

Initialization of phase fractions in Rachford–Rice equations for robust and efficient three-phase split calculation

Zhidong Li, Abbas Firoozabadi *

Reservoir Engineering Research Institute (RERI), 595 Lytton Avenue, Suite B, Palo Alto, CA 94301, United States

ARTICLE INFO

Article history:

Received 30 March 2012
Received in revised form 19 June 2012
Accepted 26 June 2012
Available online 14 July 2012

Keyword:

Three-phase split computation

ABSTRACT

A key element of efficient and robust three-phase split calculation is the initialization of phase mole fractions. While the initial guess of equilibrium ratios is made from two-phase split calculation and two-phase stability testing, initializing phase fractions through solving three-phase Rachford–Rice equations can be a challenge especially close to the bicritical points and phase boundaries. In this research, we examine three different methods: the improved two-dimensional bisection method, the minimization method, and the direct Newton method. We present a large number of three-phase examples of various degrees of complexity to demonstrate both robustness and efficiency of all these methods. The direct Newton method combined with the starting guess of phase fractions from two-phase split calculation and two-phase stability testing is the most efficient approach.

© 2012 Published by Elsevier B.V.

1. Introduction

Three-phase equilibrium computation has become increasingly important in a large number of problems in hydrocarbon energy production and various chemical processes. One important application relates to CO₂ injection in hydrocarbon reservoirs for improved oil recovery where a gas phase and two liquid phases may form. The effect of three different phases with different mobilities may be considerable on flow path and oil recovery. Phase equilibrium computations are often performed billions of times in large-scale compositional reservoir simulation. Michelsen suggested a step-wise procedure to implement phase equilibrium computation where stability analysis and phase split calculations are performed alternatively [1]. Fig. 1 schematically shows the procedure of equilibrium computation for two and three phases. In our recent work, single-phase stability testing, two-phase split calculation, and two-phase stability testing are presented for efficiency and robustness [2]. One important aspect of three-phase split calculation, i.e., the initialization of phase fractions, is discussed in this paper.

Three-phase split calculation is constructed based on two sets of equations. The first set represents the equilibrium condition that relates the equilibrium ratios to the fugacity coefficients

$$\begin{cases} \ln K_{i,y} = \ln \varphi_{i,x} - \ln \varphi_{i,y} \\ \ln K_{i,z} = \ln \varphi_{i,x} - \ln \varphi_{i,z} \end{cases} \quad (i = 1, \dots, C). \quad (1)$$

C is the number of components. $\{K_{i,y} = y_i/x_i\}$ and $\{K_{i,z} = z_i/x_i\}$ are the equilibrium ratios of components in phases y and z with respect to phase x , respectively. $\{x_i\}$, $\{y_i\}$ and $\{z_i\}$ are the mole fractions of components in phases x , y and z , respectively. $\{\varphi_{i,x}\}$, $\{\varphi_{i,y}\}$ and $\{\varphi_{i,z}\}$ are the corresponding fugacity coefficients of components. The second set known as the Rachford–Rice (RR) equations [3] describes the material balance

$$\begin{cases} RR_y = \sum_{i=1}^C (y_i - x_i) = \sum_{i=1}^C \frac{n_i(K_{i,y} - 1)}{1 + \beta_y(K_{i,y} - 1) + \beta_z(K_{i,z} - 1)} = 0 \\ RR_z = \sum_{i=1}^C (z_i - x_i) = \sum_{i=1}^C \frac{n_i(K_{i,z} - 1)}{1 + \beta_y(K_{i,y} - 1) + \beta_z(K_{i,z} - 1)} = 0 \end{cases} \quad (2)$$

$\{n_i\}$ are the overall mole fractions of components. β_y and β_z are the mole fractions of phases y and z , respectively. Note that RR_y and RR_z are monotonic with respect to only β_y and β_z , respectively. For given $\{K_{i,y}\}$, $\{K_{i,z}\}$, β_y and β_z , $\{x_i\}$, $\{y_i\}$ and $\{z_i\}$ are determined from

$$\begin{aligned} x_i &= \frac{n_i}{1 + \beta_y(K_{i,y} - 1) + \beta_z(K_{i,z} - 1)}, \quad y_i = K_{i,y}x_i, \\ z_i &= K_{i,z}x_i \quad (i = 1, \dots, C). \end{aligned} \quad (3)$$

The most robust and efficient algorithm to solve Eqs. (1) and (2) combined with Eq. (3) for the $2C + 2$ unknowns, $\{K_{i,y}\}$, $\{K_{i,z}\}$, β_y and β_z , is a hybrid approach based on the successive substitution iteration (SSI) method followed by the Newton method [2]. The SSI is used to provide a good enough estimate for the Newton method. At each SSI step, $\{K_{i,y}\}$ and $\{K_{i,z}\}$ are updated through Eq. (1) in the outer loop. β_y and β_z are updated through solving Eq. (2) by the

* Corresponding author. Tel.: +1 650 326 9259; fax: +1 650 472 9285.
E-mail address: af@rerinst.org (A. Firoozabadi).

