^n = \frac{M(u^n) - M(u^n + s^n)}{m^n(0) - m^n(s^n)} \tag{4-33}$$

Based on Nocedal and Wright (2006) and Sun and Yuan (2006), the TR method aims to locate the minimum of a specified objective function (M) by means of a quadratic model (m). In a neighborhood called the trust region, such quadratic model serves as the approximation of the objective function. By locating the minimum of the quadratic model, a TR step (s) can be obtained. Next, the quality of the TR step is evaluated based on the TR ratio (ρ). If ρ is smaller than zero, we reject the step as the objective function cannot be reduced; otherwise, we accept the step and update the variables (by setting $u = u + s$). The size of the trust region is also adjusted based on ρ. Such procedure iterates until the termination criterion is satisfied.

In our implementation, we closely follow the state-of-the-art TR method for flash calculations and stability tests (Petitfrere and Nichita, 2014; Pan et al., 2019) to develop the TR-based algorithm for EP determinations. More TR algorithm details can be found in Xu and Li (2023).

4.2.4 Construction strategy for multiphase envelopes with two-phase and three-phase branches

Depending on whether three-phase points exist, there are two types of generic multiphase envelopes. **Fig. 4-1** illustrates a generic PT phase envelope of type one. As shown in Fig. 4-1, such an envelope can be divided into six branches by two different three-phase points. Type one represents a variety of PT phase envelopes made up of some or all of the six branches. For example, the phase envelope of the fluid mixture MY10-0.2 (Xu and Li, 2023) and the generic water-inclusive envelope (Lindeloff and Michelsen, 2003) belong to this type. Note that we can also categorize the Px envelopes containing one three-phase point (Pan et al., 2019) as type-one phase envelope. Lindeloff and Michelsen (2003) developed a general strategy that could be applied to construct multiphase envelopes of type one.

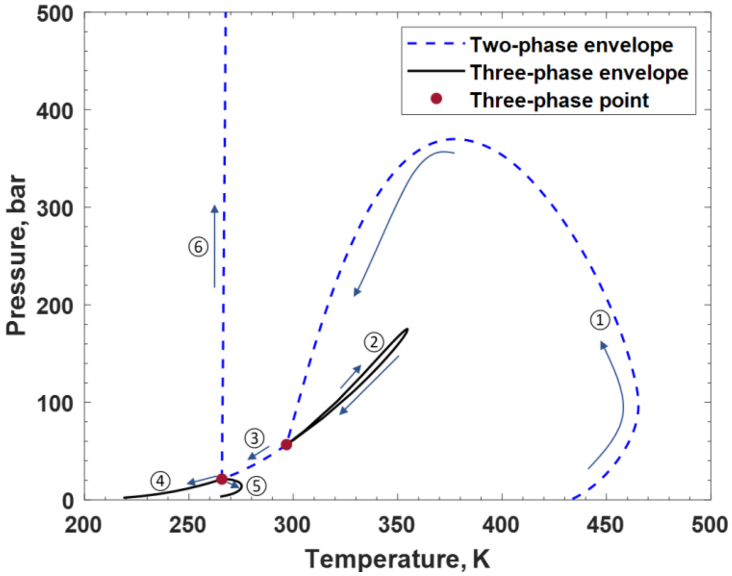


Fig. 4-1. A generic PT phase envelope containing two three-phase points as marked by filled red circles. The arrows together with the numbers indicate the construction sequence of EPs on two-phase-envelope and three-phase-envelope branches.

Fig. 4-2 illustrates a representative phase envelope that belongs to the second type. Its two-phase part will be uplifted to the high-pressure side without meeting any three-phase point. Example type-two envelopes include several PT envelopes illustrated in Agger and Sørensen (2018), Lu et al. (2021), and Xu and Li (2023). Agger and Sørensen (2018) followed the strategy developed by Lindeloff and Michelsen (2003) and extended it to construct multiphase envelopes of type two. As suggested by Agger and Sørensen (2018), the construction of the three-phase envelope begins with a low-temperature three-phase bubble point. Then the remaining three-phase envelope can be constructed just like a two-phase one. The stepwise procedure of the integral strategy covering the constructions of both type-one and type-two phase diagrams can be found in Lindeloff and Michelsen (2003) and Agger and Sørensen (2018).

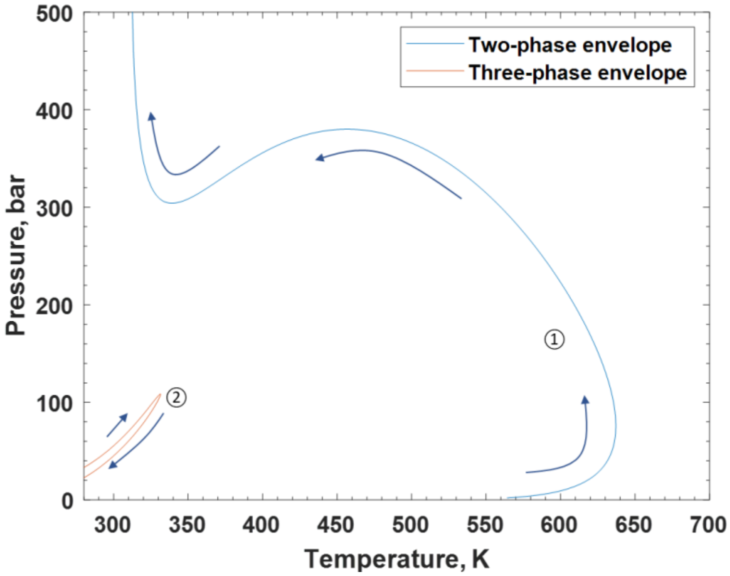


Fig. 4-2. A generic PT phase envelope without three-phase points. The arrows together with the numbers indicate the construction sequence of EPs on two-phase-envelope and three-phase-envelope branches.

4.3 Results and discussion

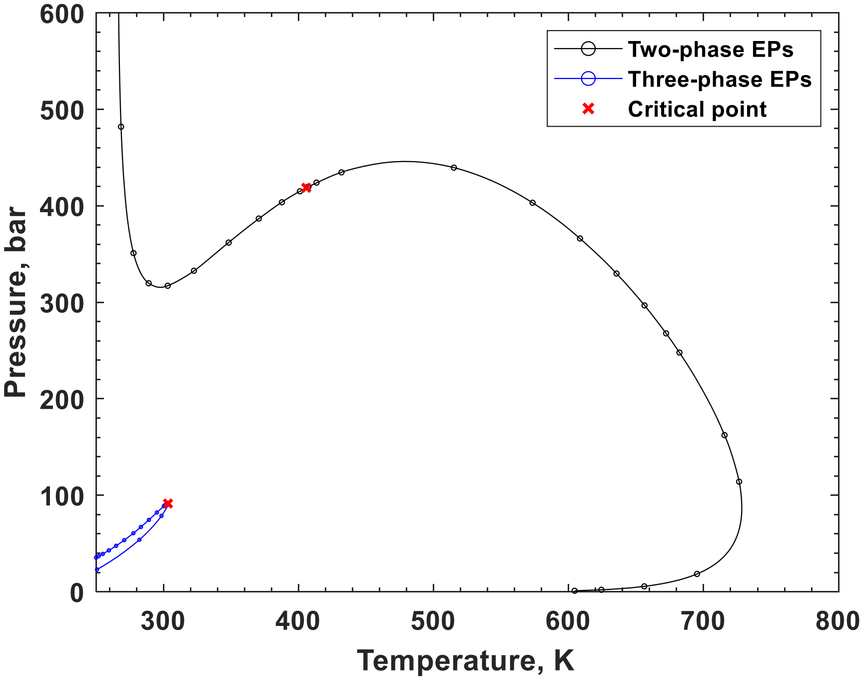
In this section, we apply the TR-based algorithm to construct a total of 15 multiphase envelopes for different types of reservoir mixtures, including hydrocarbons-CO_2 mixtures, hydrocarbons-asphaltene mixtures, and hydrocarbons-water mixtures. Specifically, the oil mixtures include the Oil G (Khan et al., 1992), the North Ward Este (NWE) oil (Khan et al., 1992), the JEMA oil (Khan et al., 1992), the Oil b (Shelton and Yarborough, 1977; Haugen et al., 2011), the near-critical NC-P oil (Pedersen and Christensen, 2006), the MY10 oil with the presence of CO_2 and water (Metcalfe and Yarborough, 1979; Johns and Orr, 1996; Imai et al., 2019), the Zick-2 oil (Zick, 1986; Jessen et al., 1998), the Glasø-A oil (Glasø, 1985; Zhao and Fang, 2020), the asphaltene sample 1 (AS-1) (Jamaluddin et al., 2002; Chen et al., 2022), and the asphaltene sample 2 (AS-2) (Buenrostro - Gonzalez et al., 2004; Chen et al., 2022). **Table 4-1** summarizes the basic information of the fluid mixtures for which phase diagrams are constructed in this study. Appendix 4-A.1 lists the detailed fluid properties used in the 15 case studies. We adopt Peng–Robinson equation of state (Robinson and Peng, 1978) in all the calculations. The termination criterion of an EP calculation is set to $err = \|F\| < tol = 10^{-9}$. The phase envelope construction code is run on MATLAB® (The MathWorks, Inc., Natick, Massachusetts, USA) using a laptop personal computer with a 64-bit operating system.

Table 4-1. Basic information of the 15 fluid mixtures considered in this study.

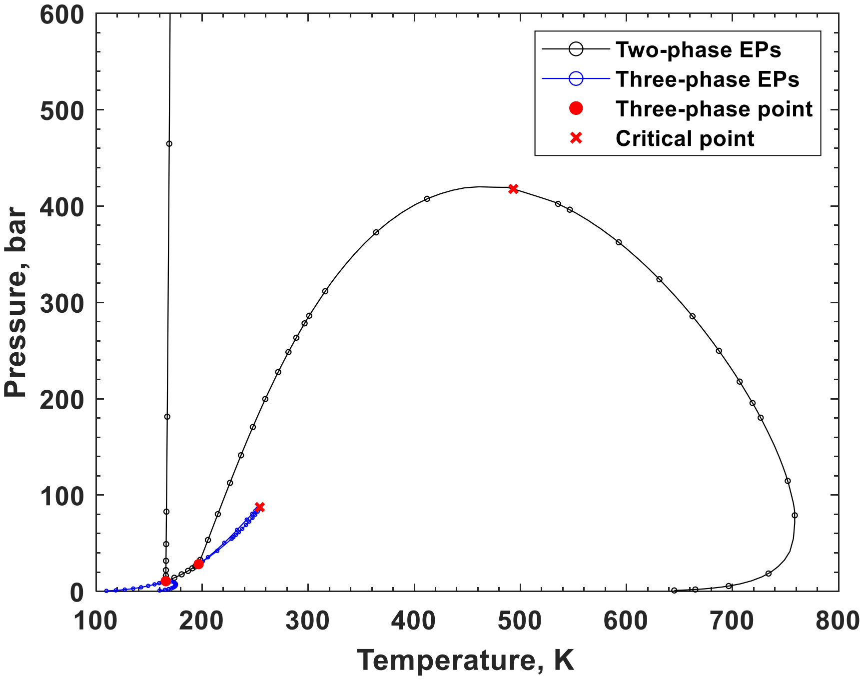
Case index	Number of components	Oil type	Gas type	Gas mole fraction, mol%	Type of phase diagram	References
OilG0.4	7	OilG	CO_2	40	PT	Khan et al., 1992
OilG0.7				70	PT	
OilG-Px				-	Px	
NWE0.7	7	NWE Oil	95 mol% CO_2 + 5 mol% CH_4	70	PT	Khan et al., 1992
JEMA0.75	7	JEMA Oil	CO_2	75	PT	Khan et al., 1992
JEMA-Px	7			-	Px	
MY10-0.2	11	MY10	CO_2	20	PT	Metcalfe and Yarborough, 1979; John and Orr, 1996
Glasø0.95	11	Glasø-A	Glasø-A	95	PT	Glasø, 1985; Zhao and Fang, 2020
MY10-water	12	MY10-water	-	-	PT	Metcalfe and Yarborough, 1979; Johns and Orr, 1996; Imai et al., 2019
Zick0.6	12	Zick-2	Zick-2	60	PT	Zick, 1986; Jessen et al., 1998
NC-P	16	NC-P	-	-	PT	Pedersen and Christensen, 2006
Oilb0.71	16	Oilb	CO_2	71	PT	Shelton and Yarborough, 1977; Haugen et al., 2011; Pan et al., 2019
Oilb-Px	16			-	Px	
AS-1	23	AS-1	-	-	PT	Jamaluddin et al., 2002; Chen et al., 2022
AS-2	22	AS-2	-	-	PT	Buenrostro-Gonzalez et al., 2004; Chen et al., 2022

Fig. 4-3 illustrates four representative multiphase PT/Px envelopes generated by the TR-based algorithm. As shown in Fig. 4-3, the phase envelopes are arranged in sequence to show a PT envelope without three-phase points (NWE0.7), a PT envelope with

two three-phase points (Zick0.6), a special PT envelope where a three-phase point appears after the two-phase envelope is curved to the high-temperature side (Oilb0.71), and a Px envelope with one three-phase point (JEMA-Px). **Table 4-2** shows an example of equilibrium phase compositions at the three-phase point (196.7 K and 28.3 bar) in Fig. 4-3b (i.e., the three-phase point with a higher pressure in Fig. 4-3b). Appendix 4-A.2 presents the remaining 11 phase envelopes constructed by the TR-based algorithm. Note that we do not encounter a single failure when constructing these phase envelopes using the TR-based algorithm. In the following discussion, we will first compare the computational efficiency of the new TR-based phase envelope construction algorithm against that of the conventional NR-based algorithm. Then we validate the correctness of the calculated EPs.

