Reliable Computation of High Pressure Phase Behavior

Gang Xu, Benito Stradi,
Robert W. Maier, Joan F. Brennecke and
Mark A. Stadtherr*

Department of Chemical Engineering
University of Notre Dame, Notre Dame, IN 46556

Paper Number 88c
AIChe Annual Meeting, Miami, FL
November 1998

*Author to whom all correspondence should be addressed; Phone: (219) 631-9318; Fax: (219) 631-8366; E-mail: markst@nd.edu
Summary

- In modeling high pressure phase behavior, computational problems due to multiple roots or multiple local optima are well known (e.g., convergence to trivial or incorrect roots; convergence to a local but not global optimum).

- Many clever techniques have been devised to alleviate such difficulties, but there has been no **general-purpose, model-independent**, and **completely reliable** method for solving all phase behavior problems.

- Interval analysis provides a **mathematically and computationally guaranteed** method for reliably solving phase behavior problems.

- This is demonstrated using example problems in phase stability and equilibrium and in computing critical points, and can be applied to a variety of other problems, such as computation of azeotropes.
Background—Interval Analysis

• A real interval $X = [a, b] = \{ x \in \mathbb{R} \mid a \leq x \leq b \}$ is a segment on the real number line and an interval vector $X = (X_1, X_2, \ldots, X_n)^T$ is an $n$-dimensional rectangle or “box”.

• Basic interval arithmetic for $X = [a, b]$ and $Y = [c, d]$ is $X \text{ op } Y = \{ x \text{ op } y \mid x \in X, y \in Y \}$ where $\text{op} \in \{+, -, \times, \div\}$. For example, $X + Y = [a + c, b + d]$.

• Computed endpoints are rounded out to guarantee the enclosure.

• Interval elementary functions (e.g. $\exp(X)$, $\log(X)$, etc.) are also available.

• The interval extension $F(X)$ encloses the range (all values) of $f(x)$ for $x \in X$.

• Interval extensions can be computed using interval arithmetic (the “natural” interval extension), or with other techniques.
Interval Approach

- **Interval Newton/Generalized Bisection (IN/GB)**
  - Given a system of equations to solve, an initial interval (bounds on all variables), and a solution tolerance
  - IN/GB can find (enclose) with mathematical and computational certainty either all solutions or determine that no solutions exist. (e.g., Kearfott 1987,1996; Neumaier 1990)

- A general purpose approach: requires no simplifying assumptions or problem reformulations

- Details of algorithm given by Schnepper and Stadtherr (1996)

- Implementation based on modifications of routines from INTBIS and INTLIB packages (Kearfott and coworkers)
Interval Approach (Cont’d)

Problem: Solve \( f(x) = 0 \) for all roots in interval \( X^{(0)} \).

Basic iteration scheme: For a particular subinterval (box), \( X^{(k)} \), perform root inclusion test:

- Compute the interval extension (range) of each function in the system.
- If 0 is not an element of each range, delete the box.
- If 0 is an element of each range, then compute the image, \( N^{(k)} \), of the box by solving

\[
F'(X^{(k)})(N^{(k)} - x^{(k)}) = -f(x^{(k)})
\]

- \( x^{(k)} \) is some point in the interior of \( X^{(k)} \).
- \( F'(X^{(k)}) \) is an interval extension of the Jacobian of \( f(x) \) over the box \( X^{(k)} \).
There was no solution in $X^{(k)}$
Unique solution in $X^{(k)}$

This solution is in $N^{(k)}$

Point Newton method will converge to it
Any solutions in $X^{(k)}$ are in intersection of $X^{(k)}$ and $N^{(k)}$.

