Reliable Process Modeling Using Interval Analysis

Mark A. Stadtherr
Department of Chemical Engineering
University of Notre Dame
Notre Dame, IN 46556

Carnegie Mellon University
February 1998
Outline

• Motivation

• Background

• Methodology

• Examples
  – Phase Stability Analysis
  – Phase Equilibrium
  – Computation of Azeotropes
  – Other
Computing Technology

- Faster and faster single processor performance

- High performance computing (HPC)
  - Many forms of scalar multiprocessing
  - Vector multiprocessing
  - Metacomputing

- Capability to
  - Solve problems faster
  - Solve larger problems
  - Solve more complex problems
  - Solve problems more reliably
Solving Problems More Reliably

- Global vs. local optimization
- Finding multiple solutions
- Existence and uniqueness of solutions
- Feasibility of NLPs
- Floating point arithmetic problems (e.g., rounding errors)
Common Misconceptions

• Dennis and Schnabel (1983)
  “In general, the questions of existence and uniqueness—does a given problem have a solution and is it unique?—are beyond the capabilities one can expect of algorithms that solve nonlinear problems”

• Heath (1997)
  “It is not possible, in general, to guarantee convergence to the correct solution or to bracket the solution to produce an absolutely safe method” [for solving nonlinear equations]
Solving Problems More Reliably

- In fact there do exist methods, based on **interval analysis**, that, within given initial bounds on each variable, can:
  - Find (enclose) any and all solutions to a nonlinear equation system to a desired tolerance
  - Determine that there is no solution of a nonlinear equation system
  - Find the global optimum of a nonlinear objective function

- These methods:
  - Provide a mathematical guarantee of reliability
  - Deal automatically with rounding error, and so also provide a **computational** guarantee of reliability
Rounding Error

- Rump’s (1988) problem

\[
    f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/2y
\]

- Evaluate \( f(x, y) \) for \( x = 77617 \) and \( y = 33096 \).

- All inputs are machine numbers (representable exactly in floating point arithmetic), so only rounding errors occur during function evaluation.
Rounding Error

- Evaluation on an IBM S/370 using a FORTRAN program

- Single precision

  \[ f = 1.172603 \ldots \]

- Double precision

  \[ f = 1.1726039400531 \ldots \]

- Extended precision

  \[ f = 1.172603940053178 \ldots \]

- The correct answer is

  \[ f = -0.827396059946 \ldots \]
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.

- 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 $X \text{ op } Y = \{x \text{ op } y \mid x \in X, \ y \in Y\}$

  
  $X + Y = [a + c, b + d]$

  $X - Y = [a - d, b - c]$

  $X \times Y = [\text{min}(ac, ad, bc, bd), \text{max}(ac, ad, bc, bd)]$

  $X \div Y = [a, b] \times [1/d, 1/c], \ 0 \notin Y$

- For $X \div Y$ when $0 \in Y$, an extended interval arithmetic is available.

- Computed endpoints are rounded out to guarantee the enclosure.
Interval Analysis (continued)

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

- The interval extension \(F(X)\) encloses all values of \(f(x)\) for \(x \in X\). That is, \(F(X) \supseteq \{f(x) \mid x \in X\}\).

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

- If a variable occurs more than once in an expression, the natural interval extension may not tightly bound the true range.
Interval Analysis (continued)

- Example: \( f(x) = x/(x - 1) \) evaluated for the interval \( X = [2, 3] \)

- The natural interval extension is

\[
F([2, 3]) = [2, 3]/([2, 3] - 1) = [2, 3]/[1, 2] = [1, 3]
\]

- Rearranged \( f(x) = x/(x - 1) = 1 + 1/(x - 1) \), the natural interval extension is

\[
F([2, 3]) = 1 + 1/([2, 3] - 1) = 1 + 1/[1, 2] = 1 + [0.5, 1] = [1.5, 2]
\]

which is the true range.

- This is the “dependency” problem. In the first case, each occurrence of \( x \) was treated as a independent interval in performing interval arithmetic.
Interval Newton Method

• For a system of nonlinear equations \( f(x) = 0 \), find (enclose) all roots in a given initial interval \( X^{(0)} \) or determine that there are none.

• At iteration \( k \), given the interval \( X^{(k)} \), if \( 0 \in F(X^{(k)}) \) solve the linear interval equation system

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

for the "image" \( N^{(k)} \), where \( F'(X^{(k)}) \) is an interval extension of the Jacobian of \( f(x) \) over the current interval \( X^{(k)} \), and \( x^{(k)} \) is a point inside \( X^{(k)} \).