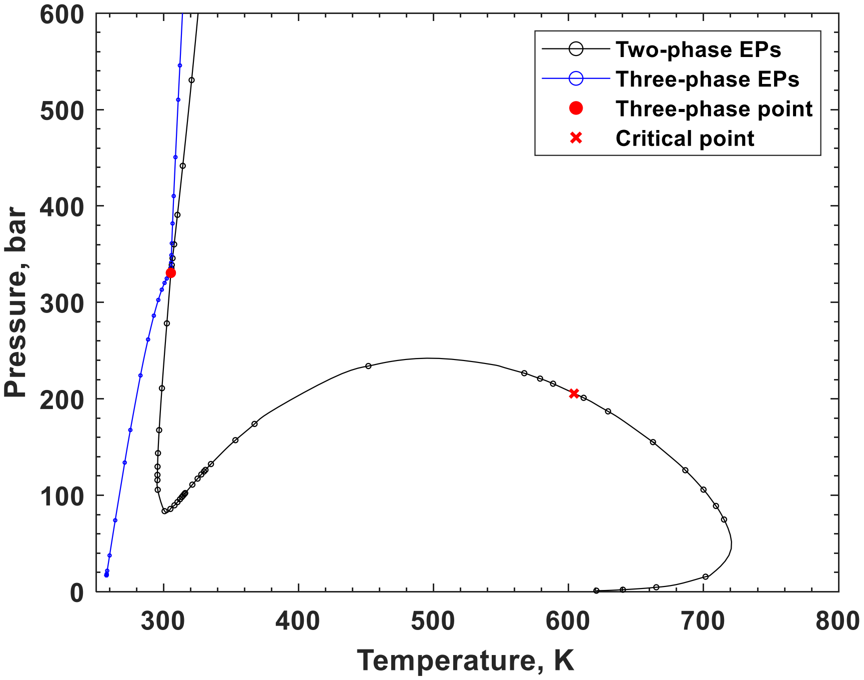


(a)



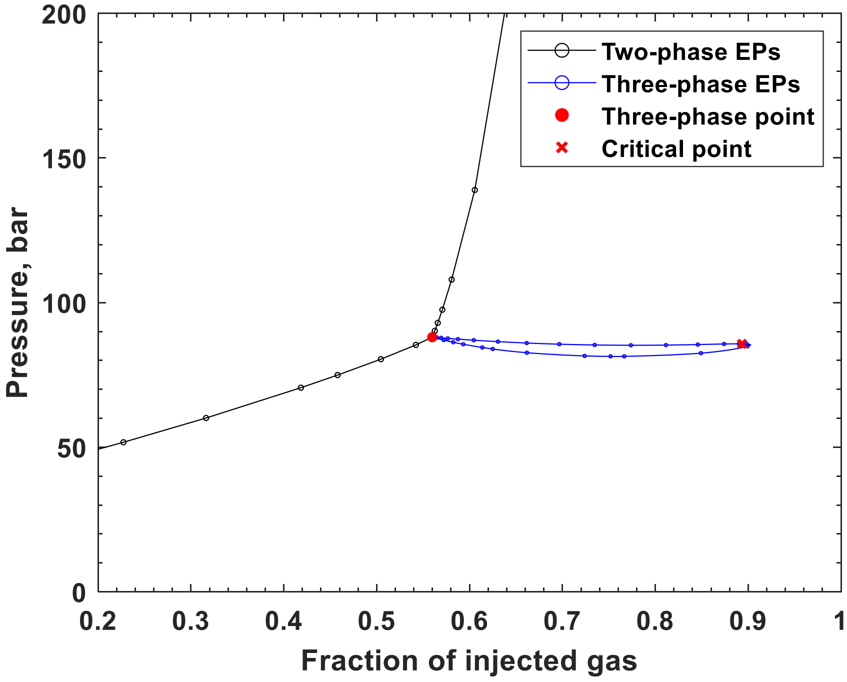
Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
0
100
200
300
400
500
600
Temperature, K
100
200
300
400
500
600
700
800

(b)



Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
0
100
200
300
400
500
600
Temperature, K
300
400
500
600
700
800

(c)



(d)

Fig. 4-3. Phase envelopes yielded by the TR-based algorithm for four representative cases: (a) NEW0.7; (b) Zick0.6; (c) Oilb0.71; (d) JEMA-Px.

Table 4-2. Equilibrium phase compositions at the three-phase point (196.7 K and 28.3 bar) in the PT diagram calculated for case Zick0.6.

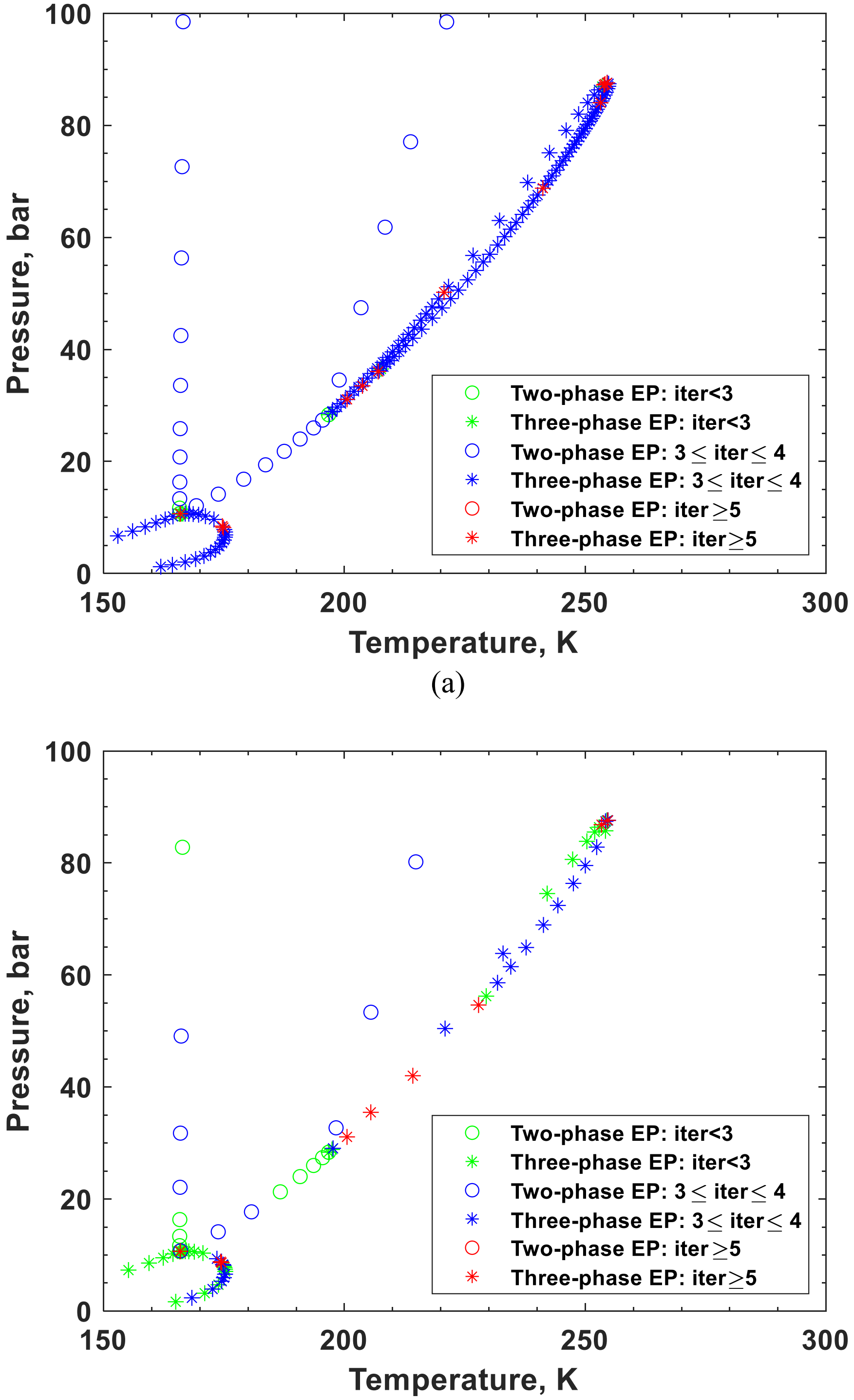
Vapor-like incipient phase	Liquid-like incipient phase	Feed
6.124E-02	1.452E-01	1.327E-01
9.166E-01	4.209E-01	3.811E-01
1.915E-02	1.365E-01	1.343E-01
2.935E-03	1.458E-01	1.467E-01
8.054E-05	2.609E-02	2.670E-02
3.412E-06	7.242E-03	7.481E-03
7.169E-07	8.566E-03	8.721E-03
9.382E-09	5.782E-02	7.165E-02
1.169E-13	2.740E-02	3.640E-02
1.010E-18	1.418E-02	2.420E-02
9.000E-25	7.008E-03	1.788E-02
1.664E-31	3.233E-03	1.208E-02

4.3.1 Computational efficiency and robustness

To make a fair comparison, we build two parallel codes: one envelope construction code embedding the TR method and a duplicate adopting the NR method, as done by Xu and Li (2023). We construct the 15 phase envelopes using both codes, and then record the EPs, the numbers of iterations, and the time consumptions yielded by the two codes.

Fig. 4-4a and **b** illustrate the EPs generated by the NR algorithm and the TR algorithm, respectively, for case Zick0.6. We can observe from Fig. 4-4 that the TR method tends to consume fewer iterations to yield one EP, resulting in a larger step size to march forward to the next EP. As such, the TR algorithm requires fewer EP calculations to trace the entire set of phase boundaries. This indicates that the TR-based algorithm is more cost-effective in constructing multiphase envelopes, especially three-phase envelopes.

Fig. 4-5 illustrates a part of the three-phase EPs yielded by the NR algorithm and the TR algorithm for AS-1. Like the phenomenon observed in Xu and Li (2023), however, we find the NR algorithm has difficulty constructing the correct three-phase envelope in this case. As shown in Fig. 4-5, the NR algorithm leads to smaller step sizes during the construction process. Because the step sizes are not large enough to generate equilibrium ratios that can "jump" across the critical point, the NR algorithm yields incorrect three-phase EPs in the near-critical region and fails to move across the critical point. Contrarily, the fast convergence behavior of the TR method leads to large step-size changes, making the TR algorithm easier to cross the critical point than the NR method. **Table 4-3** further illustrates the changes of equilibrium ratios at the EPs calculated before and after crossing the critical point. We can observe from Table 4-3 that $lnK_{i,1}$ calculated by the TR algorithm changes sign (i.e., "jumps" over the critical point), while no sign change is observed in $lnK_{i,1}$ estimated by the NR algorithm. This proves that the TR-based algorithm behaves more robustly in constructing three-phase envelopes, especially in the near-critical regions.



(a)

(b)

Fig. 4-4. Calculated EPs for Zick0.6: (a): EPs calculated by the NR method; (b) EPs calculated by the TR method. Green, blue, and red circles/stars represent EPs which require small (<3), medium (3 or 4), and large (≥5) iteration numbers to reach convergence, respectively.

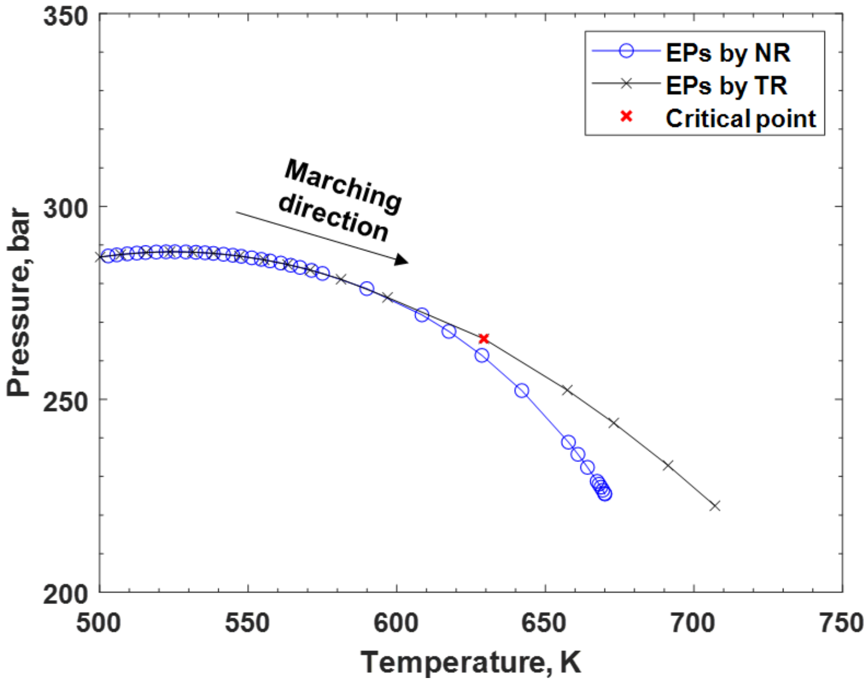


Fig. 4-5. EPs yielded by the NR-based algorithm and the TR-based algorithm for AS-1 in the three-phase region where the convergence difficulty may appear. The arrow shows the marching direction.

Table 4-3. Comparison of equilibrium ratios at the three-phase EPs calculated before and after crossing the critical point in the three-phase region shown in Fig. 4-5.

Reference three-phase EP before crossing the critical point		Incorrect three-phase EP after crossing the critical point calculated by the NR algorithm		Correct three-phase EP after crossing the critical point calculated by the TR algorithm	
$lnK_{i,1}$	$lnK_{i,2}$	$lnK_{i,1}$	$lnK_{i,2}$	$lnK_{i,1}$	$lnK_{i,2}$
2.861	9.712	0.476	7.470	-2.588	4.267
-0.241	-1.489	-0.047	-1.362	0.315	-1.007
-0.133	-0.921	-0.031	-0.981	0.220	-0.743
-0.165	-1.127	-0.035	-1.084	0.241	-0.817
-0.089	-0.858	-0.022	-0.875	0.171	-0.695
-0.039	-0.753	-0.013	-0.774	0.124	-0.652
-0.006	-0.741	-0.007	-0.733	0.092	-0.650
0.012	-0.646	-0.004	-0.670	0.077	-0.606
0.044	-0.630	0.002	-0.631	0.046	-0.606
0.053	-0.604	0.004	-0.616	0.040	-0.600
0.092	-0.609	0.011	-0.587	0.005	-0.615
0.131	-0.446	0.017	-0.472	-0.030	-0.544
0.155	-0.469	0.022	-0.468	-0.050	-0.568
0.184	-0.508	0.027	-0.472	-0.075	-0.605
0.238	-0.558	0.038	-0.470	-0.120	-0.664
0.315	-0.584	0.052	-0.446	-0.183	-0.725
0.410	-0.507	0.068	-0.357	-0.261	-0.739
0.524	-0.311	0.088	-0.189	-0.356	-0.696
0.650	-0.007	0.109	0.051	-0.463	-0.595

0.803	0.431	0.134	0.388	-0.597	-0.426
1.038	1.187	0.173	0.972	-0.807	-0.105
1.366	2.288	0.229	1.865	-1.114	0.410
1.569	2.017	0.284	2.224	-1.389	0.457
T	548.406	T	660.395	T	673.029
P	286.996	P	236.313	P	243.888

Table 4-4 lists the numbers of EPs and the numbers of iterations yielded by the NR algorithm and the TR algorithm for the 15 example calculations. **Table 4-5** shows the CPU time consumptions by the NR and TR algorithms for the 15 cases. **Table 4-6** further summarizes the reductions in the number of EPs, the number of iterations, and the computational time incurred by the TR algorithm with respect to the NR algorithm. We can observe from Table 4-4, Table **4-5**, and Table **4-6** that the TR method is effective in reducing the number of EPs, the number of iterations, and the computational load for both two-phase and three-phase envelope constructions. The gain is more obvious for the three-phase constructions. In cases such as OilG0.7, NWE0.7, and JEMA0.75, the percentage decreases in the total CPU time consumption yielded by the TR algorithm are more than 50%. In other cases, such as Zick0.6, NC-P, and AS-2, the absolute time reductions yielded by the TR algorithm are more than 10 seconds. In one of the challenging cases, i.e., AS-2, the phase envelope construction becomes extremely time-consuming as this asphaltic fluid contains 22 components. In this case, the TR-based algorithm saves 96 runs of EP calculations and 40.6 seconds. This demonstrates a significant improvement in the computational efficiency.

Table 4-4. Comparison of the number of EPs and number of iterations yielded by the TR-based and NR-based algorithms.