If intersection is sufficiently small, repeat root inclusion test; otherwise bisect the result of the intersection and apply root inclusion test to each resulting subinterval.
Example–Phase Stability Analysis

Gibbs energy formulation $T$ and $P$ constant:

$$\min_{x, v} \quad D(x, v) = \hat{G} - \hat{G}_z - \sum_{i=1}^{n} \left( \frac{\partial \hat{G}}{\partial x_i} \right)_z (x_i - z_i)$$

subject to

$$1 - \sum_{i=1}^{n} x_i = 0$$

$$EOS(x, v) = 0$$

- Trivial local optimum (minimum or maximum) at the feed composition $x = z$; may be multiple nontrivial optima. Need technique guaranteed to find the global minimum.

- If global minimum of $D$ is negative, mixture may split (unstable or metastable feed).
Example–Phase Stability Analysis (cont.)

• Stationary points of the optimization problem can be found by solving the nonlinear equation system

\[
\left[ \left( \frac{\partial \hat{G}}{\partial x_i} \right) - \left( \frac{\partial \hat{G}}{\partial x_n} \right) \right] - \left[ \left( \frac{\partial \hat{G}}{\partial x_i} \right) - \left( \frac{\partial \hat{G}}{\partial x_n} \right) \right]_z = 0,
\]

\[i = 1, \ldots, n - 1\]

\[1 - \sum_{i=1}^{n} x_i = 0\]

\[EOS(x, v) = 0\]

• Trivial root at the feed composition \(x = z\); may be multiple nontrivial roots. Need technique guaranteed to find all the roots.
Example–Phase Stability Analysis (cont.)

Alternative formulation (Nagarajan et al., 1991) for constant $T$ and $P$ based on Helmholtz energy density $E(d)$, where $d$ is the mole density vector:

$$
\min_d D(d) = E - E_z - \sum_{i=1}^n \left( \frac{\partial E}{\partial d_i} \right)_z (d_i - d_{z,i})
$$

- Unconstrained optimization with one less variable.

- Attractive for dealing with new equations of state (e.g. SAFT) which are presented naturally in the form $E(d)$.

- Again may have multiple local minima.
Example–Phase Stability Analysis (cont.)

- Equation system for stationary points in Helmholtz energy density formulation:

\[
\left( \frac{\partial E}{\partial d_i} \right) - \left( \frac{\partial E}{\partial d_i} \right)_z = 0
\]

\[i = 1, \ldots, n\]

- Again trivial root at feed, and may have multiple solutions.

- Interval approach has been applied previously to the Gibbs energy formulation (Hua et al., 1996,1998), but not to the Helmholtz energy formulation.
Problem 1

$N_2$ (1), $C_2H_6$ (2), $T = 270$ K, $P = 76$ bar, PR EOS model

<table>
<thead>
<tr>
<th>Feed $(z_1, z_2)$</th>
<th>Number of Stationary Points</th>
<th>Stable</th>
<th>CPU time (sec.)</th>
<th>Gibbs</th>
<th>Helmholtz</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.18, 0.82)</td>
<td>3</td>
<td>No</td>
<td>0.20</td>
<td>0.09</td>
<td></td>
</tr>
<tr>
<td>(0.44, 0.56)</td>
<td>3</td>
<td>No</td>
<td>0.20</td>
<td>0.09</td>
<td></td>
</tr>
<tr>
<td>(0.60, 0.40)</td>
<td>1</td>
<td>Yes</td>
<td>0.13</td>
<td>0.06</td>
<td></td>
</tr>
</tbody>
</table>

- CPU time on Sun Ultra 30.
- Both formulations give same result for stability analysis
Problem 2

\[ \text{N}_2 \ (1), \ \text{CH}_4 \ (2), \ \text{C}_2\text{H}_6 \ (3) \ T = 270 \ \text{K}, \ P = 76 \ \text{bar}, \ \text{PR EOS model} \]