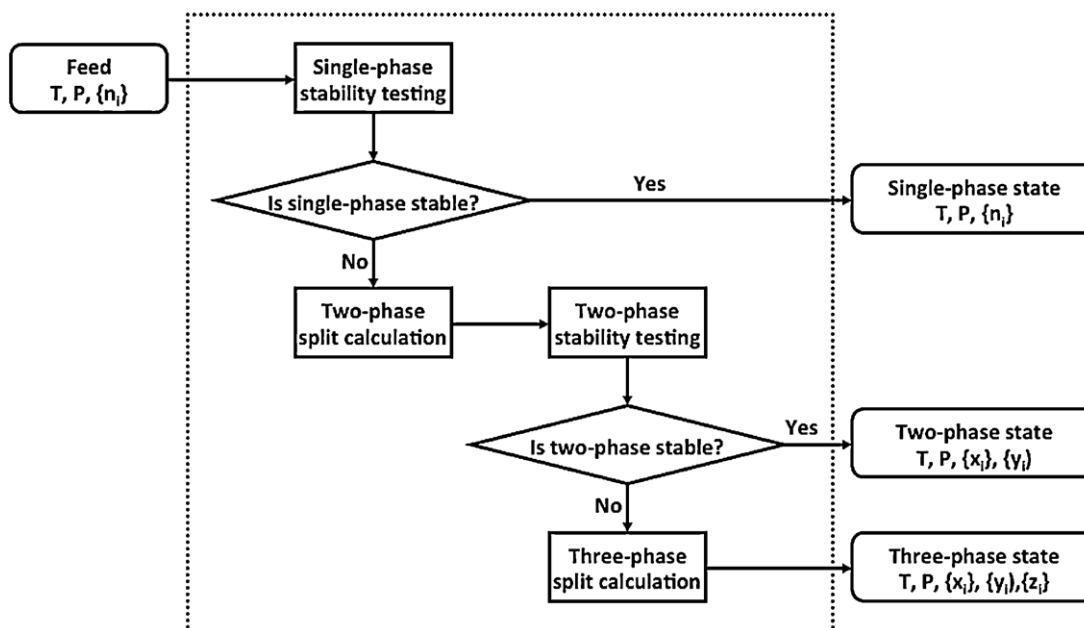


Fig. 1. Schematic flow chart of equilibrium computation for two and three phases.

Newton method in the inner loop with updated $\{K_{i,y}\}$ and $\{K_{i,z}\}$. Once the accuracy in the SSI meets a predefined switching criterion, we turn to the Newton method to solve for all the unknowns simultaneously until the accuracy meets a preset tolerance. In both SSI and Newton methods, $\{x_i\}$, $\{y_i\}$ and $\{z_i\}$ are updated through Eq. (3) to calculate the fugacity coefficients (and fugacity coefficient derivatives). In order to start the SSI, the initial guess of $\{K_{i,y}\}$, $\{K_{i,z}\}$, β_y and β_z is required. The best initial guess of $\{K_{i,y}\}$ and $\{K_{i,z}\}$ is directly made from two-phase split calculation and two-phase stability testing [1,2]. The details on how to get the initial guess of β_y and β_z robustly and efficiently are provided in this article.

The initialization of β_y and β_z requires solving the deceptively simple-looking Eq. (2) for β_y and β_z corresponding to the initial guess of $\{K_{i,y}\}$ and $\{K_{i,z}\}$. Three types of algorithms have been suggested. To avoid the confusion, we use the 'starting guess of β_y and β_z ' to represent the input (initial estimate) of the algorithm since the output (solution) will be the 'initial guess of β_y and β_z ' to start the SSI in three-phase split calculation. Haugen et al. [4] proposed the idea of a geometric approach – the two-dimensional (2D) bisection method, to solve Eq. (2). This approach does not require the starting guess of β_y and β_z and in principle should always work. In this paper, we significantly improve the work by Haugen et al. [4] and make it both robust and efficient. Eq. (2) can be solved directly by the Newton algorithm; a starting guess of β_y and β_z is required. Nelson [5] and Bunz et al. [6] solved a transform of Eq. (2) by using the Newton method. The starting guess of β_y and β_z is from the two-phase split. Leibovici and Neoschil [7] used the Newton approach including the line search and stated that any feasible point can be the starting guess of β_y and β_z . The feasible region is defined by the poles where the denominators in Eq. (2) are zero and RR_y and RR_z cannot be defined. Convergence problem may occur close to the poles. In this paper, we suggest using the conventional Newton algorithm to solve Eq. (2) directly with the starting guess of β_y and β_z from two-phase split calculation and two-phase stability testing. It is mathematically similar to that by Nelson [5] and Bunz et al. [6] except that these authors did not incorporate two-phase stability testing. Solving Eq. (2) also can be formulated as the minimization of a convex function whose gradients consist of $\{-RR_y, -RR_z\}$; a starting guess of β_y and β_z is needed. Michelsen [8] presented the method first and adopted the Newton algorithm with line search to

locate the minimum. Leibovici and Nichita [9] extended the method to negative flash by using the feasible region consisting of the poles as in Ref. [7]. They used a constrained optimization software to solve the minimization and 1/3 as the starting guess of β_y and β_z . Convergence difficulty still may occur close to the poles. Okuno et al. [10] improved the method further by defining a smaller feasible region without the poles and used the line-search Newton approach. They also described how to get a starting guess of β_y and β_z .