Case index	Total number of two-phase EPs		Total number of iterations consumed for two-phase EPs		Total number of three-phase EPs		Total number of iterations consumed for three-phase EPs	
	NR	TR	NR	TR	NR	TR	NR	TR

OilG0.4	60	37	223	113	41	28	130	75
OilG0.7	25	16	109	70	60	15	226	48
OilG-Px	27	22	136	86	74	46	288	179
NWE0.7	30	26	151	111	61	18	229	58
JEMA0.75	24	17	123	72	67	16	252	48
JEMA-Px	30	18	145	45	50	26	190	93
MY10-0.2	45	30	164	87	31	22	104	64
Glasø0.95	42	28	159	99	33	29	179	142
MY10-water	28	23	100	66	70	45	255	145
Zick0.6	71	46	272	150	112	64	437	209
NC-P	52	37	207	126	49	18	193	58
Oilb0.71	63	50	231	163	51	37	182	107
Oilb-Px	30	18	104	47	79	54	310	196
AS-1	21	12	72	35	-	68	-	273
AS-2	26	16	86	45	153	67	692	258

Table 4-5. Comparison of the CPU time consumptions between the TR-based and NR-based algorithms for the 15 cases.

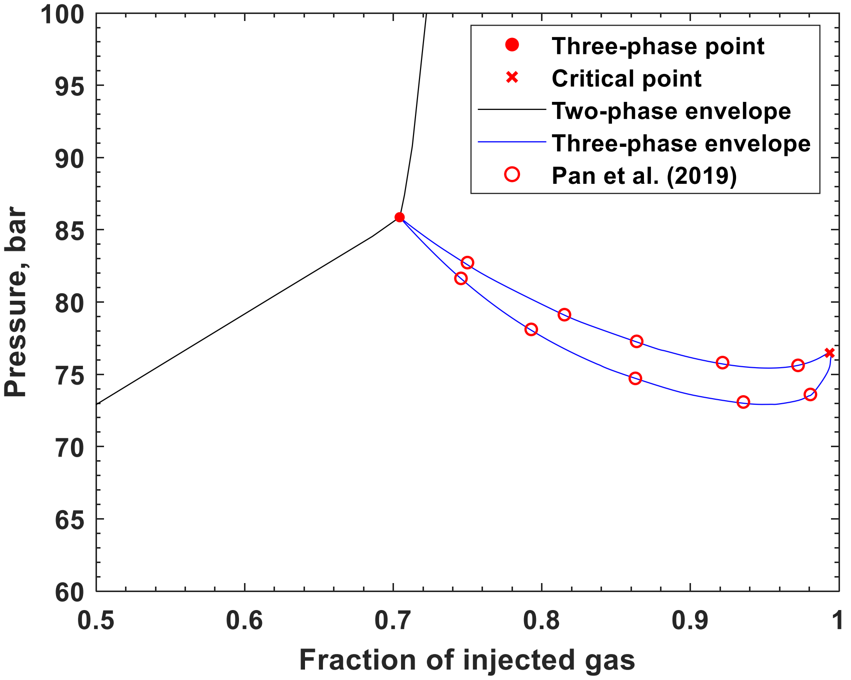
Case index	CPU time consumption (s) for two-phase EPs		CPU time consumption (s) for three-phase EPs		Overall CPU time consumption (s)	
	NR	TR	NR	TR	NR	TR
OilG0.4	5.07	3.43	3.82	2.27	8.89	5.70
OilG0.7	2.37	1.87	6.70	1.14	9.07	3.01
OilG-Px	2.52	2.15	7.06	4.16	9.57	6.31
NWE0.7	3.05	2.96	6.69	1.47	9.75	4.43
JEMA0.75	2.63	2.02	7.68	1.37	10.31	3.39
JEMA-Px	2.63	1.77	4.92	2.21	7.55	3.98
MY10-0.2	5.20	3.43	4.83	2.88	10.03	6.31
Glasø0.95	5.03	3.62	5.57	4.79	10.61	8.41
MY10-water	3.86	3.48	13.22	8.22	17.08	11.70
Zick0.6	9.46	6.68	20.44	10.60	29.90	17.29
NC-P	10.05	7.43	13.13	4.17	23.18	11.60
Oilb0.71	13.34	10.51	13.76	9.14	27.10	19.65
Oilb-Px	5.21	3.29	20.02	13.81	25.23	17.10
AS-1	7.90	4.56	-	30.25	-	34.81
AS-2	8.40	5.44	65.29	27.66	73.68	33.10

Table 4-6. Decreases in the number of EPs, the number of iterations, and the CPU time consumptions yielded by the TR method with respect to the NR method.

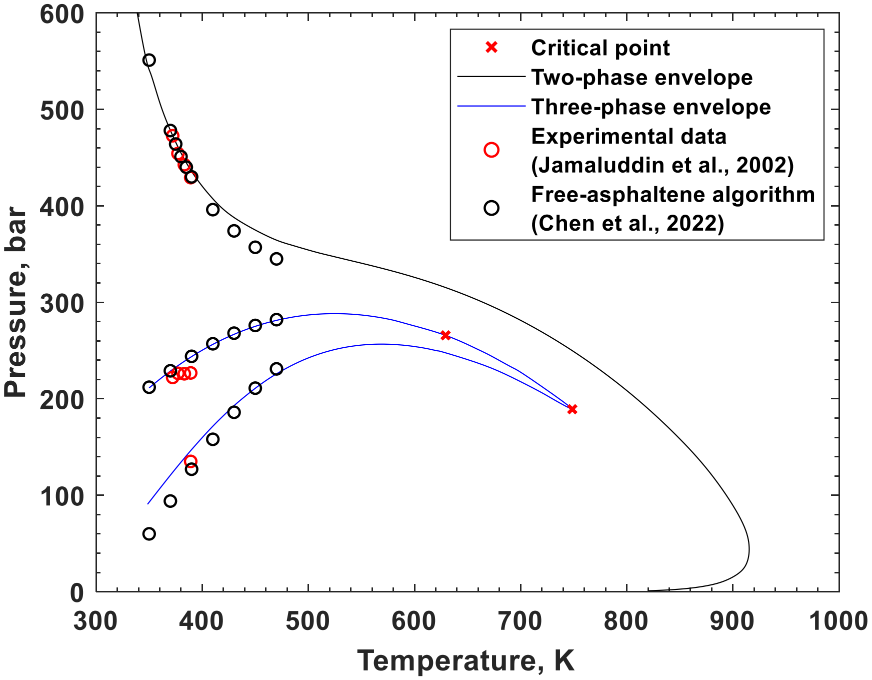
Case index	Decrease in the number of two-phase EPs	Decrease in the number of iterations consumed for two-phase EPs	Decrease in CPU time (s) for calculating two-phase EPs	Decrease in the number of three-phase EPs	Decrease in the number of iterations consumed for three-phase EPs	Decrease in CPU time (s) for calculating three-phase EPs	Decrease in CPU time (s) of the overall CPU time consumption	Percentage decrease of the overall CPU time consumption, %
OilG0.4	23	110	1.64	13	55	1.55	3.19	35.84
OilG0.7	9	39	0.49	45	178	5.56	6.05	66.77
OilG-Px	5	50	0.36	28	109	2.90	3.26	34.08
NWE0.7	4	40	0.09	43	171	5.22	5.31	54.50
JEMA0.75	7	51	0.62	51	204	6.30	6.92	67.13
JEMA-Px	12	100	0.86	24	97	2.71	3.57	47.27
MY10-0.2	15	77	1.77	9	40	1.95	3.72	37.07
Glasø0.95	14	60	1.42	4	37	0.78	2.20	20.74
MY10-water	5	34	0.38	25	110	5.00	5.38	31.52
Zick0.6	25	122	2.78	48	228	9.84	12.62	42.19
NC-P	15	81	2.62	31	135	8.95	11.58	49.95
Oilb0.71	13	68	2.83	14	75	4.62	7.45	27.49
Oilb-Px	12	57	1.92	25	114	6.21	8.13	32.22
AS-1	9	37	3.34	-	-	-	-	-
AS-2	10	41	2.96	86	434	37.62	40.58	55.07

4.3.2 Validations

To begin with, we validate the correctness of the EPs obtained from the TR-based algorithm by comparing the locations of the EPs with those of the reference EPs retrieved from the literature. **Fig. 4-6a-b** illustrate the validation results for two representative cases. We can observe from Fig. 4-6a that the constructed three-phase envelope by the TR-based algorithm is fully consistent with the results in Pan et al. (2019). As shown in Fig. 4-6b, the estimated phase envelopes by the TR-based algorithm are very close to the free-asphaltene flash calculation results reported by Chen et al. (2022), as well as the collected experimental data points. These good agreements illustrated in Fig. 4-6 verify that the phase envelopes generated by the TR-based algorithm reveal the correct phase behavior.



(a)

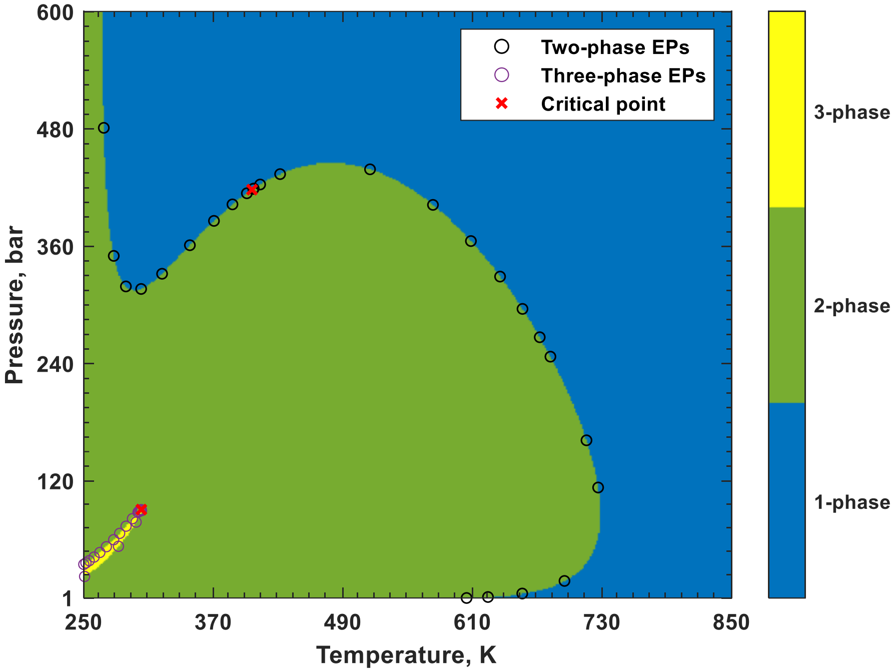


(b)

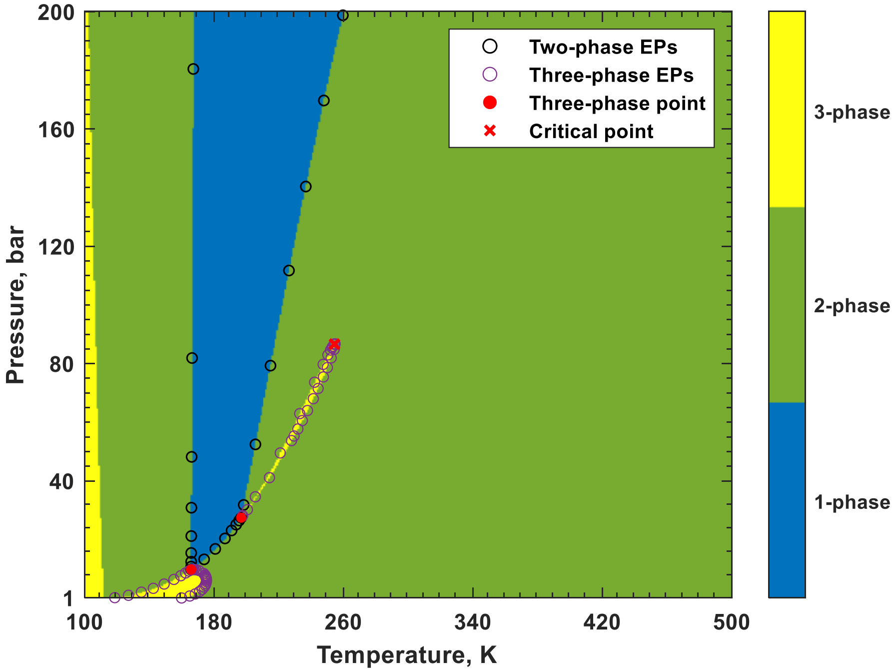
Fig. 4-6. Phase envelopes generated by the TR algorithm, together with the reference data points, for two representative cases: (a) Oilb-Px; (b) AS-1.

We further validate the correctness of the EPs by comparing the locations of the EPs with the phase boundaries that are demarcated in the discretized phase diagrams, as done by Agger and Sørensen (2018). We generate high-resolution discretized phase diagrams using the robust multiphase (up to three phases) equilibrium calculation framework applied in our previous publications (Lu et al., 2021; Xu and Li, 2023). Each discretized phase diagram involves a total of $400 \times 400 =$ 160,000 runs of multiphase equilibrium calculations. **Fig. 4-7a-d** illustrate the validation results for cases NWE0.7, Zick0.6, Oilb0.71, and JEMA-Px, respectively. We can observe from Fig. 4-7a, b, and d that the EPs yielded by the TR-based algorithm are fully consistent with the phase boundaries demarcated by the multiphase equilibrium calculations. Fig. 4-7c shows that a three-phase point, which is different from the ones illustrated for other CO_2-inclusive mixtures, is located on the near-perpendicular part of the two-phase envelope. Besides, we observe in Fig. 4-7c that the calculated three-phase EPs fail to envelope the true three-phase boundary in the low-temperature and low-pressure region. Based on the phase diagrams presented by Kohse and Heidemann (1992), we infer that the current algorithm generates incorrect solutions after passing a so-called "four-phase point", i.e., the intersection point of a three-phase envelope and a four-phase region. The TR-based algorithm fails to locate such a "four-phase point" as it only considers two-phase and three-phase formulations.

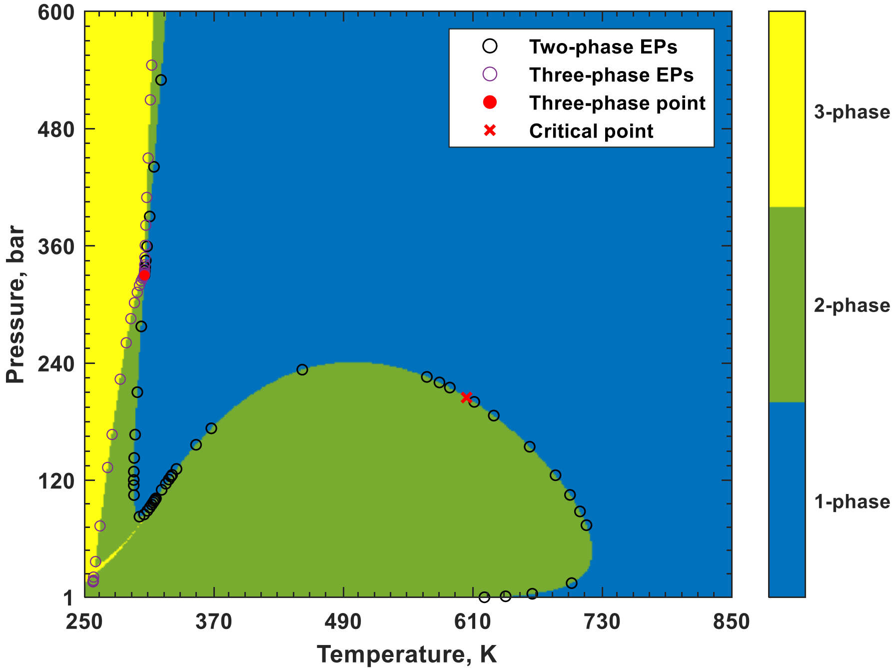
To validate the existence of four-phase equilibria, we apply four-phase equilibrium calculations (Li et al., 2023) in the low-temperature and low-pressure region of Fig. 4-7c. **Table 4-7** illustrates the equilibrium phase compositions and fractions calculated for the mixture Oilb0.71 at 256 K and 28 bar. As shown in Table 4-7, the mixture will split into four hydrocarbon phases, including two liquid phases and two vapor phases. Because of this, the current algorithm fails to construct the correct phase envelope in this region. Modifications to the TR-based algorithm are required to address this issue. Such a modified algorithm should be able to locate the "four-phase point" and construct four-phase envelopes.