<table>
<thead>
<tr>
<th>Feed [(z_1, z_2, z_3)]</th>
<th>Number of\nStationary Points</th>
<th>Stable [?]</th>
<th>CPU time (sec.)</th>
<th>Gibbs</th>
<th>Helmholtz</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.30, 0.10, 0.60)</td>
<td>3</td>
<td>No</td>
<td>1.34</td>
<td>1.19</td>
<td></td>
</tr>
<tr>
<td>(0.15, 0.30, 0.55)</td>
<td>3</td>
<td>No</td>
<td>3.37</td>
<td>2.12</td>
<td></td>
</tr>
<tr>
<td>(0.08, 0.38, 0.54)</td>
<td>1</td>
<td>Yes</td>
<td>2.54</td>
<td>1.43</td>
<td></td>
</tr>
<tr>
<td>(0.05, 0.05, 0.90)</td>
<td>1</td>
<td>Yes</td>
<td>0.54</td>
<td>0.60</td>
<td></td>
</tr>
</tbody>
</table>

- CPU time on Sun Ultra 30.

- Both formulations give same result for stability analysis
Example—Alternative Mixing Rules

• Previous applications of the interval method to phase stability and equilibrium problems used standard mixing rules — quadratic for $a$, linear for $b$.

• Can the interval approach be used in connection with more complex mixing rules?

• In the example, the interval method is used to compute phase equilibrium when the Wong-Sandler mixing rules are used.
Wong-Sandler Mixing Rules

\[ a = \frac{RTQ_{ws}D_{ws}}{1 - D_{ws}} = RTD_{ws}b \]

\[ b = \frac{Q_{ws}}{1 - D_{ws}} \]

where

\[ D_{ws} = \frac{A_{\infty}^E}{cRT} + \sum_{i=1}^{n} \frac{x_i a_{ii}}{RT b_i} \]

\[ Q_{ws} = \sum_{i=1}^{n} \sum_{j=1}^{n} x_i x_j \left( \frac{b_i + b_j}{2} - \frac{\sqrt{a_{ii} a_{jj}}}{RT} (1 - k_{ij}) \right) \]

and

\[ \frac{A_{\infty}^E}{RT} = \sum_{i} x_i \left( \frac{\sum_j x_j \tau_{ji} g_{ji}}{\sum_k x_k g_{ki}} \right) \]

(NRTL equation)
**Problem 3**

$\text{CO}_2 \ (1), \ \text{H}_2\text{O} \ (2), \ T = 550 \ \text{K}, \ \text{Feed: } z_1 = 0.2, \ z_2 = 0.8, \ \text{PRSV EOS with Wong-Sandler mixing rules}$

<table>
<thead>
<tr>
<th>Pressure (bar)</th>
<th>Phase Equilibrium</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Split Fraction</td>
<td>$(x_1, x_2, v \ [\text{cm}^3/\text{mol}])$</td>
</tr>
<tr>
<td>200</td>
<td>0.29</td>
<td>(0.619, 0.381, 182)</td>
</tr>
<tr>
<td></td>
<td>0.71</td>
<td>(0.029, 0.971, 28.9)</td>
</tr>
<tr>
<td>3000</td>
<td>No Phase Split</td>
<td></td>
</tr>
<tr>
<td>8000</td>
<td>0.28</td>
<td>(0.320, 0.680, 22.3)</td>
</tr>
<tr>
<td></td>
<td>0.72</td>
<td>(0.153, 0.847, 22.2)</td>
</tr>
<tr>
<td>25000</td>
<td>0.55</td>
<td>(0.037, 0.963, 20.3)</td>
</tr>
<tr>
<td></td>
<td>0.45</td>
<td>(0.395, 0.605, 20.2)</td>
</tr>
</tbody>
</table>

CPU time on Sun Ultra 30.
Problem 3

\[ \text{CO}_2 \ (1), \ \text{H}_2\text{O} \ (2), \ T = 550 \ \text{K}, \ \text{Feed: } z_1 = 0.2, \ z_2 = 0.8, \ \text{PRSV EOS with Wong-Sandler mixing rules} \]
Example—Computing Critical Points

- Formulation as system of nonlinear equations
  - Determinants
  - Method of Heidemann and Khalil (1979)

- Nonlinear equation system to be solved for critical points has unknown number of roots

- Interval method provides an approach guaranteed to find all roots, or to determine with certainty that there are none.