The remainder of this article is organized as follows. In Section 2 we introduce the fundamentals of the improved 2D bisection method, the minimization method by Okuno et al. [10], and the direct Newton method. In Section 3 we evaluate these methods by many three-phase examples with various degrees of complexity. In Section 4 the main results and conclusions are summarized.

2. Mathematical background

2.1. Improved 2D bisection method

Fig. 2 (upper panel) presents the surfaces of both RR_y and RR_z . In the triangle defined by the vertices (0,1), (0,0) and (1,0) that is the solution domain, the lines $RR_y = 0$ and $RR_z = 0$ intersect each other which is the solution of Eq. (2), as shown by Fig. 2 (lower panel). In the 2D bisection method, at each step, the mother-triangle is equally divided into two sub-triangles by connecting the middle of the hypotenuse and the right-angle vertex.

Haugen et al. [4] proposed that a sub-triangle is kept if both RR_y and RR_z have different signs at the three vertices and discard it otherwise. This criterion may keep a sub-triangle not containing the solution and may discard a sub-triangle containing the solution. Fig. 3 presents a case for which both RR_y and RR_z have different signs at the three vertices of a sub-triangle but the lines $RR_y = 0$ and $RR_z = 0$ do not intersect. Close to the bicritical point, there will be a large number of such sub-triangles since the lines $RR_y = 0$ and $RR_z = 0$ nearly overlap. Fig. 4 presents a case that RR_y (RR_z) may have the same signs at the three vertices of a sub-triangle but the lines $RR_y = 0$ and $RR_z = 0$ may cross each other. It is because the line $RR_y = 0$ ($RR_z = 0$) may intersect one side of the sub-triangle twice due to the non-monotonicity of RR_y (RR_z) in β_z (β_y) direction.

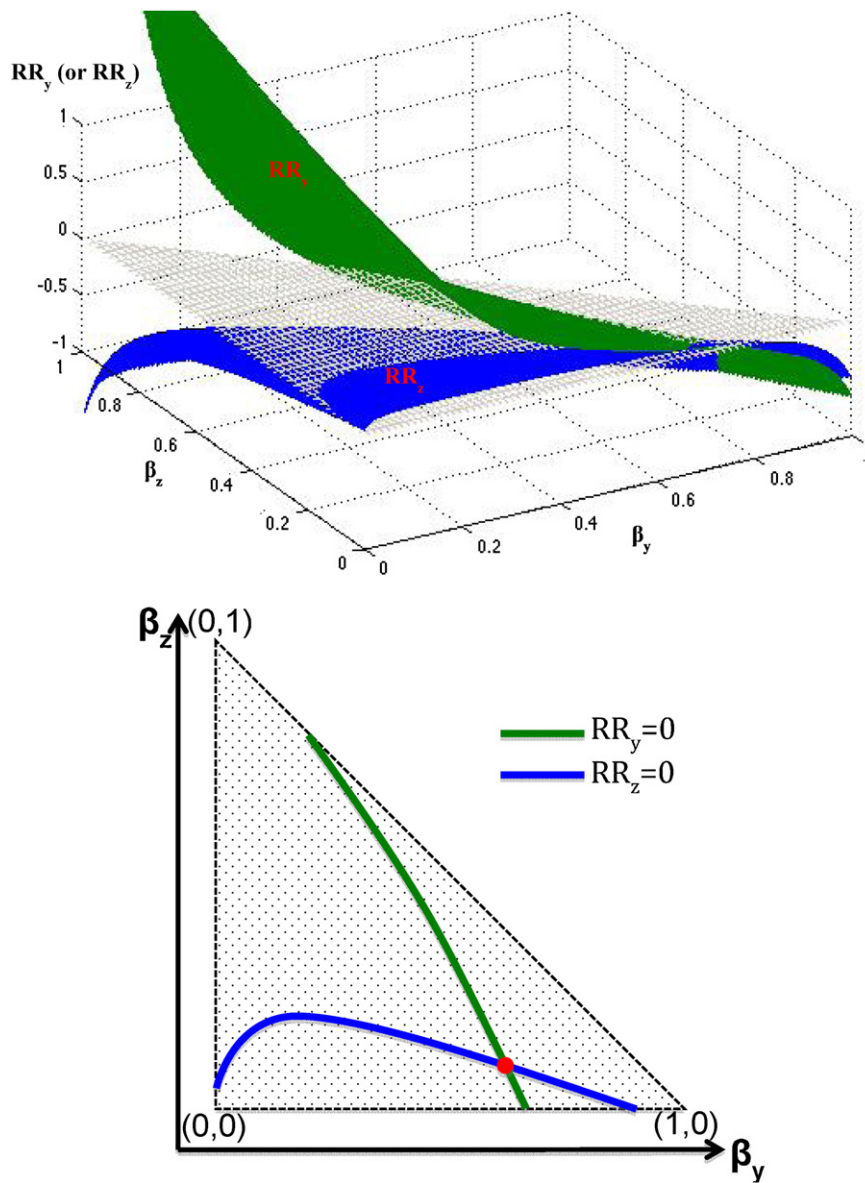


Fig. 2. Upper panel: the surfaces of RR_y and RR_z intersect the triangular solution domain. Lower panel (schematic): the solution domain defined by the vertices $(0,1)$, $(0,0)$ and $(1,0)$ contains the two lines $RR_y=0$ and $RR_z=0$. Their intersection is the solution of Eq. (2). $\{n_i\}$, $\{K_{i,j}\}$ and $\{K_{i,z}\}$ are provided in Table 1.