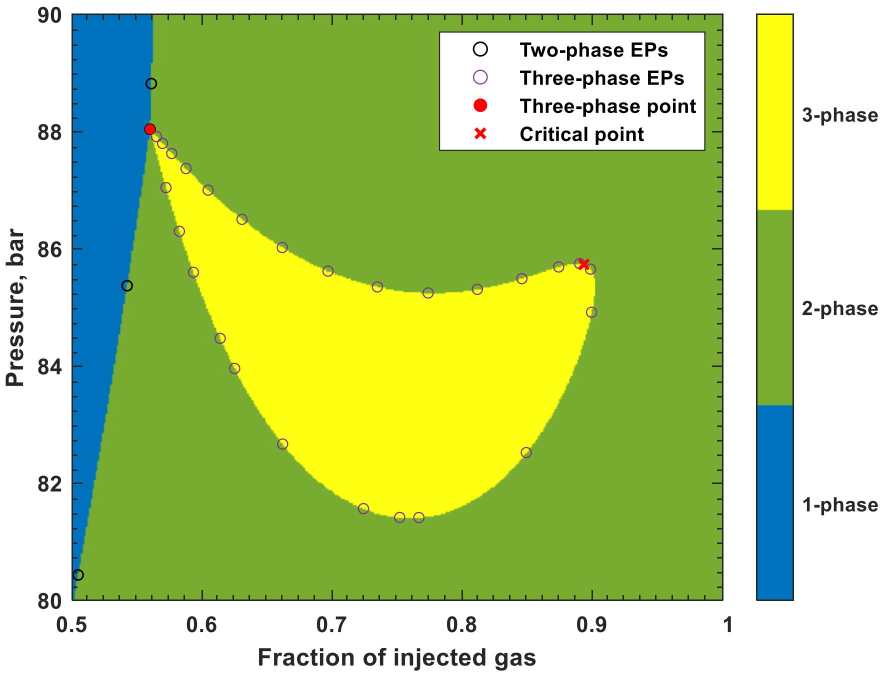
(a)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
3-phase
2-phase
1-phase
Pressure, bar
200
160
120
80
40
1
100
180
260
340
420
500
Temperature, K

(b)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
3-phase
2-phase
1-phase
Pressure, bar
600
480
360
240
120
1
250
370
490
610
730
850
Temperature, K

(c)



(d)

Fig. 4-7. Discretized phase diagrams, together with EPs, generated for four representative cases: (a) NWE0.7; (b) Zick0.6; (c) Oilb0.71; (d) JEMA-Px. The discretized diagram is generated by a total of 400 × 400 = 160,000 runs of multiphase equilibrium calculations.

Table 4-7. Phase compositions at equilibrium calculated for case Oilb0.71 at 256 K and 28 bar.

Components	Light vapor phase	Heavy vapor phase	Light liquid phase	Heavy liquid phase
CO_2	7.523E-01	8.967E-01	6.100E-01	5.557E-01
N_2	1.286E-02	9.999E-04	9.683E-04	1.101E-03
CH_4	2.233E-01	4.062E-02	4.119E-02	4.005E-02
C_2	8.731E-03	9.004E-03	1.328E-02	1.319E-02
C_3	1.817E-03	6.209E-03	1.028E-02	9.759E-03
iC_4	9.248E-05	6.592E-04	1.304E-03	1.230E-03
C_4	5.860E-04	6.113E-03	1.189E-02	1.088E-02
iC_5	1.098E-04	2.634E-03	5.884E-03	5.297E-03
C_5	1.106E-04	3.558E-03	8.027E-03	7.102E-03
C_6	6.298E-05	4.955E-03	1.270E-02	1.069E-02
$C_{7+}(1)$	3.328E-05	1.807E-02	7.553E-02	6.373E-02
$C_{7+}(2)$	6.249E-07	8.058E-03	7.154E-02	5.974E-02
$C_{7+}(3)$	4.513E-09	2.068E-03	5.770E-02	5.174E-02
$C_{7+}(4)$	1.320E-11	2.919E-04	4.436E-02	4.883E-02
$C_{7+}(5)$	1.169E-14	1.092E-05	2.645E-02	4.888E-02
$C_{7+}(6)$	4.352E-19	8.891E-09	8.852E-03	7.209E-02
Phase fraction	0.035	0.336	0.612	0.018

4.4 Conclusions

We extend the TR-based algorithm, which is previously developed to construct two-phase envelopes by Xu and Li (2023), to handle three-phase envelopes. The developed TR-based algorithm can handle multiphase envelopes involving multiple two-phase and three-phase branches. We compare the performance of the TR algorithm with that of the conventional NR algorithm in 15 case studies. The comparison results indicate that the new TR algorithm results in a significant improvement in computational efficiency. We demonstrate that the TR-based algorithm behaves more robustly than the conventional NR-based algorithm in one challenging case study (i.e., AS-1). Furthermore, using five examples, we validate that the phase envelopes constructed with the TR algorithm predict the correct phase behavior of reservoir fluids. However, the current algorithm may yield incorrect phase envelopes when four-phase equilibria appear. The current version of the TR-based algorithm could be extended to handle four-phase envelope constructions in future work.

Nomenclature

d = sensitivity vector
F = the nonlinear equation set
g = gradient vector
H = Hessian matrix
i, j = component index
J = Jacobian matrix
k = envelope point index in the phase envelope construction
K = equilibrium ratio
m = the trust-region model function
M = merit/objective function
n = iteration index in the trust-region method
n_3 = number of three-phase points
n_b = number of three-phase envelope branches
nc = number of components
P = pressure
P_c = critical pressure

r = injection gas fraction
s = the vector for updating independent variables
S = specification variable
T = temperature
T_c = critical temperature
u = independent variables
u_{2p} = independent variables defined by Eq. 4-6
u_{3p} = independent variables defined by Eq. 4-18
u_{3pp} = independent variables defined by Eq. 4-24
u^* = solution of the independent variables
w, x, y = phase composition
z = feed
z_o = oil composition
z_g = gas composition
β = phase fraction
ΔS = the step size of the specification variable
ρ = trust-region ratio
φ = fugacity coefficient
ω = acentric factor

References

Agger, C.S., Sørensen, H., 2018. Algorithm for constructing complete asphaltene PT and Px phase diagrams. Ind. Eng. Chem. Res. 57, 392–400. https://doi.org/10.1021/acs.iecr.7b04246

Buenrostro-Gonzalez, E., Lira-Galeana, C., Gil-Villegas, A., Wu, J., 2004. Asphaltene precipitation in crude oils: Theory and experiments. AIChE J. 50, 2552–2570. https://doi.org/10.1002/aic.10243

Chen, Z., Li, R., Li, H., 2022. A new vapor-liquid-asphaltene three-phase equilibrium computation algorithm based on the free-asphaltene assumption. Fluid Phase Equilib. 556, 113392. https://doi.org/10.1016/j.fluid.2022.113392

Cismondi, M., 2018. Phase envelopes for reservoir fluids with asphaltene onset lines: an integral computation strategy for complex combinations of two- and three-phase behaviors. Energy Fuels 32, 2742–2748. https://doi.org/10.1021/acs.energyfuels.7b02790

Glasø, Ø., 1985. Generalized minimum miscibility pressure correlation. SPE J. 25, 927–934. https://doi.org/10.2118/12893-PA

Haugen, K.B., Firoozabadi, A., Sun, L., 2011. Efficient and robust three-phase split computations. AIChE J. 57, 2555–2565. https://doi.org/10.1002/aic.12452

Imai, M., Pan, H., Connolly, M., Tchelepi, H., Kurihara, M., 2019. Reduced variables method for four-phase equilibrium calculations of hydrocarbon-water-CO_2 mixtures at a low temperature. Fluid Phase Equilib. 497, 151–163. https://doi.org/10.1016/j.fluid.2019.06.002

Jamaluddin, A.K.M., Creek, J., Kabir, C.S., McFadden, J.D., D'Cruz, D., Manakalathil, J., Joshi, N., Ross, B., 2002. Laboratory techniques to measure thermodynamic asphaltene instability. J. Can. Pet. Technol. 41, 44-52. https://doi.org/10.2118/02-07-04

Jessen, K., Michelsen, M.L., Stenby, E.H., 1998. Global approach for calculation of minimum miscibility pressure. Fluid Phase Equilib. 153, 251–263. https://doi.org/10.1016/S0378-3812(98)00414-2

Johns, R.T., Orr, F.M., Jr., 1996. Miscible gas displacement of multicomponent oils. SPE J. 1, 39–50. https://doi.org/10.2118/30798-PA

Khan, S.A., Pope, G.A., Sepehrnoori, K., 1992. Fluid characterization of three-phase CO_2/oil mixtures. Presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, Oklahoma, USA, 22–24 April. https://doi.org/10.2118/24130-MS

Kohse, B.F., Heidemann, R.A., 1992. Tricritical lines and multiphase equilibria in quaternary mixtures. Fluid Phase Equilib. 75, 11–22. https://doi.org/10.1016/0378-3812(92)87003-6

Li, H., 2021. Multiphase Equilibria of Complex Reservoir Fluids: An Equation of State Modeling Approach, Springer Nature.

Li, S., Xu, L., Li, H., 2023. A robust four-phase equilibrium calculation algorithm based on the scaled trust-region method and a nonmonotonic strategy for water/hydrocarbon mixtures. To be submitted.

Li, Y.-K., Nghiem, L.X., 1982. The development of a general phase envelope construction algorithm for reservoir fluid studies. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 26–29 September. https://doi.org/10.2118/11198-MS

Li, Z., Firoozabadi, A., 2012. General strategy for stability testing and phase-split calculation in two and three phases. SPE J. 17, 1096–1107. https://doi.org/10.2118/129844-PA

Lindeloff, N., Michelsen, M.L., 2003. Phase envelope calculations for hydrocarbon-water mixtures. SPE J. 8, 298–303. https://doi.org/10.2118/85971-PA

Lu, C., Jin, Z., Li, H., Xu, L., 2021. Simple and robust algorithm for multiphase equilibrium computations at temperature and volume specifications. SPE J. 26, 2397–2416. https://doi.org/10.2118/205499-PA

Metcalfe, R.S., Yarborough, L., 1979. The effect of phase equilibria on the CO_2 displacement mechanism. SPE J. 19, 242–252. https://doi.org/10.2118/7061-PA

Michelsen, M.L., 1980. Calculation of phase envelopes and critical points for multicomponent mixtures. Fluid Phase Equilib. 4, 1–10. https://doi.org/10.1016/0378-3812(80)80001-X

Michelsen, M.L., 1982. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. https://doi.org/10.1016/0378-3812(82)85001-2

Michelsen, M.L., 1986. Some aspects of multiphase calculations. Fluid Phase Equilib. 30, 15–29. https://doi.org/10.1016/0378-3812(86)80037-1

Nocedal, J., Wright, S.J., 2006. Numerical Optimization, 2nd ed, Springer Series in Operations Research and Financial Engineering, Springer, New York. https://doi.org/10.1007/978-0-387-40065-5

Pan, H., Connolly, M., Tchelepi, H., 2019. Multiphase equilibrium calculation framework for compositional simulation of CO_2 injection in low-temperature reservoirs. Ind. Eng. Chem. Res. 58, 2052–2070. https://doi.org/10.1021/acs.iecr.8b05229

Pedersen, K., Christensen, P., 2006. Phase Behavior of Petroleum Reservoir Fluids, CRC Press. https://doi.org/10.1201/9781420018257

Petitfrere, M., Nichita, D.V., 2014. Robust and efficient trust-region based stability analysis and multiphase flash calculations. Fluid Phase Equilib. 362, 51–68. https://doi.org/10.1016/j.fluid.2013.08.039

Robinson, D.B., Peng, D., 1978. The Characterization of the Heptanes and Heavier Fractions for the GPA Peng-Robinson Programs, Gas Processors Association, Tulsa, Oklahoma.

Shelton, J.L., Yarborough, L., 1977. Multiple phase behavior in porous media during CO_2 or rich-gas flooding. J. Pet. Technol. 29, 1171–1178. https://doi.org/10.2118/5827-PA

Sun, W., Yuan, Y.X., 2006. Optimization Theory and Methods: Nonlinear Programming, Springer Science & Business Media.

Xu, L., Li, H., 2023. A new trust-region-based phase envelope construction algorithm. J. Pet. Sci. Eng. In revision. Available at SSRN: https://doi.org/10.2139/ssrn.4191872

Zhao, H., Fang, Z., 2020. Improved multiple-mixing-cell method for accelerating minimum miscibility pressure calculations. SPE J. 25, 1681–1696. https://doi.org/10.2118/199360-PA

Zick, A.A., 1986. A combined condensing/vaporizing mechanism in the displacement of oil by enriched gases. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 5–8 October. https://doi.org/10.2118/15493-MS

Appendices of Chapter 4

4-A.1 Fluid properties

Table 4-A1 to **Table 4-A8** list the properties of the non-asphaltic reservoir fluids that are examined in this study.

Table 4-A1. Fluid properties of the OilG and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.69	100	44.01	304.2	73.76	0.225	0
C_1	17.52	0	16.043	174.44	46	0.008	0.085
C_{2-3}	22.44	0	37.9086	347.26	44.69	0.1331	0.085
C_{4-6}	16.73	0	68.6715	459.74	34.18	0.2358	0.085
C_{7-14}	24.22	0	135.0933	595.14	21.87	0.5977	0.104
C_{15-25}	12.16	0	261.103	729.98	16.04	0.9118	0.104
C_{26+}	5.24	0	479.6983	910.18	15.21	1.2444	0.104

Table 4-A2. Fluid properties of the NWE oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	0.77	95	44.01	304.2	73.76	0.225	0
C_1	20.25	5	16.04	190.6	46	0.008	0.12
C_{2-3}	11.8	0	38.4	343.64	45.05	0.13	0.12
C_{4-6}	14.84	0	72.82	466.41	33.5	0.244	0.12
C_{7-14}	28.63	0	135.82	603.07	24.24	0.6	0.12
C_{15-24}	14.9	0	257.75	733.79	18.03	0.903	0.12
C_{25+}	8.81	0	479.95	923.2	17.26	1.229	0.12

Table 4-A3. Fluid properties of the JEMA oil and the injected gas (Khan et al., 1992).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	1.92	100	44.01	304.2	73.76	0.225	0
C_1	6.93	0	16.04	166.67	46	0.008	0.05
C_{2-3}	17.42	0	36.01	338.81	45.53	0.126	0.05
C_{4-6}	19.44	0	70.52	466.12	33.68	0.244	0.05
C_{7-16}	31.38	0	147.18	611.11	20.95	0.639	0.09
C_{17-29}	15.49	0	301.48	777.78	15.88	1	0.09
C_{30+}	7.42	0	562.81	972.22	15.84	1.281	0.09

Table 4-A4. Fluid properties of the MY10 oil, the injected gas, and a MY10-water mixture (Metcalfe and Yarborough, 1979; Johns and Orr, 1996; Imai et al., 2019).