• Any root \( x^* \in X^{(k)} \) is also contained in the image \( N^{(k)} \), suggesting the iteration scheme \( X^{(k+1)} = X^{(k)} \cap N^{(k)} \) (Moore, 1966).

• It follows that if \( X^{(k)} \cap N^{(k)} = \emptyset \), then there is no root in \( X^{(k)} \). This is also the conclusion if \( 0 \not\in F(X^{(k)}) \).
Interval Newton Method (continued)

- Interval Newton provides an existence and uniqueness test: If \( N^{(k)} \subset X^{(k)} \), then:
  - There is a **unique** zero of \( f(x) \) in \( X^{(k)} \).
  - The interval Newton iteration \( X^{(k+1)} = X^{(k)} \cap N^{(k)} \) will converge quadratically to a tight enclosure of the root.
  - The point Newton method will converge quadratically to the root starting from any point in \( X^{(k)} \).

- If a unique root cannot be confirmed (\( N^{(k)} \subset X^{(k)} \)) or ruled out (\( X^{(k)} \cap N^{(k)} = \emptyset \)), then either:
  - Continue with the next iterate \( X^{(k+1)} \) if it is sufficiently smaller than \( N^{(k)} \)
  - **Bisect** \( X^{(k+1)} \) and perform interval Newton on the resulting intervals

This is the interval Newton/generalized bisection (IN/GB) approach.
Any solutions in $X^{(k)}$ are in intersection of $X^{(k)}$ and $N^{(k)}$
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
Interval Newton Method (continued)

- For \( f(x) = 0 \), this method can enclose with mathematical and computational certainty any and all solutions in a given initial interval, or can determine that there are none.

- A preconditioned interval Gauss-Seidel-like technique is often used to solve for the image \( N^{(k)} \) (Hansen and coworkers).

- Our implementation is based on modifications of routines taken from the packages INTBIS and INTLIB (Kearfott and coworkers).

- The interval Newton procedure can be performed on multiple intervals independently and in parallel.

- IN/GB was first implemented for process modeling problems by Schnepper and Stadtherr (1990).


Phase Stability Problem

- Will a mixture (feed) at a given $T$, $P$, and composition $x$ split into multiple phases?

- A key subproblem in determination of phase equilibrium, and thus in the design and analysis of separation operations.

- Using tangent plane analysis, can be formulated as a minimization problem, or as an equivalent nonlinear equation solving problem.

- Equation system to be solved may have trivial and/or multiple roots (optimization problem has multiple local optima).

- Conventional techniques may fail to converge, or converge to false or trivial solutions.
Tangent Plane Analysis

- A phase at $T$, $P$, and feed composition $z$ is unstable if the Gibbs energy of mixing vs. composition surface

\[ m(x, v) = \Delta g_{mix} = \Delta \hat{G}_{mix}/RT \]

ever falls below a plane tangent to the surface at $z$

\[ m_{tan}(x) = m(z, v_z) + \sum_{i=1}^{n} \left( \frac{\partial m}{\partial x_i} \right) \bigg|_z (x_i - z_i) \]

- That is, if the tangent plane distance

\[ D(x, v) = m(x, v) - m_{tan}(x) \]

is negative for any composition $x$, the phase is unstable.

- In this context, “unstable” refers to both the metastable and classically unstable cases.
Example 0

\( n \)-Butyl Acetate—Water, NRTL Model

Gibbs energy of mixing \( m \) vs. \( x_1 \)
Example 0 (continued)

Feed composition $z_1 = 0.95$

Phase of this composition is stable ($D$ is never negative).
Example 0 (continued)

Feed composition $x_1 = 0.62$

Phase of this composition is unstable and can split ($D$ becomes negative).
Optimization Formulation

- To determine if $D$ ever becomes negative, determine the minimum of $D$ and examine its sign

$$\min_{x,v} D(x, v)$$

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.
Equation Solving Formulation

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

\[
\left[ \left( \frac{\partial m}{\partial x_i} \right) - \left( \frac{\partial m}{\partial x_n} \right) \right] - \left[ \left( \frac{\partial m}{\partial x_i} \right) - \left( \frac{\partial m}{\partial x_n} \right) \right]_{z} = 0,
\]

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

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

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

- Trivial root at the feed composition \(x = z\); may be multiple nontrivial roots. Need technique guaranteed to find all the roots.
Example 1