In our modification, at each dividing step, if sub-triangles belong to the Type I as shown in Fig. 5 (left panel) where the hypotenuse is parallel to one of the axes, they are kept due to the complexity shown in Fig. 4. If sub-triangles belong to the Type II as shown in Fig. 5 (right panel) where both right-angle sides are parallel to the axes, the following criterion is applied to determine whether they should be kept or discarded.

When both RR_y and RR_z have different signs at the three vertices of a sub-triangle, if the lines $RR_y=0$ and $RR_z=0$ cross each other, their intersections with the sides of the sub-triangle have fixed positions. If the line $RR_y=0$ ($RR_z=0$) intersects a side, RR_y (RR_z) has different signs at the two vertices of that side. Fig. 6 shows the six possibilities where the solution is in a sub-triangle for one of the configurations of Type II. We use the first possibility as an example. If a sub-triangle contains the solution, on the vertical side, the intersection of blue line should be above that of the green line and on the horizontal side the intersection of the blue line should be on the left of the green line. The coordinates of all the four intersections in the first row and of only two intersections in the second

row are required, as marked in the figure. There are 24 possibilities for all the four configurations of Type II with six possibilities for each configuration.

When RR_y has the same signs at the three vertices of a sub-triangle, if the line $RR_y=0$ intersects the vertical side (β_z direction) or the hypotenuse twice (RR_y is monotonic in β_y direction), $\partial RR_y / \partial \beta_z$ has different signs at the two vertices of that side. And RR_y at the extremum ($\partial RR_y / \partial \beta_z = 0$) between these two vertices has a different sign than the vertices. Similarly, when RR_z has the same signs at the three vertices of a sub-triangle, the signs of RR_z and $\partial RR_z / \partial \beta_y$ are used to determine whether the line $RR_z=0$ intersects the horizontal side (β_y direction) or the hypotenuse twice (RR_z is monotonic in β_z direction). The sub-triangle is kept if both lines $RR_y=0$ and $RR_z=0$ appear but may not intersect, i.e., one of the three conditions is satisfied: (1) the line $RR_y=0$ intersects one side twice and RR_z has different signs at the three vertices; (2) RR_y has different signs at the three vertices and the line $RR_y=0$ intersects one side twice; (3) both lines $RR_y=0$ and $RR_z=0$ intersect a side or sides twice.

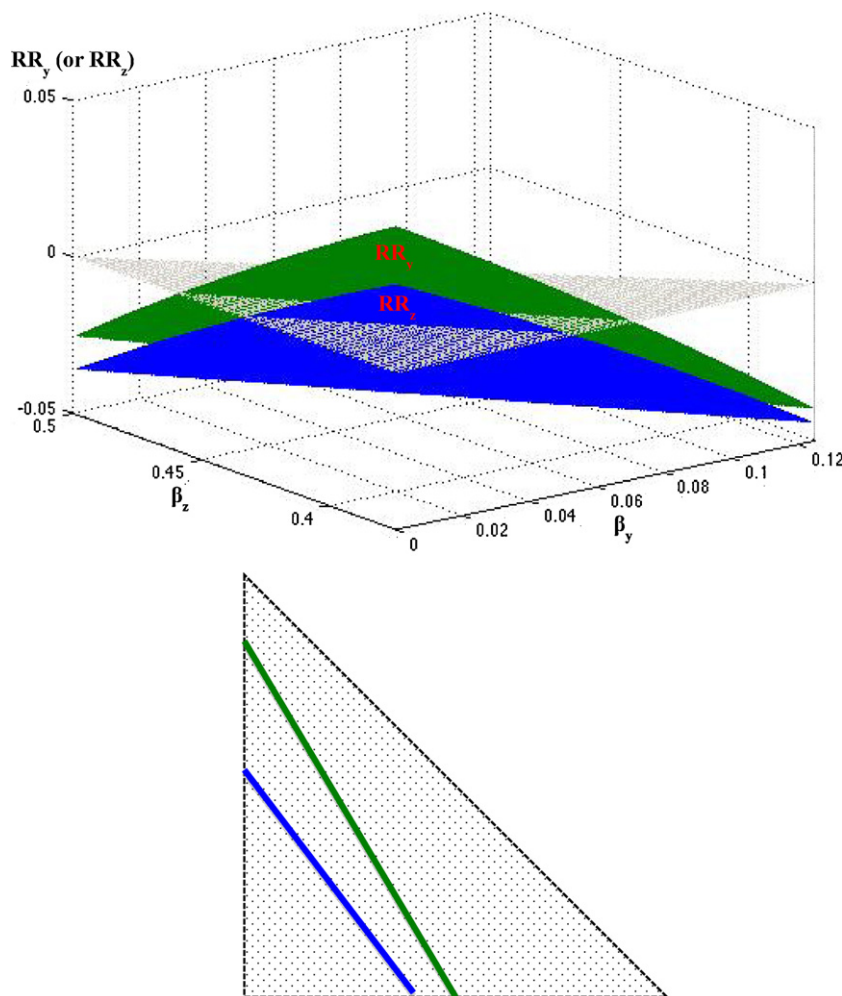


Fig. 3. A sub-triangle is kept because both RR_y and RR_z have different signs at the three vertices although it does not contain the solution using the 2D bisection method by Haugen et al. [4]. The sub-triangle is defined by (0,0.5), (0,0.375) and (0.125,0.375). $\{n_i\}$, $\{K_{i,y}\}$ and $\{K_{i,z}\}$ are provided in Table 2.