Components	MY10-water composition (mol%)	MY10 composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with H_2O	BIP with CO_2	BIP with CH_4	BIP with C_2	BIP with C_3	BIP with C_4
H_2O	28.50	0	0	16	647	220.5	0.344	-	-	-	-	-	-
CO_2	5.00	0	100	44.01	304.04	73.84	0.225	0.095	-	-	-	-	-
CH_4	23.275	35	0	16.04	190.59	46.04	0.01	0.45	0.1	-	-	-	-
C_2	1.995	3	0	30.07	305.21	48.84	0.099	0.5	0.13	0	-	-	-
C_3	2.66	4	0	44.1	369.71	42.57	0.152	0.5	0.135	0	0	-	-
C_4	3.99	6	0	58.12	419.04	37.46	0.187	0.5	0.13	0	0	0	-
C_5	2.66	4	0	72.15	458.98	32.77	0.252	0.5	0.125	0	0	0	0
C_6	1.995	3	0	86.18	507.54	29.72	0.296	0.5	0.12	0.02	0.03	0.03	0.03
C_7	3.325	5	0	100.2	540.32	27.37	0.351	0.5	0.12	0.03	0.03	0.03	0.03
C_8	3.325	5	0	114.23	568.93	25.1	0.394	0.5	0.12	0.035	0.03	0.03	0.03
C_{10}	19.95	30	0	142.29	615.15	22.06	0.491	0.5	0.12	0.04	0.03	0.03	0.03
C_{14}	3.325	5	0	198.39	694.82	15.86	0.755	0.5	0.12	0.06	0.03	0.03	0.03

Table 4-A5. Fluid properties of the Glasø-A oil and the injected gas (Glasø, 1985; Zhao and Fang, 2020).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with N_2
CO_2	0.49	0.76	44.01	304.2	73.76	0.225	-	-
N_2	0.47	0.5	28.014	126.2	33.94	0.04	0	-
C_1	42.01	72.04	16.043	190.6	46.00	0.008	0.025	0.105
C_2	6.05	12.41	30.07	305.4	48.84	0.098	0.01	0.13
C_3	2.93	8.6	44.097	369.8	42.46	0.152	0.09	0.125
iC_4	0.61	1.19	58.12	408.1	36.48	0.176	0.095	0.12
nC_4	0.99	2.55	58.12	425.2	38.00	0.193	0.095	0.115
iC_5	0.58	0.58	72.15	460.4	33.84	0.227	0.1	0.115
nC_5	0.42	0.65	72.15	469.6	33.74	0.251	0.11	0.115
C_6	0.92	0.35	86.177	507.4	29.69	0.296	0.11	0.115
C_7+	44.53	0.37	196	731.04	21.99	0.578	0.11	0.115

Table 4-A6. Fluid properties of Zick-2 oil and the injected gas (Zick, 1986; Jessen et al., 1998).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2
CO_2	6.56	17.75	44.01	304.14	73.74	0.228	-
CH_4	37.11	38.78	16.043	190.6	45.92	0.008	0.12
C_2	5.38	18.8	30.07	305.4	48.75	0.098	0.15
C_3	3.73	21.96	44.097	369.8	42.38	0.152	0.15
C_4	2.61	2.71	58.123	425.2	37.93	0.193	0.15
C_5	1.87	0	72.15	469.6	33.68	0.251	0.15
C_6	2.18	0	86.177	507.4	29.64	0.296	0.15
C_{7+}(1)	17.91	0	118.3	616.2	28.83	0.454	0.15
C_{7+}(2)	9.10	0	172	698.9	19.32	0.787	0.15
C_{7+}(3)	6.05	0	236	770.4	16.59	1.048	0.15
C_{7+}(4)	4.47	0	338.8	853.1	15.27	1.276	0.15
C_{7+}(5)	3.02	0	451	1001.2	14.67	1.299	0.15

Table 4-A7. Fluid properties of the NC-P oil (Pedersen and Christensen, 2006).

Components	Oil composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with N_2	BIP with CO_2
N_2	0.46	28	126.2	33.94	0.04	0.032	-
CO_2	3.36	44	304.3	73.76	0.225	0.028	0.12
CH_4	62.36	16	190.6	46	0.008	0.041	0.12
C_2	8.9	30.1	305.5	48.84	0.098	0.076	0.12
C_3	5.31	44.1	369.9	42.46	0.152	0.094	0.12
iC_4	0.92	58.1	408.2	36.48	0.176	0.07	0.12
C_4	2.08	58.1	425.3	38	0.193	0.087	0.12
iC_5	0.73	72.2	460.5	33.84	0.227	0.088	0.12
C_5	0.85	72.2	469.7	33.74	0.251	0.08	0.12
C_6	1.05	86.2	507.5	29.69	0.296	0.08	0.1
C_7	1.85	95	532.9	31.95	0.465	0.08	0.1
C_8	1.75	106	553.1	29.44	0.497	0.08	0.1
C_9	1.4	121	575.8	26.08	0.54	0.08	0.1
C_{10}-C_{15}	4.64	165.1	635.0	20.82	0.669	0.08	0.1
C_{16}-C_{25}	2.91	265.1	732.1	15.8	0.919	0.08	0.1
C_{26}-C_{80}	1.43	488.8	908.1	13.06	1.246	0.08	0.1

Table 4-A8. Fluid properties of the Oil b and the injected gas (Shelton and Yarborough, 1977; Haugen et al., 2011; Pan et al., 2019).

Components	Oil composition (mol%)	Gas composition (mol%)	Molecular weight	T_c, K	P_c, bar	ω	BIP with CO_2	BIP with N_2	BIP with CH_4
CO_2	0.11	100	44.01	304.2	72.8	0.225	-	-	-
N_2	0.48	0	28.01	126.2	33.5	0.04	-0.02	-	-
CH_4	16.3	0	16.04	190.6	45.4	0.008	0.075	0.08	-
C_2	4.03	0	30.07	305.4	48.2	0.098	0.08	0.07	0.003
C_3	2.97	0	44.1	369.8	41.9	0.152	0.08	0.07	0.01
iC_4	0.36	0	58.12	408.1	36	0.176	0.085	0.06	0.018
C_4	3.29	0	58.12	425.2	37.5	0.193	0.085	0.06	0.018
iC_5	1.58	0	72.15	460.4	33.4	0.227	0.085	0.06	0.025
C_5	2.15	0	72.15	469.6	33.3	0.251	0.085	0.06	0.026
C_6	3.32	0	84	506.35	33.9	0.299	0.095	0.05	0.036
C_{7+}(1)	18.4152	0	112.8	566.55	25.3	0.3884	0.095	0.1	0.049
C_{7+}(2)	16.3891	0	161.2	647.06	19.1	0.5289	0.095	0.12	0.073
C_{7+}(3)	12.7256	0	223.2	719.44	14.2	0.6911	0.095	0.12	0.098
C_{7+}(4)	9.6888	0	304.4	784.93	10.5	0.8782	0.095	0.12	0.124
C_{7+}(5)	5.8808	0	417.5	846.33	7.5	1.1009	0.095	0.12	0.149
C_{7+}(6)	2.3105	0	636.8	919.39	4.76	1.4478	0.095	0.12	0.181

Fluid properties of AS-1 and AS-2 oil samples are shown in **Table 4-A9** and **Table 4-A10**. The BIPs used for AS-1 and AS-2 oil samples are shown in **Table 4-A11** and **Table 4-A12**, respectively.

Table 4-A9. Fluid properties of AS-1 oil (Jamaluddin et al., 2002; Chen et al., 2022).

Component	Composition (mol%)	T_c (K)	P_c (bar)	V_c (m^3/kmol)	ω	MW (g/mol)
N_2	0.481	126.200	33.94	89.80	0.040	28.014
CO_2	0.922	304.200	73.76	94.00	0.225	44.01
C_1	43.501	190.600	46.00	99.00	0.008	16.043
C_2	11.038	305.400	48.84	148.00	0.098	30.070
C_3	6.561	369.800	42.46	203.00	0.152	44.097
i-C_4	0.791	408.100	36.48	263.00	0.176	58.124
n-C_4	3.706	425.200	38.00	255.00	0.193	58.124
i-C_5	1.282	460.400	33.84	306.00	0.227	72.151
n-C_5	2.254	469.600	33.74	304.00	0.251	72.151
C_6	2.704	507.400	29.69	370.00	0.296	86.178
C_7	2.488	538.119	29.9	446.59	0.338	96.000
C_8	2.257	558.729	27.59	476.11	0.374	107.000

C_9	2.047	582.505	25.16	522.24	0.421	121.000
C_{10}-C_{12}	5.070	622.122	22.00	616.07	0.505	146.458
C_{13}-C_{14}	2.643	669.274	19.15	754.01	0.615	182.135
C_{15}-C_{17}	3.113	715.409	17.21	919.98	0.731	220.660
C_{18}-C_{20}	2.324	760.422	15.85	1102.36	0.846	262.221
C_{21}-C_{23}	1.734	802.703	14.91	1294.39	0.952	303.790
C_{24}-C_{27}	1.648	847.816	14.15	1518.93	1.058	350.513
C_{28}-C_{33}	1.530	910.362	13.39	1860.17	1.181	419.041
C_{34}-C_{42}	1.124	996.391	12.68	2376.67	1.275	519.014
C_{43}-C_{80}	0.692	1135.398	11.45	3547.56	1.020	721.267
C_{43}-C_{80} A	0.089	1423.076	15.48	3547.56	1.274	721.267

Table 4-A10. Fluid properties of AS-2 oil (Buenrostro-Gonzalez et al., 2004; Chen et al., 2022).

Component	Composition (mol%)	T_c (K)	P_c (bar)	V_c (m^3/kmol)	ω	MW (g/mol)
N_2	0.937	126.200	33.94	89.80	0.040	28.014
CO_2	1.617	304.200	73.76	94.00	0.225	44.010
H_2S	5.551	373.200	89.37	98.50	0.100	34.080
C_1	24.740	190.600	46.00	99.00	0.008	16.043
C_2	10.392	305.400	48.84	148.00	0.098	30.070
C_3	9.867	369.800	42.46	203.00	0.152	44.097
iC_4	1.885	408.100	36.48	263.00	0.176	58.124
nC_4	4.975	425.200	38.00	255.00	0.193	58.124
iC_5	2.338	460.400	33.84	306.00	0.227	72.151
nC_5	2.822	469.600	33.74	304.00	0.251	72.151
C_6	4.913	507.400	29.69	370.00	0.296	86.178
C_7	1.312	538.851	30.10	433.59	0.338	96.000
C_8	1.257	559.155	27.69	468.83	0.374	107.000
C_9	1.204	582.660	25.20	519.71	0.421	121.000
C_{10}-C_{16}	7.128	662.330	19.62	751.15	0.603	173.920
C_{17}-C_{22}	4.619	766.814	15.47	1151.31	0.866	268.651
C_{23}-C_{28}	3.572	846.502	13.90	1530.00	1.059	350.742
C_{29}-C_{33}	2.350	915.931	13.02	1900.57	1.195	428.801
C_{34}-C_{40}	2.548	986.513	12.41	2310.52	1.273	511.604
C_{41}-C_{47}	1.888	1065.615	11.93	2800.55	1.268	609.604
C_{48}-C_{80}	3.610	1196.515	11.11	4070.84	0.717	837.328
C_{48}-C_{80} A	0.475	1679.781	11.90	4070.84	1.274	837.328

Table 4-A11. BIPs used for AS-1 oil (Chen et al., 2022).

	N_2	CO_2	C_{43}-C_{80} A
N_2	0	-0.01700	0.08000
CO_2	-0.01700	0	0.10000
C_1	0.03110	0.12000	0.01668

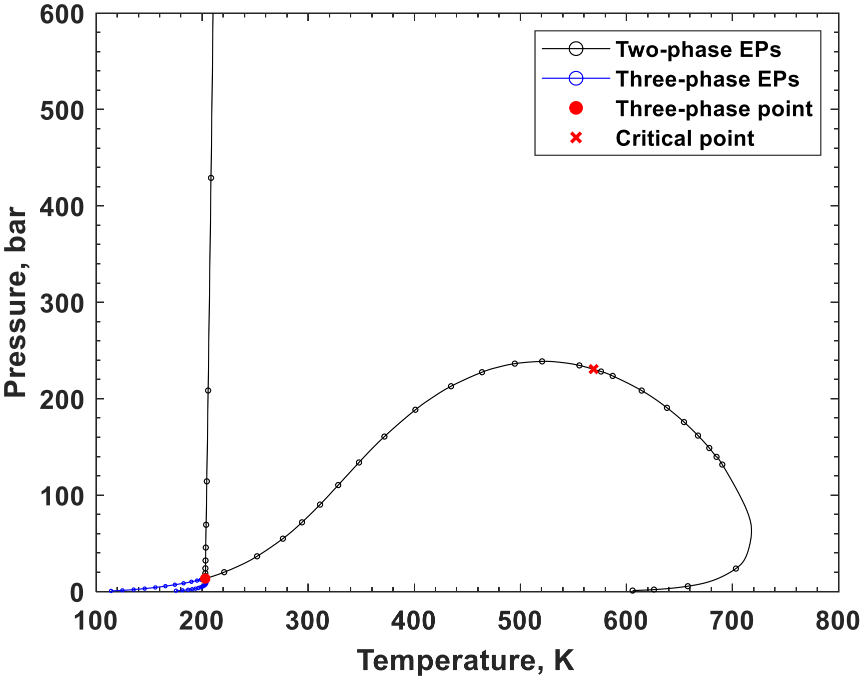
C_2	0.05150	0.12000	0.01332
C_3	0.08520	0.12000	0.01090
i-C_4	0.10330	0.12000	0.00908
n-C_4	0.08000	0.12000	0.00929
i-C_5	0.09220	0.12000	0.00808
n-C_5	0.10000	0.12000	0.00813
C_6	0.08000	0.12000	0.00691
C_7	0.08000	0.10000	0.00583
C_8	0.08000	0.10000	0.00549
C_9	0.08000	0.10000	0.00500
C_{10}-C_{12}	0.08000	0.10000	0.00419
C_{13}-C_{14}	0.08000	0.10000	0.00329
C_{15}-C_{17}	0.08000	0.10000	0.00251
C_{18}-C_{20}	0.08000	0.10000	0.00188
C_{21}-C_{23}	0.08000	0.10000	0.00140
C_{24}-C_{27}	0.08000	0.10000	0.00100
C_{28}-C_{33}	0.08000	0.10000	0.00058
C_{34}-C_{42}	0.08000	0.10000	0.00022
C_{43}-C_{80}	0.08000	0.10000	0
C_{43}-C_{80} A	0.08000	0.10000	0