CH₄, H₂S, \( T = 190 \) K, \( P = 40 \) atm, \( z_1 = 0.0187 \), SRK model. Tangent plane distance \( D \) vs. \( x_1 \)

- Five stationary points (four minima, one maximum).
- Standard local methods (e.g. Michelsen, 1982) known to fail (predict stability when system is actually unstable).
Example 1 (continued)

CH$_4$, H$_2$S, $T = 190$ K, $P = 40$ atm, $z_1 = 0.0187$, SRK model. Tangent plane distance $D$ vs. $x_1$ (region near origin)
Some Current Solution Methods

- Various local methods — Fast, but initialization dependent (may use multiple initial guesses), and not always reliable

- Some more reliable approaches
  - Exhaustive search on grid (Eubank et al., 1992)
  - Homotopy-continuation (Sun and Seider, 1995)
  - Topological degree (Wasylkiewicz et al., 1996)
  - Branch and bound (McDonald and Floudas, 1995, 1997): Guarantee of global optimum when certain activity coefficient models are used

- Interval analysis
  - Provides a general-purpose, model-independent method for solving phase stability problem with complete certainty.
Interval Analysis for Phase Stability

- Initial interval includes all physically feasible values of mole fraction and molar volume

- To reduce overestimation in interval extensions due to dependency problem:
  - Identify and use function monotonicity
  - Let monotonicity information be inherited when an interval is bisected
  - Use special properties of mole fraction weighted averages

- “Standard” mixing rules used
  - Quadratic for $a$
  - Linear for $b$
Example 1 — Phase Stability

CH₄, H₂S, \( T = 190 \text{ K} \), \( P = 40 \text{ atm} \), \( z_1 = 0.0187 \), SRK model

<table>
<thead>
<tr>
<th>Feed ((z_1, z_2)) and CPU time</th>
<th>Stationary Points (roots) ((x_1, x_2, v \text{ [cm}^3/\text{mol]}))</th>
<th>(D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.0187, 0.9813) 0.20 sec</td>
<td>(0.885, 0.115, 36.6)</td>
<td>0.011</td>
</tr>
<tr>
<td></td>
<td>(0.0187, 0.9813, 207.3)</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>(0.031, 0.969, 115.4)</td>
<td>0.008</td>
</tr>
<tr>
<td></td>
<td>(0.077, 0.923, 64.1)</td>
<td>-0.004</td>
</tr>
<tr>
<td></td>
<td>(0.491, 0.509, 41.5)</td>
<td>0.073</td>
</tr>
</tbody>
</table>

- CPU time on Sun Ultra 2/1300.
- All stationary points easily found, showing the feed to be unstable.
- Presence of multiple real volume roots causes no difficulties.
Example 2 — Phase Stability

\( \text{CO}_2, \text{CH}_4, T = 220 \text{ K}, P = 60.8 \text{ bar}, \text{PR model} \)

<table>
<thead>
<tr>
<th>Feed ((z_1, z_2))</th>
<th>Number of Stationary Points</th>
<th>(D_{min})</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.10, 0.90)</td>
<td>1</td>
<td>0.0</td>
<td>0.11</td>
</tr>
<tr>
<td>(0.20, 0.80)</td>
<td>3</td>
<td>-0.007</td>
<td>0.33</td>
</tr>
<tr>
<td>(0.30, 0.70)</td>
<td>3</td>
<td>-0.0002</td>
<td>0.36</td>
</tr>
<tr>
<td>(0.43, 0.57)</td>
<td>3</td>
<td>-0.001</td>
<td>0.35</td>
</tr>
<tr>
<td>(0.60, 0.40)</td>
<td>1</td>
<td>0.0</td>
<td>0.29</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300.
Example 3 — Phase Stability

Green et al. (1993) ternary, $T = 400$ K, $P = 80$ atm, VDW model

<table>
<thead>
<tr>
<th>Feed ($z_1, z_2, z_3$)</th>
<th>Number of Stationary Points</th>
<th>$D_{min}$</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.83, 0.085, 0.085)</td>
<td>3</td>
<td>-0.0099</td>
<td>0.70</td>
</tr>
<tr>
<td>(0.77, 0.115, 0.115)</td>
<td>3</td>
<td>-0.0036</td>
<td>0.76</td>
</tr>
<tr>
<td>(0.72, 0.14, 0.14)</td>
<td>3</td>
<td>-0.0036</td>
<td>0.83</td>
</tr>
<tr>
<td>(0.69, 0.155, 0.155)</td>
<td>3</td>
<td>0.0</td>
<td>0.85</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300.
Example 4 — Phase Stability