2.2. Minimization method by Okuno et al. [10]

This method is presented in Ref. [10]. We briefly review it for three phases. The convex function to be minimized is $F(\beta) = \sum_{i=1}^C \{-n_i \ln[1 + \beta_y(K_{i,y} - 1) + \beta_z(K_{i,z} - 1)]\}$ with $\beta = \{\beta_y, \beta_z\}$. Because $\{x_i\}$, $\{y_i\}$ and $\{z_i\}$ are within $[0,1]$, the feasible region is defined as $S = \{\beta \mid a_i^T \beta \leq b_i, i = 1, \dots, C\}$ where $a_i = \{1 - K_{i,y}, 1 - K_{i,z}\}$ and $b_i = \min\{1 - n_i, 1 - K_{i,y}n_i, 1 - K_{i,z}n_i\}$.

The Newton method with line search is adopted to perform the minimization. At step k , β is updated by $\beta^{(k+1)} = \beta^{(k)} + s^{(k)} \lambda_{\max}^{(k)} d^{(k)}$. $d^{(k)} = -(\nabla^2 F^{(k)})^{-1} \nabla F^{(k)}$ is the Newton direction with $\nabla F = \{-RR_y, -RR_z\}$. $\lambda_{\max}^{(k)} = \min\{1, \{(b_i - a_i^T \beta^{(k)}) / a_i^T d^{(k)} : a_i^T d^{(k)} > 0, i = 1, \dots, C\}\}$ is the maximum step length which assures that $\beta^{(k+1)}$ does not move out of the feasible region. $0 \leq s^{(k)} \leq 1$ is determined through minimizing F within $[\beta^{(k)}, \beta^{(k)} + \lambda_{\max}^{(k)} d^{(k)}]$ using the Newton method.

Ref. [10] proposed that a starting guess of β_y and β_z can be obtained by the equally weighted mean of the vertices of the common area of S and $P = \{\beta \mid \beta_y \geq 0, \beta_z \geq 0, \beta_y + \beta_z \leq 1\}$. Firstly we solve for all the intersections of any two lines among those defined by $a_i^T \beta = b_i$ ($i = 1, \dots, C$), $\beta_y + \beta_z = 1$, $\beta_y = 0$, and $\beta_z = 0$. The required vertices are the intersections within both S and P . More effort may be needed close to the bicritical point since all the lines $a_i^T \beta = b_i$ ($i = 1, \dots, C$) are nearly parallel.

2.3. Direct Newton method

We suggest the conventional Newton method to solve Eq. (2) directly. The successful implementation critically depends on the starting guess of β_y and β_z . We have tried many arbitrary starting guesses of β_y and β_z , and find that there is often failure. That is perhaps the main reason that many authors have suggested complicated and expensive alternatives. A starting guess of β_y and β_z from two-phase split calculation and two-phase stability testing can provide the success to solving Eq. (2) without additional computational cost.

We start from two-phase split calculation, and select phases x^{2P} (higher M_W) and y^{2P} (lower M_W). M_W is the molar weight. For phase y^{2P} , the equilibrium ratios are $\{K_{i,y}^{2P}\}$ with respect to phase x^{2P} and phase fraction is β_y^{2P} . In the next step where we perform two-phase stability testing, we use phase x^{2P} as the test phase. If the two-phase state $x^{2P} + y^{2P}$ is unstable, the equilibrium ratios for the trial phase with the lowest value of Gibbs tangent plane distance (TPD) function are $\{K_{i,t}^{2P}\}$ with respect to phase x^{2P} . This trial phase is selected to represent the third phase. In three-phase split calculation, $\{K_{i,y}^{2P}\}$ and $\{K_{i,t}^{2P}\}$ are the best initial guess of $\{K_{i,y}\}$ and $\{K_{i,z}\}$ [1,2]. β_y^{2P} and zero (the trial phase has infinitesimal amount) are an excellent starting guess of β_y and β_z .