Table 4-A12. BIPs used for AS-2 oil (Chen et al., 2022)

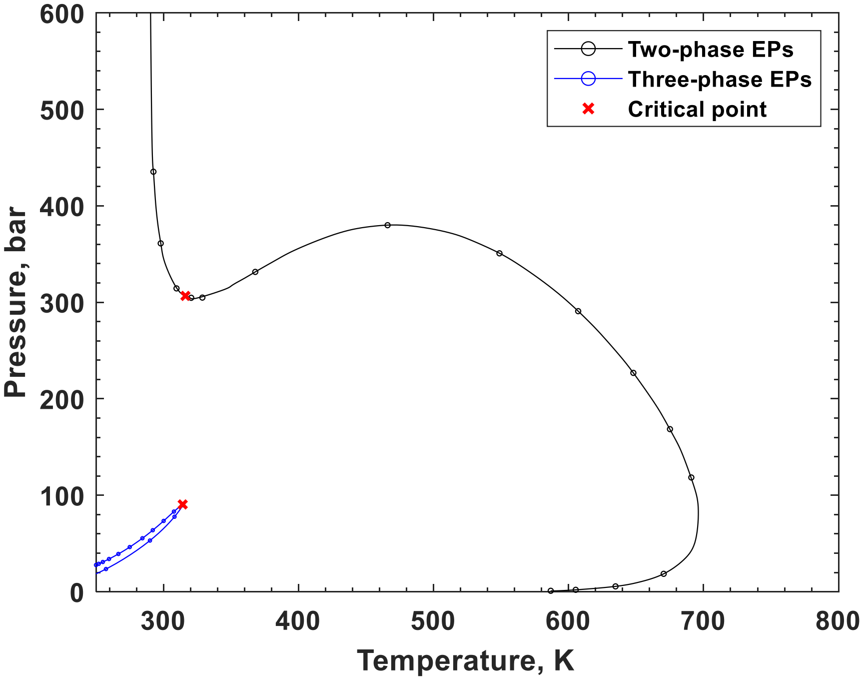
	N_2	CO_2	H_2S	C_{43}-C_{80} A
N_2	0	-0.01700	0.17700	0.08000
CO_2	-0.01700	0	0.09700	0.10000
H_2S	0.17700	0.09700	0	0.06000
C_1	0.03100	0.07000	0.08000	0.01791
C_2	0.05200	0.07000	0.08300	0.01443
C_3	0.08500	0.07000	0.08800	0.01193
iC_4	0.10300	0.07000	0.04700	0.01003
nC_4	0.08000	0.07000	0.06000	0.01025
iC_5	0.09200	0.07000	0.06000	0.00899
nC_5	0.10000	0.07000	0.06000	0.00903
C_6	0.08000	0.07000	0.06000	0.00775
C_7	0.08000	0.07000	0.06000	0.00679
C_8	0.08000	0.07000	0.06000	0.00633
C_9	0.08000	0.07000	0.06000	0.00576
C_{10}-C_{16}	0.08000	0.07000	0.06000	0.00391
C_{17}-C_{22}	0.08000	0.07000	0.06000	0.00220
C_{23}-C_{28}	0.08000	0.07000	0.06000	0.00132
C_{29}-C_{33}	0.08000	0.07000	0.06000	0.00080
C_{34}-C_{40}	0.08000	0.07000	0.06000	0.00044
C_{41}-C_{47}	0.08000	0.07000	0.06000	0.00019
C_{48}-C_{80}	0.08000	0.07000	0.06000	0
C_{48}-C_{80} A	0.00568	0.00555	0.00542	0

4-A.2 Constructed phase envelopes by the TR-based algorithm

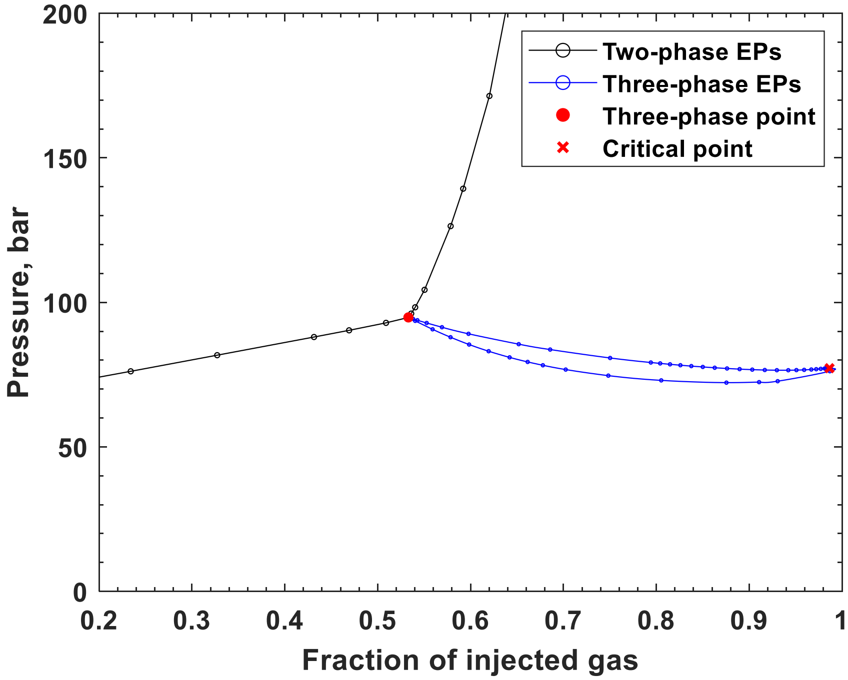
Fig. 4-A1 presents the remaining 11 phase envelopes calculated by the TR-based algorithm.



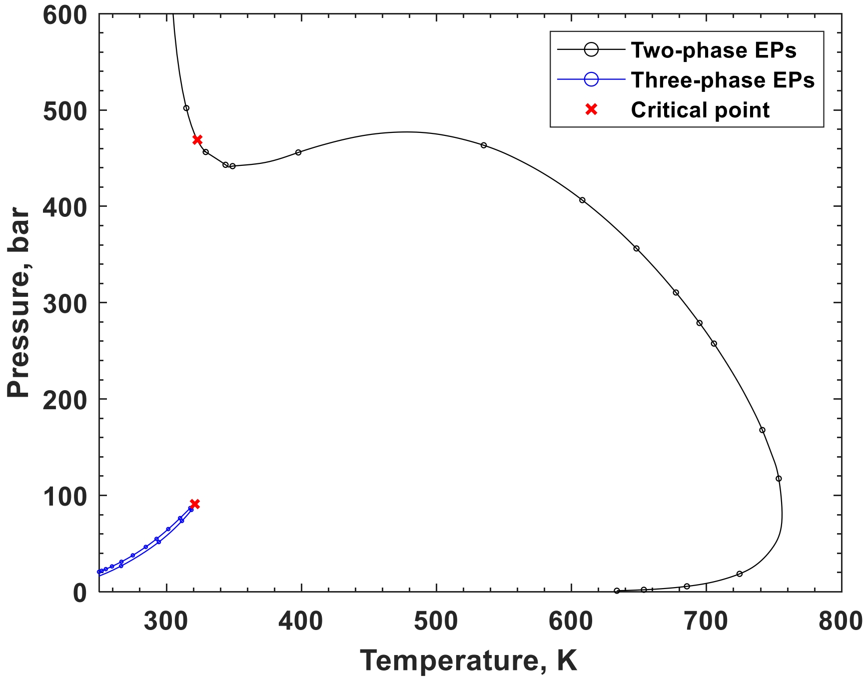
(a)



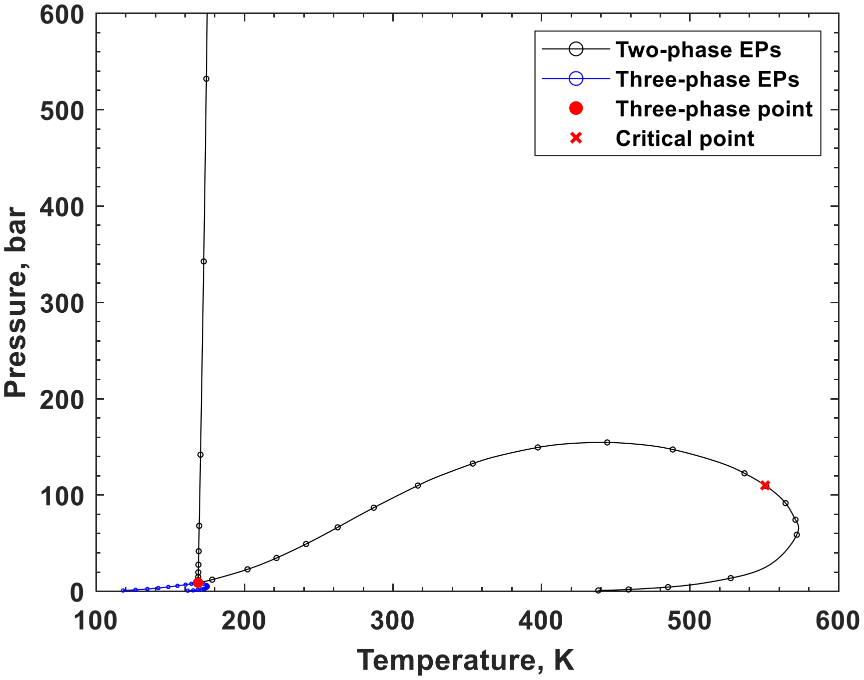
(b)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
Fraction of injected gas
0
50
100
150
200
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

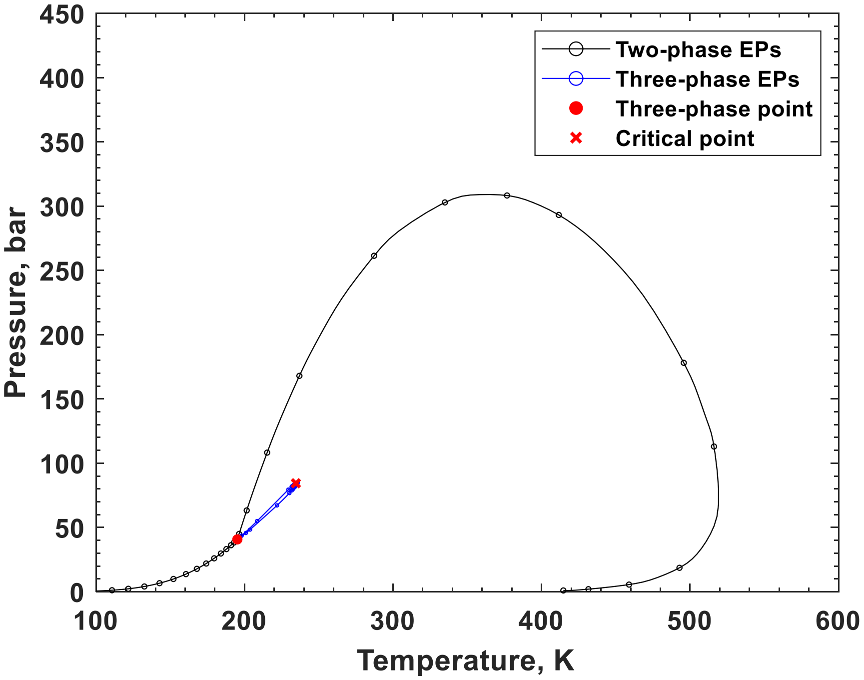
(c)


Two-phase EPs
Three-phase EPs
Critical point
Pressure, bar
Temperature, K
0
100
200
300
400
500
600
300
400
500
600
700
800

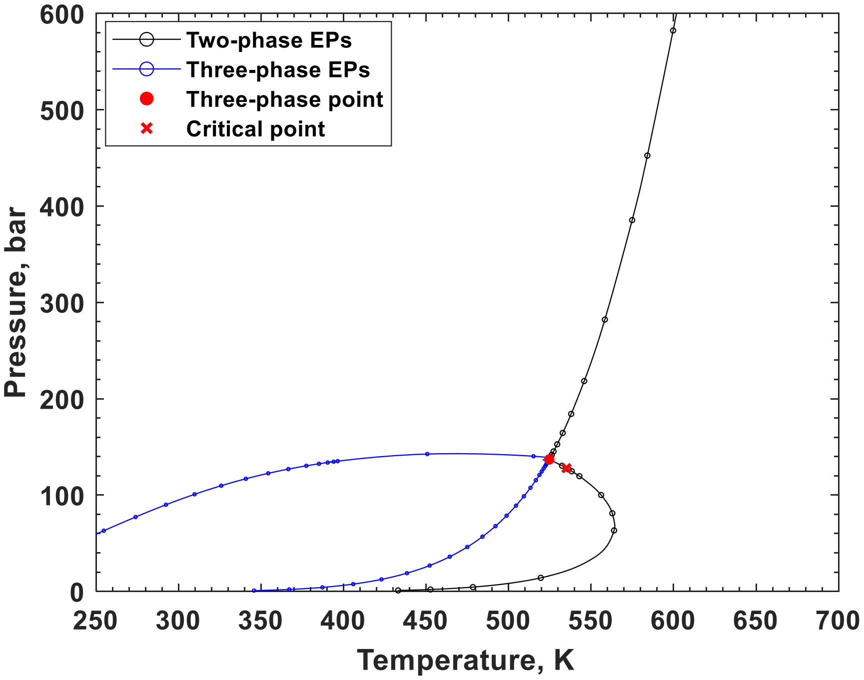
(d)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
Temperature, K
600
500
400
300
200
100
0
100
200
300
400
500
600

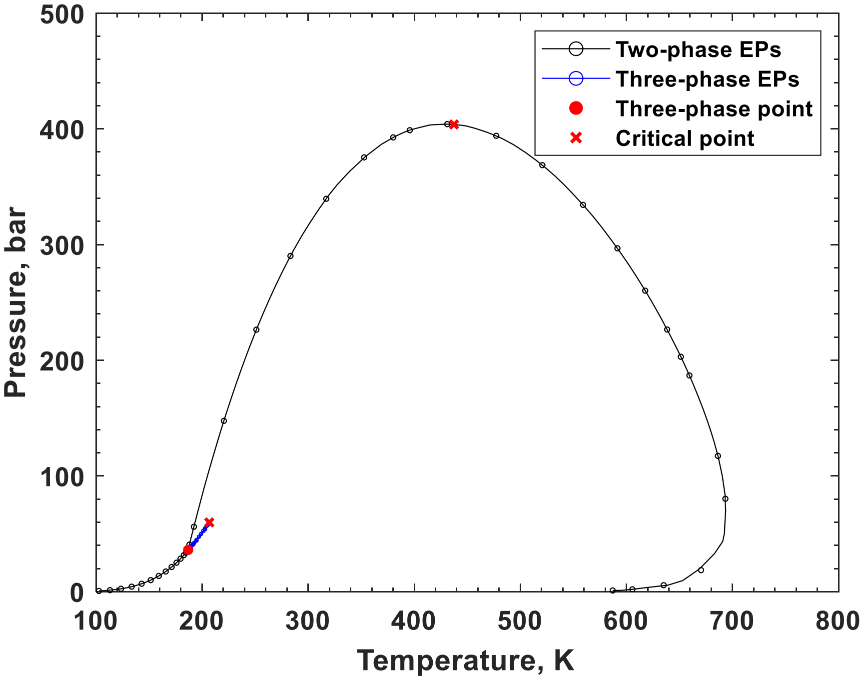
(e)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
Temperature, K
450
400
350
300
250
200
150
100
50
0
100
200
300
400
500
600