$N_2$, $CH_4$, $C_2H_6$, $T = 270$ K, $P = 76$ bar, PR model

<table>
<thead>
<tr>
<th>Feed ($z_1, z_2, z_3$)</th>
<th>Number of Stationary Points</th>
<th>$D_{min}$</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.30,0.10,0.60)</td>
<td>3</td>
<td>-0.015</td>
<td>1.3</td>
</tr>
<tr>
<td>(0.15,0.30,0.55)</td>
<td>3</td>
<td>-0.001</td>
<td>3.4</td>
</tr>
<tr>
<td>(0.08,0.38,0.54)</td>
<td>1</td>
<td>0.0</td>
<td>2.5</td>
</tr>
<tr>
<td>(0.05,0.05,0.90)</td>
<td>1</td>
<td>0.0</td>
<td>0.54</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300.
Example 5 — Phase Stability

\[ \text{CH}_4, \text{CO}_2, \text{H}_2\text{S}, \text{H}_2\text{O}, \text{PR model} \]

<table>
<thead>
<tr>
<th>Feed</th>
<th>Number of Stationary Points</th>
<th>(D_{\text{min}})</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
<td>-0.027</td>
<td>60.4</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>-1.201</td>
<td>9.8</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>-0.295</td>
<td>10.2</td>
</tr>
<tr>
<td>D</td>
<td>3</td>
<td>-0.027</td>
<td>129.2</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300.

It is not really necessary to find all the stationary points; only need to find the global minimum.
Finding the global minimum

• Requires evaluation of an interval extension of the objective function $D$. This extra expense does not pay off on small problems.

• There is a known upper bound of zero (the tangent point) on the global minimum of $D$.

• If interval extension of $D$ has positive lower bound over some interval, that interval cannot contain the global minimum and can be discarded.

• If interval extension of $D$ has negative upper bound over some interval, global minimum will be negative, proving instability

• This is a special form of interval branch and bound combined with interval Newton

• For feed D in Problem 5, CPU time reduced from 129.2 sec to 2.9 sec.
Incorporating Local Techniques

- If a local method indicates instability then this is the correct answer as it means a point at which $D < 0$ has been found.

- If a local method indicates stability, however, this may not be the correct answer since the local method may have missed the global minimum in $D$.

- Combined local/global approach:
  - Use local methods to try to demonstrate instability.
  - If instability not found, only then use global interval method to confirm stability or identify instability.
Approach Used

- If \( m(z, v_z) > 0 \) \( \Rightarrow \) unstable

- Evaluate \( D \) at pure components. If any \( D < 0 \) \( \Rightarrow \) unstable

- For a number of randomly chosen compositions:
  - If \( D < 0 \) \( \Rightarrow \) unstable
  - If \( D \geq 0 \), then start a local solver (Newton) and try to converge to a stationary point. If at termination \( D < 0 \) \( \Rightarrow \) unstable

- If still not shown unstable, then apply interval approach to confirm stability or find instability missed by local techniques.

- This approach is implemented in the code INTSTAB (Hua et al., 1997)
Effect of Local Approach

- Typical results comparing combined local/global approach with global only approach

<table>
<thead>
<tr>
<th>Example Problem</th>
<th>Stable?</th>
<th>CPU time (sec)</th>
<th>Global</th>
<th>Local/Global</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>N</td>
<td>0.20</td>
<td>0.002</td>
<td></td>
</tr>
<tr>
<td>3 (feed 1)</td>
<td>N</td>
<td>0.70</td>
<td>0.001</td>
<td></td>
</tr>
<tr>
<td>3 (feed 4)</td>
<td>Y</td>
<td>0.85</td>
<td>0.88</td>
<td></td>
</tr>
<tr>
<td>4 (feed 1)</td>
<td>N</td>
<td>1.30</td>
<td>0.002</td>
<td></td>
</tr>
<tr>
<td>4 (feed 4)</td>
<td>Y</td>
<td>0.54</td>
<td>0.58</td>
<td></td>
</tr>
</tbody>
</table>

- CPU times on Sun Ultra 2/1300 using INTSTAB.

- For unstable mixtures, instability generally detected in milliseconds.