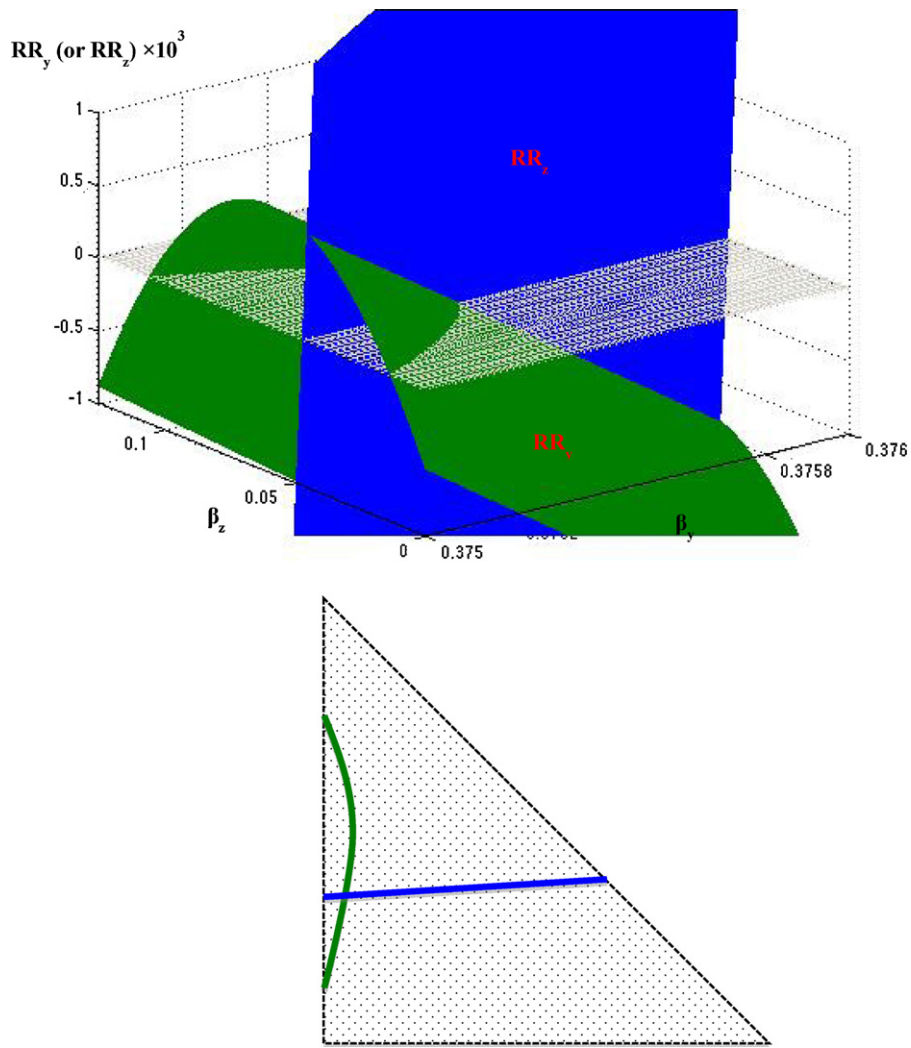


Fig. 4. A sub-triangle is discarded because RR_y has the same signs at the three vertices although it contains the solution using the 2D bisection method by Haugen et al. [4]. The right-angle side in β_y direction is only shown within $[0.375,0.376]$ in the upper panel. The sub-triangle is defined by $(0.375,0.125)$, $(0.375,0)$ and $(0.5,0)$. $\{n_i\}$, $\{K_{i,y}\}$ and $\{K_{i,z}\}$ are provided in Table 3.

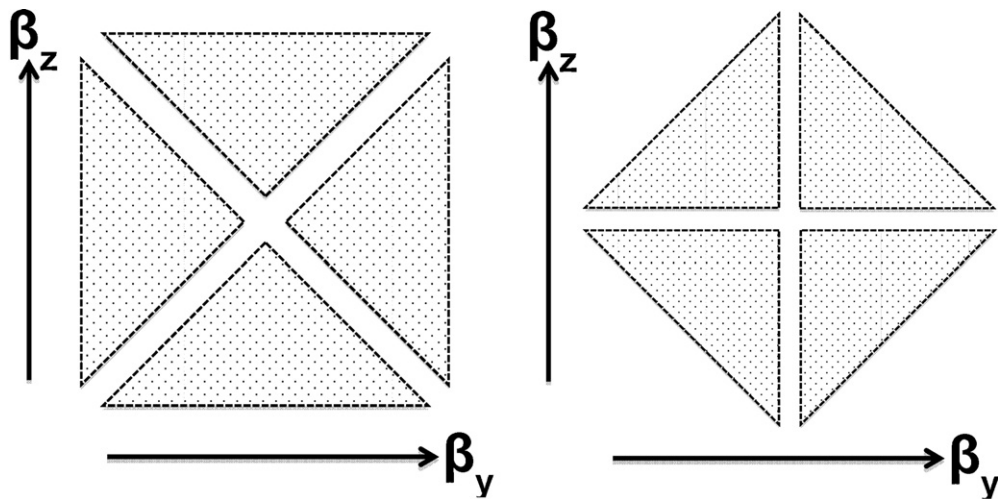


Fig. 5. Type I (left panel) and Type II (right panel) configurations of sub-triangles at each dividing step in the 2D bisection method.

Table 1
{ n_i }, { $K_{i,y}$ } and { $K_{i,z}$ } used in preparing Fig. 2.

n_i	$K_{i,y}$	$K_{i,z}$
0.47	0.886975564280731	1.85133355509695
0.126754033873246	183.729456216368	0.567851997436811
0.123759275876241	28.8439229979536	0.291644844783998
0.190491864809508	0.762796901964099	0.182989507250403
5.352678894647322e-2	6.805250689498878e-2	8.745408265736165e-2
3.546803696453197e-2	0.345376016039736	0.623957189693138

Our proposal for the starting guess of β_y and β_z is mathematically analogous to that by Nelson [5] and Bunz et al. [6]. But our starting guess has a solid foundation, i.e., it is also from two-phase split calculation and two-phase stability testing. The direct Newton method is based on a reliable two-phase stability testing that has been proposed in our recent work [2].