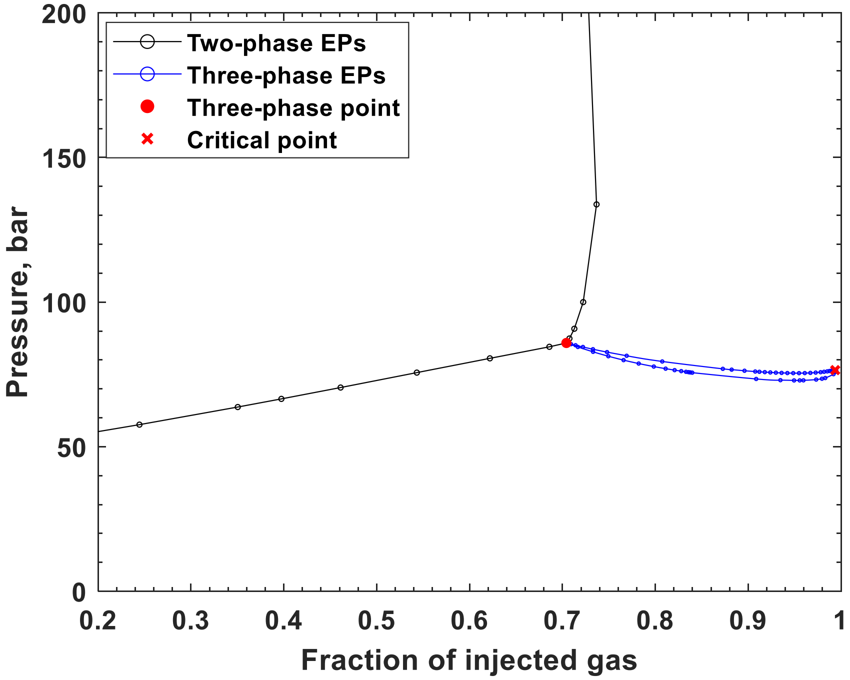
(f)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
Temperature, K
0
100
200
300
400
500
600
250
300
350
400
450
500
550
600
650
700

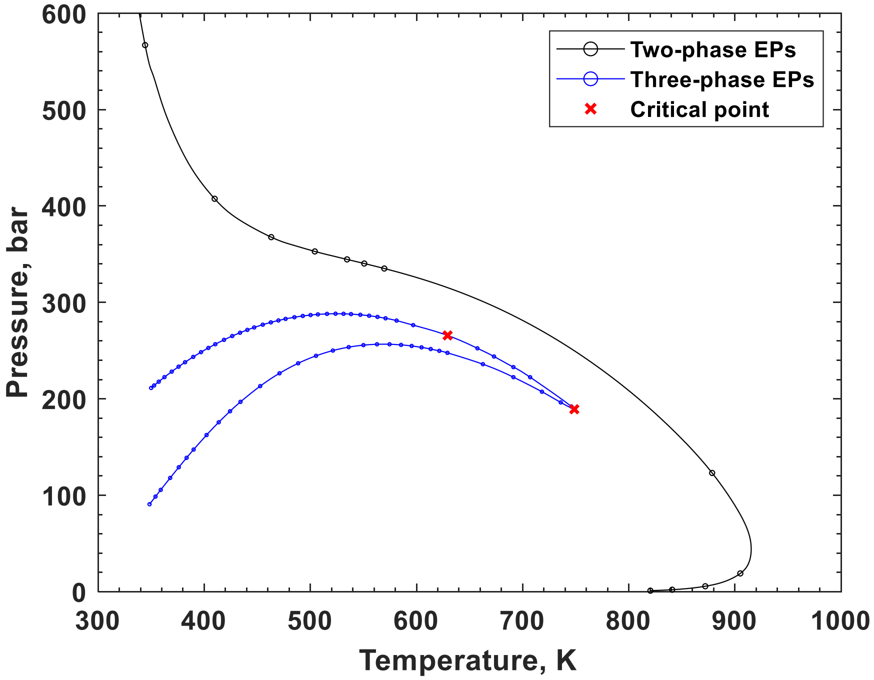
(g)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
Temperature, K
0
100
200
300
400
500
100
200
300
400
500
600
700
800

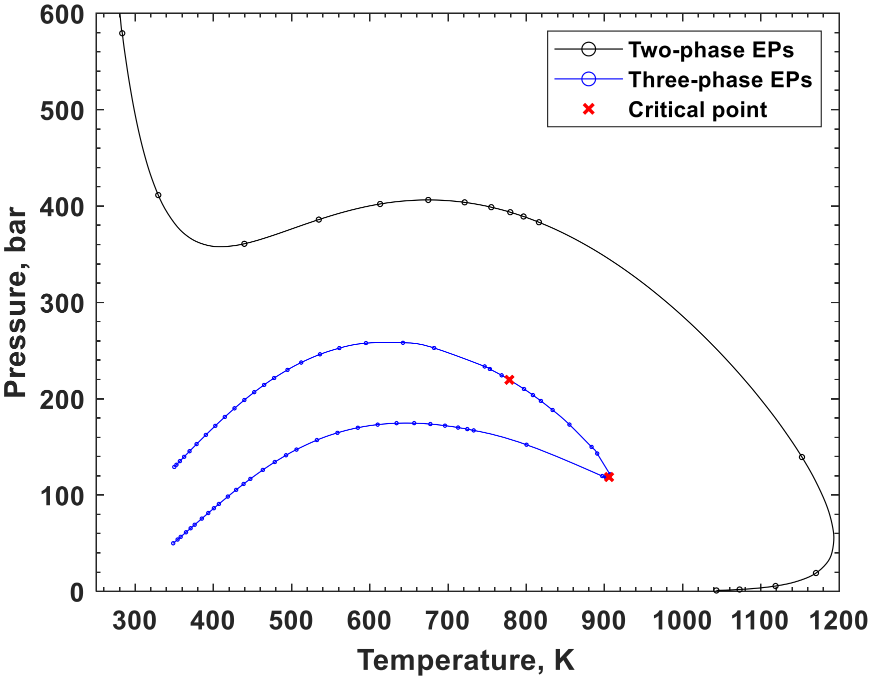
(h)


Two-phase EPs
Three-phase EPs
Three-phase point
Critical point
Pressure, bar
200
150
100
50
0
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Fraction of injected gas

(i)


Two-phase EPs
Three-phase EPs
Critical point
Pressure, bar
600
500
400
300
200
100
0
300
400
500
600
700
800
900
1000
Temperature, K

(j)



(k)

Fig. 4-A1. Phase envelopes yielded by the TR-based algorithm for the following cases: (a) OilG0.4; (b) OilG0.7; (c) OilG-Px; (d) JEMA0.75; (e) MY10-0.2; (f) Glasø0.95; (g) MY10-water; (h) NC-P; (i) Oilb-Px; (j) AS-1; (k) AS-2.

CHAPTER 5 CONCLUSIONS, CONTRIBUTIONS AND RECOMMENDATIONS

5.1 Conclusions and scientific contributions to the literature

This study develops a series of trust-region-based algorithms for improving the computational efficiency and robustness involved in flash calculations and phase envelope constructions. Specifically, the developed algorithms include a hybrid trust-region flash calculation algorithm for the simultaneous solution of equilibrium ratios and phase fractions, a trust-region-based phase envelope construction algorithm for two-phase envelope constructions, and an extended algorithm for constructing three-phase envelopes.

CHAPTER 2:

In this chapter, we develop a hybrid multiphase flash calculation algorithm where the trust-region method and Newton-Raphson method are properly hybridized to solve lnK and θ simultaneously. The hybrid algorithm is compared with two Newton-Raphson-based algorithms documented in the literature in two case studies. The comparison demonstrates that new hybrid algorithm is more robust and efficient than the two counterparts. The hybrid algorithm is then successfully applied to generate phase diagrams for a total of 6 reservoir fluid mixtures in an efficient manner.

CHAPTER 3:

In this chapter, we propose a trust-region-based phase envelope construction algorithm for constructing two-phase PT and Px envelopes. We apply a new trust-region method with an exact subproblem solver for solving the nonlinear equations to obtain the envelope points. This trust-region method is then embedded to build a new trust-region-based envelope construction algorithm. We conduct a comparative analysis between the trust-region-based algorithm and the conventional Newton-based algorithm, using a total of 17 envelope construction cases. The results indicate that the new trust-region-based algorithm behaves more robustly and efficiently than the Newton-based algorithm. Using the results calculated from multiphase equilibrium calculations, we further prove that the trust-region-based algorithm correctly predicts simple two-phase envelopes. However, the algorithm may yield incorrect solutions in cases where a three-phase point appears along the two-phase envelope. Such an issue will be addressed in chapter 4.

CHAPTER 4:

In this chapter, the trust-region-based algorithm developed in chapter 3 is extended to three-phase envelope constructions. Specifically, the developed algorithm can handle multiphase envelopes that contain multiple two-phase and three-phase branches and three-phase points. The new trust-region-based algorithm is compared to the Newton-Raphson-based algorithm in a total of 15 case studies. We observe that the new trust-region-based algorithm further improves the computational efficiency and robustness in the studied cases. In five validation examples, the phase envelopes yielded by the trust-region-based algorithm reveal the correct phase behavior of reservoir fluids. One limitation of the current algorithm lies in that it may predict incorrect phase envelopes when four-phase equilibria appear.

5.2 Suggested future works

- Extend and examine the capability of the hybrid flash calculation algorithm to handle four-phase equilibria, especially the vapor-liquid-liquid-water equilibria.
- Develop an integrated numerical algorithm for constructing phase envelopes where the three-phase regions are separated from the two-phase envelope (i.e., the phase

envelopes without a three-phase point). We have illustrated several multiphase PT/Px phase envelopes that can be successfully generated by the developed algorithm. However, a multiphase Px envelope without a three-phase point is beyond the capability of the current algorithm. We are lacking an approach that is more effective than the exhaustive searching approach. To address this issue, it is promising to combine the VL/LL phase boundary tracking algorithm developed by Chen et al. (2023) and the algorithm developed in this research to build a new integrated phase envelope construction algorithm.

- Extend the trust-region-based phase envelope algorithm to handle four-phase envelope constructions. New formulations and modifications are required to handle four-phase envelope points and four-phase points.
- Use the trust-region method to perform stability test and flash calculation with different specifications, such as the pressure-enthalpy (PH) specifications (Michelsen, 1987; Gupta et al., 1990; Zhu and Okuno, 2014, 2015, and 2016) and volume-temperature (VT) specifications (Jindrová and Mikyška, 2013 and 2015; Nichita, 2018, 2023).

References

Chen, Z., Xu, L., Zhou, Y., Li, R., Li, H., 2023. A robust and efficient algorithm for vapor-liquid-equilibrium/liquid-liquid-equilibrium (VLE/LLE) phase boundary tracking. Chem. Eng. Sci. 266, 118286. https://doi.org/10.1016/j.ces.2022.118286

Gupta, A.K., Bishnoi, P.R., Kalogerakis, N., 1990. Simultaneous multiphase isothermal/isenthalpic flash and stability calculations for reacting/non-reacting systems. Gas Sep. Purif. 4, 215–222. https://doi.org/10.1016/0950-4214(90)80045-M

Jindrová, T., Mikyška, J., 2013. Fast and robust algorithm for calculation of two-phase equilibria at given volume, temperature, and moles. Fluid Phase Equilib. 353, 101–114. https://doi.org/10.1016/j.fluid.2013.05.036

Jindrová, T., Mikyška, J., 2015. General algorithm for multiphase equilibria calculation at given volume, temperature, and moles. Fluid Phase Equilib. 393, 7–25. https://doi.org/10.1016/j.fluid.2015.02.013

Michelsen, M.L., 1987. Multiphase isenthalpic and isentropic flash algorithms. Fluid Phase Equilib. 33, 13–27. https://doi.org/10.1016/0378-3812(87)87002-4

Nichita, D.V., 2018. Volume-based phase stability testing at pressure and temperature specifications. Fluid Phase Equilib. 458, 123–141. https://doi.org/10.1016/j.fluid.2017.10.030

Nichita, D.V., 2023. Robustness and efficiency of phase stability testing at VTN and UVN conditions. Fluid Phase Equilib. 564, 113624. https://doi.org/10.1016/j.fluid.2022.113624

Zhu, D., Okuno, R., 2014. A robust algorithm for isenthalpic flash of narrow-boiling fluids. Fluid Phase Equilib. 379, 26–51. https://doi.org/10.1016/j.fluid.2014.07.003

Zhu, D., Okuno, R., 2015. Robust isenthalpic flash for multiphase water/hydrocarbon mixtures. SPE J. 20, 1350–1365. https://doi.org/10.2118/170092-PA

Zhu, D., Okuno, R., 2016. Multiphase isenthalpic flash integrated with stability analysis. Fluid Phase Equilib. 423, 203–219. https://doi.org/10.1016/j.fluid.2016.04.005

BIBLIOGRAPHY

Agger, C.S., Sørensen, H., 2018. Algorithm for constructing complete asphaltene PT and Px phase diagrams. Ind. Eng. Chem. Res. 57, 392–400. https://doi.org/10.1021/acs.iecr.7b04246

Ammar, M.N., Renon, H., 1987. The isothermal flash problem: new methods for phase split calculations. AIChE J. 33, 926–939. https://doi.org/10.1002/aic.690330606

Buenrostro-Gonzalez, E., Lira-Galeana, C., Gil-Villegas, A., Wu, J., 2004. Asphaltene precipitation in crude oils: Theory and experiments. AIChE J. 50, 2552–2570. https://doi.org/10.1002/aic.10243

Chen, Z., Li, R., Li, H., 2022. A new vapor-liquid-asphaltene three-phase equilibrium computation algorithm based on the free-asphaltene assumption. Fluid Phase Equilib. 556, 113392. https://doi.org/10.1016/j.fluid.2022.113392

Chen, Z., Xu, L., Zhou, Y., Li, R., Li, H., 2023. A robust and efficient algorithm for vapor-liquid-equilibrium/liquid-liquid-equilibrium (VLE/LLE) phase boundary tracking. Chem. Eng. Sci. 266, 118286. https://doi.org/10.1016/j.ces.2022.118286

Cismondi, M., 2018. Phase envelopes for reservoir fluids with asphaltene onset lines: an integral computation strategy for complex combinations of two- and three-phase behaviors. Energy Fuels 32, 2742–2748. https://doi.org/10.1021/acs.energyfuels.7b02790

Cismondi, M., Michelsen, M.L., 2007a. Global phase equilibrium calculations: Critical lines, critical end points and liquid–liquid–vapour equilibrium in binary mixtures. J. Supercrit. Fluids. 39, 287–295. https://doi.org/10.1016/j.supflu.2006.03.011

Cismondi, M., Michelsen, M., 2007b. Automated calculation of complete Pxy and Txy diagrams for binary systems. Fluid Phase Equilib. 259, 228–234. https://doi.org/10.1016/j.fluid.2007.07.019

Cismondi, M., Michelsen, M.L., Zabaloy, M.S., 2008. Automated Generation of Phase Diagrams for Binary Systems with Azeotropic Behavior. Ind. Eng. Chem. Res. 47, 9728–9743. https://doi.org/10.1021/ie8002914

Conn, A.R., Gould, N.I.M., Toint, P.L., 2000. Trust Region Methods, MOS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9780898719857

Deiters, U.K., Bell, I.H., 2019. Calculation of phase envelopes of fluid mixtures through parametric marching. AIChE J. 65, e16730. https://doi.org/10.1002/aic.16730

Firoozabadi, A., 2015. Thermodynamics and Applications of Hydrocarbons Energy Production, McGraw Hill Professional.