- Example problem computes critical point(s) for mixture of CH₄ and H₂S at various compositions

- Temperature range searched is 110–800 K; Volume range searched is 1.1b – 4.0b.
### Problem 4

**CH₄ (1), H₂S (2), SRK EOS**

<table>
<thead>
<tr>
<th>Feed ((z_1, z_2))</th>
<th>Critical Point(s)</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(V_c \text{ [cm}^3/\text{mol}])</td>
<td>(T_c \text{ [K]})</td>
</tr>
<tr>
<td>((0.97, 0.03))</td>
<td>107.70</td>
<td>196.74</td>
</tr>
<tr>
<td>((0.93, 0.07))</td>
<td>97.75</td>
<td>204.78</td>
</tr>
<tr>
<td></td>
<td>44.72</td>
<td>114.77</td>
</tr>
<tr>
<td>((0.85, 0.15))</td>
<td>74.03</td>
<td>212.99</td>
</tr>
<tr>
<td></td>
<td>59.41</td>
<td>190.98</td>
</tr>
</tbody>
</table>

CPU time on Sun Ultra 30.
Problem 4 (Cont.)

CH₄ (1), H₂S (2), SRK EOS

<table>
<thead>
<tr>
<th>Feed (z₁, z₂)</th>
<th>Critical Point(s)</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.53, 0.47)</td>
<td>no critical point</td>
<td>17.35</td>
</tr>
<tr>
<td>(0.52, 0.48)</td>
<td>59.26</td>
<td>30.02</td>
</tr>
<tr>
<td></td>
<td>54.93</td>
<td></td>
</tr>
<tr>
<td></td>
<td>270.02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>146.07</td>
<td></td>
</tr>
<tr>
<td></td>
<td>260.27</td>
<td></td>
</tr>
<tr>
<td></td>
<td>149.00</td>
<td></td>
</tr>
<tr>
<td>(0.51, 0.49)</td>
<td>63.37</td>
<td>25.60</td>
</tr>
<tr>
<td></td>
<td>50.31</td>
<td></td>
</tr>
<tr>
<td></td>
<td>249.01</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160.10</td>
<td></td>
</tr>
</tbody>
</table>

CPU time on Sun Ultra 30.
Concluding Remarks

Interval analysis is a general-purpose and model-independent approach for solving phase behavior problems, providing a mathematical and computational guarantee of reliability.

- Phase stability and equilibrium
  - Gibbs energy formulation (e.g., Hua et al., 1998)
  - Helmholtz energy density formulation
  - Any EOS and mixing rule
  - Or any activity coefficient model (e.g., Stadtherr et al., 1995)

- Critical Points
  - Any EOS and mixing rule

- Azeotropes
  - Homogeneous (Maier et al., 1998a)
  - Heterogeneous
  - Reactive (Maier et al., 1998b)
  - From any EOS or activity coefficient model
Concluding Remarks (Cont.)

- Interval analysis provides powerful problem solving techniques with many other applications in the modeling of thermodynamics and phase behavior and in other process modeling problems.

- Continuing advances in computing hardware and software (e.g., compiler support for interval arithmetic) will make this approach even more attractive.
• Acknowledgments

  – ACS PRF 30421-AC9
  – NSF CTS95-22835
  – NSF DMI96-96110
  – NSF EEC97-00537-CRCD
  – EPA R824731-01-0
  – DOE DE-FG07-96ER14691
  – Sun Microsystems, Inc.

• For more information:

  – Contact Prof. Stadtherr at markst@nd.edu
  – See also
    http://www.nd.edu/~markst