3. Results and discussions

We have performed 1395 three-phase equilibrium computations for CO₂ mixing with the acid gas, oil B, Maljamar reservoir oil and Maljamar separator oil [2,4,11,12], and for a gas mixture containing 95 mol% CO₂ and 5 mol% CH₄ mixing with the Bob Slaughter Block oil and North Ward Estes oil [2,13–15]. Some of the points are very close to the phase boundaries and the bicriticals. The Peng–Robinson equation of state (PR-EOS) [16,17] is applied to perform both stability testing and phase-split calculation. The EOS parameters are provided in Ref. [2].

Let Δ represent the absolute value of difference for β_y (β_z) between two consecutive iterations. In the improved 2D bisection method, to locate the intersections with sides (if both RR_y and RR_z have different signs at the three vertices of a sub-triangle) and extrema (if RR_y and/or RR_z have the same signs at the three vertices of a sub-triangle), we adopt the 1D bisection method until the length of the line is less than 10^{-5} and then the Newton method until $\Delta_{\beta_y}(\Delta_{\beta_z}) < 10^{-10}$. Similarly, to locate the solution, we also perform the improved 2D bisection method until the length of the right-angle sides of a sub-triangle is less than 10^{-5} and then the Newton method until $\max(\Delta_{\beta_y}, \Delta_{\beta_z}) < 10^{-10}$. That means the 1D or 2D bisection is applied to provide a better estimate for the Newton method. A good estimate is the mean of the two endpoints of the line in the 1D bisection or of the three vertices of the sub-triangle in the 2D bisection before switching to the Newton method. This combination improves the efficiency. The same convergence criterion, $\max(\Delta_{\beta_y}, \Delta_{\beta_z}) < 10^{-10}$, is used for both minimization and direct Newton methods.

In the improved 2D bisection method, for the acid gas sometimes two sub-triangles are kept but for the other five fluids only the sub-triangle containing the solution is kept after discarding other sub-triangles. For the Maljamar separator oil, we note three points where the ‘positive flash’ condition is not satisfied because two-phase split calculation converges to a local minimum of Gibbs free energy. In that case, two-phase split calculation is automatically

Table 4
CPU time of three-phase equilibrium computation for all the points of each fluid using different methods.

Mixture	Number of three-phase points	CPU time (s)		
		2D bisection	Minimization	Direct Newton
Acid gas	886	3.62	3.03	2.97
Oil B	80	3.26	3.14	3.06
Maljamar reservoir oil	116	7.55	7.53	7.34
Maljamar separator oil	68	0.44	0.35	0.34
Bob Slaughter Block oil	113	0.38	0.27	0.27
North Ward Estes oil	132	1.37	1.21	1.19

Table 2
{ n_i }, { $K_{i,y}$ } and { $K_{i,z}$ } used in preparing Fig. 3.

n_i	$K_{i,y}$	$K_{i,z}$
0.66731	1.40089114681102	1.42336619958799
0.09575	2.41359153035331	1.56360101270076
0.03540	0.684675481993755	0.805778846552492
0.04452	0.192706323169157	0.437918929556065
0.08589	1.344316808771735e-2	0.136423337258229
0.04470	2.913379631601974e-4	2.241151325196582e-2
0.02643	9.614643893437818e-8	3.114699395928320e-4

Table 3
{ n_i }, { $K_{i,y}$ } and { $K_{i,z}$ } used in preparing Fig. 4.

n_i	$K_{i,y}$	$K_{i,z}$
0.466	0.367489928755904	1.45983188593810
0.127710667872289	91.9551101941298	0.627700554178016
0.124693307875307	17.6437660816506	0.405472131110146
0.191929538808070	0.523968443113866	0.291902855037650
5.393076494606923e-2	5.444380423358842e-2	0.172272959622522
3.573572096426427e-2	0.192716832533260	0.704057279260822

repeated by using another initial guess of { $K_{i,y}$ } and { $K_{i,z}$ } to detect the global minimum of Gibbs free energy. The minimization and direct Newton methods do not have such a requirement.

All the three methods work robustly, particularly the direct Newton method becomes very reliable with a cost-free starting guess of β_y and β_z . We compare the efficiency of these methods in our three-phase flash code. All the runs are executed on a Dell Inspiron E1505 laptop with Intel® Core™ Duo Processor T2300 (1.66 GHz) and 1GB RAM, a five-year old machine. In Table 4, we present the CPU time of three-phase equilibrium computations for each fluid. To avoid the error from the timer, we add a loop to continuously run all the points of each fluid for 100 times. The CPU time shown in Table 4 is obtained from the total CPU time divided by 100. All the three methods work efficiently. As expected, the direct Newton method is the most efficient, followed by the minimization method by Okuno et al. [10], and then the improved 2D bisection method. For the Maljamar reservoir oil where one point is extremely close to the bicritical point, generation of the starting guess of β_y and β_z requires a higher effort so that the minimization method has similar CPU time as the improved 2D bisection method. For the other five fluids, the minimization method is only slightly more expensive than the direct Newton method. If the cost-free starting guess of β_y and β_z in the direct Newton method is