Gill, P.E., Murray, W., 1974. Newton-type methods for unconstrained and linearly constrained optimization. Math. Program. 7, 311–350.

https://doi.org/10.1007/BF01585529

Glasø, Ø., 1985. Generalized minimum miscibility pressure correlation. SPE J. 25, 927–934. https://doi.org/10.2118/12893-PA

Gupta, A.K., Bishnoi, P.R., Kalogerakis, N., 1990. Simultaneous multiphase isothermal/isenthalpic flash and stability calculations for reacting/non-reacting systems. Gas Sep. Purif. 4, 215–222. https://doi.org/10.1016/0950-4214(90)80045-M

Haugen, K.B., Firoozabadi, A., Sun, L., 2011. Efficient and robust three-phase split computations. AIChE J. 57, 2555–2565. https://doi.org/10.1002/aic.12452

Heidemann, R.A., 1983. Computation of high pressure phase equilibria. Fluid Phase Equilib. 14, 55–78. https://doi.org/10.1016/0378-3812(83)80115-0

Høier, L., 1997. Miscibility Variations in Compositionally Grading Petroleum Reservoirs. Ph.D. Thesis, Norwegian University of Science and Technology, Trondheim, Norway.

Hoteit, H., Firoozabadi, A., 2006. Simple phase stability-testing algorithm in the reduction method. AIChE J. 52, 2909–2920. https://doi.org/10.1002/aic.10908

Imai, M., Pan, H., Connolly, M., Tchelepi, H., Kurihara, M., 2019. Reduced variables method for four-phase equilibrium calculations of hydrocarbon-water-CO_2 mixtures at a low temperature. Fluid Phase Equilib. 497, 151–163. https://doi.org/10.1016/j.fluid.2019.06.002

Jamaluddin, A.K.M., Creek, J., Kabir, C.S., McFadden, J.D., D'Cruz, D., Manakalathil, J., Joshi, N., Ross, B., 2002. Laboratory techniques to measure thermodynamic asphaltene instability. J. Can. Pet. Technol. 41, 44-52. https://doi.org/10.2118/02-07-04

Jessen, K., Michelsen, M., Stenby, E.H., 1998. Global approach for calculating minimum miscibility pressure. Fluid Phase Equilib. 153, 251–263. https://doi.org/10.1016/S0378-3812(98)00414-2

Jindrová, T., Mikyška, J., 2013. Fast and robust algorithm for calculation of two-phase equilibria at given volume, temperature, and moles. Fluid Phase Equilib. 353, 101–114. https://doi.org/10.1016/j.fluid.2013.05.036

Jindrová, T., Mikyška, J., 2015. General algorithm for multiphase equilibria calculation at given volume, temperature, and moles. Fluid Phase Equilib. 393, 7–25. https://doi.org/10.1016/j.fluid.2015.02.013

Johns, R.T., Orr, F.M., Jr., 1996. Miscible gas displacement of multicomponent oils. SPE J. 1, 39–50. https://doi.org/10.2118/30798-PA

Khan, S.A., Pope, G.A., Sepehrnoori, K., 1992. Fluid characterization of three-phase CO_2/oil mixtures. Presented at the SPE/DOE Enhanced Oil Recovery Symposium, Tulsa, Oklahoma, USA, 22–24 April. https://doi.org/10.2118/24130-MS

Kohse, B.F., Heidemann, R.A., 1992. Tricritical lines and multiphase equilibria in quaternary mixtures. Fluid Phase Equilib. 75, 11–22. https://doi.org/10.1016/0378-3812(92)87003-6

Kunz, O., Klimeck, R., Wagner, W., Jaeschke, M., 2007. The GERG-2004 Wide-Range Equation of State for Natural Gases and Other Mixtures, VDI Verlag GmbH, Verlag des Vereins Deutscher Ingenieure, Düsseldorf, Germany.

Li, H., 2021. Multiphase Equilibria of Complex Reservoir Fluids: An Equation of State Modeling Approach, Springer Nature.

Li, H., Yang, D., Li, X., 2013. Determination of three-phase boundaries of solvent(s)–CO_2–heavy oil systems under reservoir conditions. Energy Fuels 27, 145–153. https://doi.org/10.1021/ef301549a

Li, S., Xu, L., Li, H., 2023. A robust four-phase equilibrium calculation algorithm based on the scaled trust-region method and a nonmonotonic strategy for water/hydrocarbon mixtures. To be submitted.

Li, Y.-K., Nghiem, L.X., 1982. The development of a general phase envelope construction algorithm for reservoir fluid studies. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 26–29 September. https://doi.org/10.2118/11198-MS

Li, Z., Firoozabadi, A., 2012. General strategy for stability testing and phase-split calculation in two and three phases. SPE J. 17, 1096–1107. https://doi.org/10.2118/129844-PA

Lindeloff, N., Michelsen, M.L., 2003. Phase envelope calculations for hydrocarbon-water mixtures. SPE J. 8, 298–303. https://doi.org/10.2118/85971-PA

Lu, C., Jin, Z., Li, H., Xu, L., 2021. Simple and robust algorithm for multiphase equilibrium computations at temperature and volume specifications. SPE J. 26, 2397–2416. https://doi.org/10.2118/205499-PA

Medeiros, F.d.A., Stenby, E.H., Yan, W., 2022. Saturation point and phase envelope calculation for reactive systems based on the RAND formulation. Chem. Eng. Sci. 247, 116911. https://doi.org/10.1016/j.ces.2021.116911

Mehra, R.K., Heidemann, R.A., Aziz, K., 1983. An accelerated successive substitution algorithm. Can. J. Chem. Eng. 61, 590–596. https://doi.org/10.1002/cjce.5450610414

Metcalfe R.S., Yarborough, L., 1979. The effect of phase equilibria on the CO_2 displacement mechanism. SPE J., 19, 242-252. https://doi.org/10.2118/7061-PA

Michelsen, M.L., 1980. Calculation of phase envelopes and critical points for multicomponent mixtures. Fluid Phase Equilib. 4, 1–10. https://doi.org/10.1016/0378-3812(80)80001-X

Michelsen, M.L., 1982a. The isothermal flash problem. Part I. Stability. Fluid Phase Equilib. 9, 1–19. https://doi.org/10.1016/0378-3812(82)85001-2

Michelsen, M.L., 1982b. The isothermal flash problem. Part II. Phase-split calculation. Fluid Phase Equilib. 9, 21–40. https://doi.org/10.1016/0378-3812(82)85002-4

Michelsen, M.L., 1986. Some aspects of multiphase calculations. Fluid Phase Equilib. 30, 15–29. https://doi.org/10.1016/0378-3812(86)80037-1

Michelsen, M.L., 1987. Multiphase isenthalpic and isentropic flash algorithms. Fluid Phase Equilib. 33, 13–27. https://doi.org/10.1016/0378-3812(87)87002-4

Michelsen, M.L., Mollerup, J.M., 2007. Thermodynamic Models: Fundamentals & Computational Aspects, 2nd ed, Tie-Line Publications, Holte, Denmark.

NETL, 2010. Carbon Dioxide Enhanced Oil Recovery: Untapped Domestic Energy Supply and Long Term Carbon Storage Solution, US Department of Energy.

Nghiem, L.X., Li, Y.-K., 1984. Computation of multiphase equilibrium phenomena with an equation of state. Fluid Phase Equilibr. 17, 77–95. https://doi.org/10.1016/0378-3812(84)80013-8

Nichita, D.V., 2018. Density-based phase envelope construction. Fluid Phase Equilib. 478, 100–113. https://doi.org/10.1016/j.fluid.2018.09.007

Nichita, D.V., 2018. Volume-based phase stability testing at pressure and temperature specifications. Fluid Phase Equilib. 458, 123–141. https://doi.org/10.1016/j.fluid.2017.10.030

Nichita, D.V., 2023. Robustness and efficiency of phase stability testing at VTN and UVN conditions. Fluid Phase Equilib. 564, 113624. https://doi.org/10.1016/j.fluid.2022.113624

Nichita, D.V., Gomez, S., Luna, E., 2002. Multiphase equilibria calculation by direct minimization of Gibbs free energy with a global optimization method. Comput. Chem. Eng. 26, 1703–1724. https://doi.org/10.1016/S0098-1354(02)00144-8

Nikolaidis, I.K., Boulougouris, G.C., Peristeras, L.D., Economou, I.G., 2019. Efficient and robust methods for direct saturation point calculations, Fluid Phase Equilib. 500, 112242. https://doi.org/10.1016/j.fluid.2019.112242

Nikolaidis, I.K., Boulougouris, G.C., Peristeras, L.D., Economou, I.G., 2020. Construction of phase envelopes for binary and multicomponent mixtures with Euler-Newton predictor-corrector methods. Fluid Phase Equilib. 505, 112338. https://doi.org/10.1016/j.fluid.2019.112338

Nikolaidis, I.K., Economou, I.G., Boulougouris, G.C., Peristeras, L.D., 2016. Calculation of the phase envelope of multicomponent mixtures with the bead spring method. AIChE J. 62, 868–879. https://doi.org/10.1002/aic.15064

Nocedal, J., Wright, S.J., 2006. Numerical Optimization, 2nd ed, Springer Series in Operations Research and Financial Engineering, Springer, New York. https://doi.org/10.1007/978-0-387-40065-5

Okuno, R., 2009. Modeling of Multiphase Behavior for Gas Flooding Simulation. Ph.D.

thesis, University of Texas at Austin.

Orr, F.M., Jr., Yu, A.D., Lien, C.L., 1981. Phase behavior of CO_2 and crude oil in low-temperature reservoirs. SPE J. 21, 480–492. https://doi.org/10.2118/8813-PA

Pan, H., Connolly, M., Tchelepi, H., 2019. Multiphase equilibrium calculation framework for compositional simulation of CO_2 injection in low-temperature reservoirs. Ind. Eng. Chem. Res. 58, 2052–2070. https://doi.org/10.1021/acs.iecr.8b05229

Pan, H., Firoozabadi, A., 1998. Complex multiphase equilibrium calculations by direct minimization of gibbs free energy by use of simulated annealing. SPE Reserv. Eval. Eng. 1, 36–42. https://doi.org/10.2118/37689-PA

Pan, H., Imai, M., Connolly, M., Tchelepi, H., 2021. Solution of multiphase Rachford-Rice equations by trust region method in compositional and thermal simulations. J. Pet. Sci. Eng. 200, 108150. https://doi.org/10.1016/j.petrol.2020.108150

Pedersen, K., Christensen, P., 2006. Phase Behavior of Petroleum Reservoir Fluids, CRC Press.

Pedersen, K.S., Michelsen, M.L., Fredheim, A.O., 1996. Phase equilibrium calculations for unprocessed well streams containing hydrate inhibitors. Fluid Phase Equilib. 126, 13–28. https://doi.org/10.1016/S0378-3812(96)03142-1

Peng, D.-Y., Robinson, D.B., 1976. A new two-constant equation of state. Ind. Eng. Chem. Fund. 15, 59–64. https://doi.org/10.1021/i160057a011

Perschke, D.R., 1988. Equation of State Phase Behavior Modeling for Compositional Simulatio. Ph.D. thesis, University of Texas at Austin.

Petitfrere, M., Nichita, D.V., 2014. Robust and efficient trust-region based stability analysis and multiphase flash calculations. Fluid Phase Equilib. 362, 51–68. https://doi.org/10.1016/j.fluid.2013.08.039

Petitfrere, M., Nichita, D.V., 2016. On a choice of independent variables in Newton iterations for multiphase flash calculations. Fluid Phase Equilib. 427, 147–151. https://doi.org/10.1016/j.fluid.2016.06.050

Rachford, H.H., Jr., Rice, J.D., 1952. Procedure for use of electronic digital computers in calculating flash vaporization hydrocarbon equilibrium. J. Pet. Tech. 4, 19–3. https://doi.org/10.2118/952327-G

Robinson, D.B., Peng, D., 1978. The Characterization of the Heptanes and Heavier Fractions for the GPA Peng-Robinson Programs, Gas Processors Association, Tulsa, Oklahoma.

Shelton, J.L., Yarborough, L., 1977. Multiple phase behavior in porous media during CO_2 or rich-gas flooding. J. Pet. Technol. 29, 1171–1178. https://doi.org/10.2118/5827-PA

Sun, W., Yuan, Y.-X., 2006. Optimization Theory and Methods: Nonlinear Programming, Springer.

Venkatarathnam G., 2014a. Density marching method for calculating phase envelopes, Ind. Eng. Chem. Res. 53, 3723-3730. https://doi.org/10.1021/ie403633d

Venkatarathnam G., 2014b. Density marching method for calculating phase envelopes. 2. Three-phase envelopes, Ind. Eng. Chem. Res. 53, 12122-12128. https://doi.org/10.1021/ie501838y

Wilson, G.M., 1969. A modified Redlich-Kwong equation of state, application to general physical data calculations. Presented at the 65th National AIChE Meeting, Cleveland, OH, USA.

Xu, L., Li, H., 2023. A new trust-region-based phase envelope construction algorithm. J. Pet. Sci. Eng. In revision. Available at SSRN: https://doi.org/10.2139/ssrn.4191872

Young, L.C., 2022. Compositional reservoir simulation: a review. SPE J. 1–47. https://doi.org/10.2118/208610-PA

Zhao, H., Fang, Z., 2020. Improved multiple-mixing-cell method for accelerating minimum miscibility pressure calculations. SPE J. 25, 1681–1696. https://doi.org/10.2118/199360-PA

Zhu, D., Okuno, R., 2014. A robust algorithm for isenthalpic flash of narrow-boiling fluids. Fluid Phase Equilib. 379, 26–51. https://doi.org/10.1016/j.fluid.2014.07.003

Zhu, D., Okuno, R., 2015. Robust isenthalpic flash for multiphase water/hydrocarbon mixtures. SPE J. 20, 1350–1365. https://doi.org/10.2118/170092-PA

Zhu, D., Okuno, R., 2016. Multiphase isenthalpic flash integrated with stability analysis. Fluid Phase Equilib. 423, 203–219. https://doi.org/10.1016/j.fluid.2016.04.005

Zick, A.A., 1986. A combined condensing/vaporizing mechanism in the displacement of oil by enriched gases. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, USA, 5–8 October. https://doi.org/10.2118/15493-MS