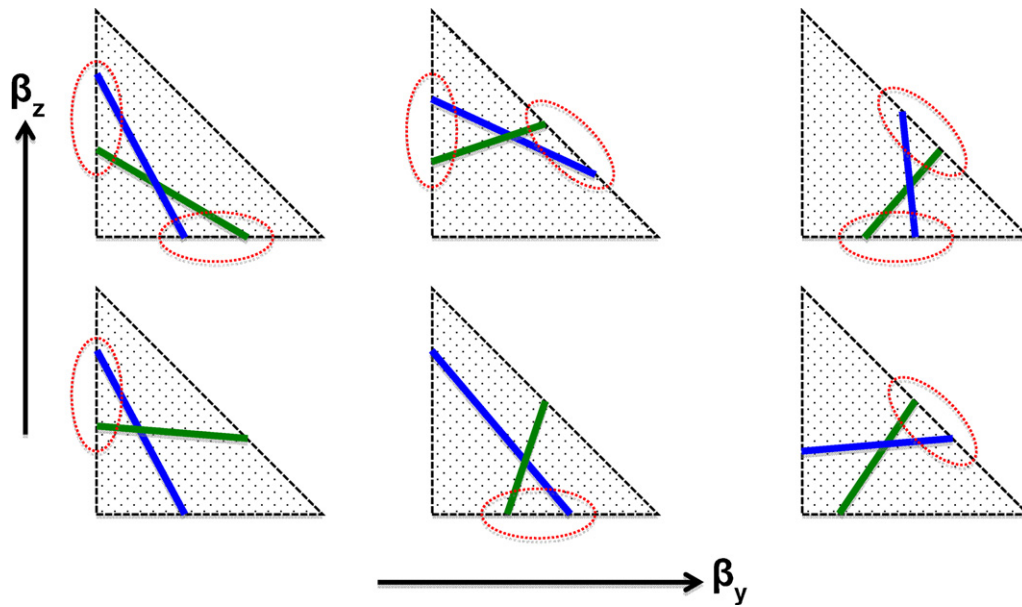


Fig. 6. Possibilities where a sub-triangle contains the solution if both RR_y and RR_z have different signs at the three vertices for one of the configurations of Type II in the improved 2D bisection method.

used in the minimization method, the efficiency of the minimization method can be further improved especially for the Maljamar reservoir oil.

4. Conclusions

One challenge in three-phase split calculation is the initialization of phase fractions through solving the deceptively simple-looking material balance, i.e., the Rachford–Rice equations. A number of authors have proposed various techniques for the solution of these two non-linear equations. In this work we compare the performance of the improved two-dimensional bisection method, the minimization method by Okuno et al. [10], and the direct Newton method. After systematic examination by using a large number of three-phase examples of different complexity, all the three methods demonstrate robustness and efficiency. Among them, the direct Newton method is the simplest and most efficient. It is based on a reliable two-phase stability testing that has been proposed in our recent work [2].

Acknowledgements

The authors are grateful to Kjetil Haugen, Joachim Moortgat, and Ryosuke Okuno for the fruitful discussions. This work was

financially supported by the Qatar National Research Fund under grant NPRP-29-6-7-30 and by the member companies of the Reservoir Engineering Research Institute (RERI). Their supports are greatly appreciated.

References

- [1] M.L. Michelsen, *Fluid Phase Equilibria* 9 (1982) 21–40.
- [2] Z.D. Li, A. Firoozabadi, *SPE J.*, in press.
- [3] H.H. Rachford, J.D. Rice, *Trans. Am. Inst. Min. Metall. Eng.* 195 (1952) 327–328.
- [4] K.B. Haugen, A. Firoozabadi, L.X. Sun, *AIChE J.* 57 (2011) 2555–2565.
- [5] P.A. Nelson, *Comput. Chem. Eng.* 11 (1987) 581–591.
- [6] A. Bunz, R. Dohrn, J. Prausnitz, *Comput. Chem. Eng.* 15 (1991) 47–51.
- [7] C.F. Leibovici, J. Neoschil, *Fluid Phase Equilibria* 112 (1995) 217–221.
- [8] M.L. Michelsen, *Comput. Chem. Eng.* 18 (1994) 545–550.
- [9] C.F. Leibovici, D.V. Nichita, *Fluid Phase Equilibria* 267 (2008) 127–132.
- [10] R. Okuno, R.T. Johns, K. Sepehrnoori, *SPE J.* 15 (2010) 313–325.
- [11] J.L. Shelton, L. Yarborough, *J. Petrol. Technol.* 29 (1977) 1171–1178.
- [12] F.M. Orr, A.D. Yu, C.L. Lien, *SPE J.* 21 (1981) 480–492.
- [13] S.A. Khan, G.A. Pope, K. Sepehrnoori, *SPE* 24130 (1992).
- [14] R. Okuno, R.T. Johns, K. Sepehrnoori, *SPE* 129846 (2010).
- [15] R. Okuno, R.T. Johns, K. Sepehrnoori, *SPE J.* 15 (2010) 689–703.
- [16] D.Y. Peng, D.B. Robinson, *Ind. Eng. Chem. Fund.* 15 (1976) 59–64.
- [17] D.B. Robinson, D.Y. Peng, S.Y.K. Chung, *Fluid Phase Equilibria* 24 (1985) 25–41.