- For stable mixtures, negligible increase in computation time.
Phase Equilibrium Problem

- Can formulate as global minimization of total Gibbs energy, subject to material balance constraints. May have multiple local minima.

- Can also formulate as equation solving problem: equifugacity equations and material balances. May have multiple solutions.

- Need to seek global solution, but local methods can be applied since phase stability analysis can be used as a global optimality test that can be applied to any local solution (Baker et al., 1982).

- Correct solution of the phase stability problem is thus the key to correct solution of the phase equilibrium problem.

- Interval analysis guarantees correct solution of the phase stability problem, and so can also guarantee correct solution of the phase equilibrium (split) problem.
Global Solution of Phase Equilibrium Problem

- Can combine the global stability analysis with any standard phase split (or flash) algorithm.

- One approach
  - Perform global stability analysis. If unstable, use the local minima in $D$ to generate initial guesses for the solution to the phase split problem.
  - For each such initial guess, use a local optimizer (SQP) to solve the phase split problem and then test for stability.
  - If global solution not found increase number of phases and continue.

- This approach is implemented in the code INTFLASH (Hua et al., 1997)
Example 6 — Phase Equilibrium

\( \text{CH}_4, \text{CO}_2, \text{H}_2\text{S}, T = 282.15 \, \text{K}, \, P = 59.5 \, \text{bar}, \, \text{PR model}, \, z_1 = 0.4995, \, z_2 = 0.0977, \, z_3 = 0.4028 \)

<table>
<thead>
<tr>
<th>Phase I (L)</th>
<th>( \beta^I )</th>
<th>0.1748</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( v^I )</td>
<td>41.95 , \text{cm}^3/\text{mol}</td>
</tr>
<tr>
<td></td>
<td>( x^I )</td>
<td>(0.1047, 0.0727, 0.8226)</td>
</tr>
<tr>
<td>Phase II (V)</td>
<td>( \beta^{II} )</td>
<td>0.8352</td>
</tr>
<tr>
<td></td>
<td>( v^{II} )</td>
<td>280.1 , \text{cm}^3/\text{mol}</td>
</tr>
<tr>
<td></td>
<td>( x^{II} )</td>
<td>(0.5832, 0.1030, 0.3138)</td>
</tr>
<tr>
<td>CPU</td>
<td></td>
<td>2.05 , \text{sec}</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300 using INTFLASH.
Example 7 — Phase Equilibrium

\[ \text{CH}_4, \text{CO}_2, \text{H}_2\text{S}, T = 208 \text{ K}, P = 54.9 \text{ bar}, \text{PR model}, \]
\[ z_1 = 0.4989, z_2 = 0.0988, z_3 = 0.4023 \]

<table>
<thead>
<tr>
<th>Phase I (V)</th>
<th>( \beta^I )</th>
<th>0.0702</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( v^I )</td>
<td>141.9 cm(^3)/mol</td>
</tr>
<tr>
<td></td>
<td>( x^I )</td>
<td>(0.9120, 0.0417, 0.0463)</td>
</tr>
<tr>
<td>Phase II (L)</td>
<td>( \beta^{II} )</td>
<td>0.3816</td>
</tr>
<tr>
<td></td>
<td>( v^{II} )</td>
<td>53.46 cm(^3)/mol</td>
</tr>
<tr>
<td></td>
<td>( x^{II} )</td>
<td>(0.7539, 0.0848, 0.1613)</td>
</tr>
<tr>
<td>Phase III (L)</td>
<td>( \beta^{III} )</td>
<td>0.5482</td>
</tr>
<tr>
<td></td>
<td>( v^{III} )</td>
<td>35.69 cm(^3)/mol</td>
</tr>
<tr>
<td></td>
<td>( x^{III} )</td>
<td>(0.2685, 0.1158, 0.6157)</td>
</tr>
<tr>
<td>CPU</td>
<td></td>
<td>9.0 sec</td>
</tr>
</tbody>
</table>

CPU times on Sun Ultra 2/1300 using INTFLASH.
Computing Homogeneous Azeotropes

• Why
  – Identify limitations in separation operations
  – Construction of residue curve maps for design and synthesis of separation operations
  – Evaluation of thermodynamic models

• How
  – Solve system(s) of nonlinear equations derived from equifugacity condition
  – These equation system(s) often have multiple and/or trivial roots, or may have no solutions
Formulation: Simultaneous Approach

\[ x_i \left( \ln P - \ln P_i^{sat} - \ln \gamma_i \right) = 0, \ i \in C \]

\[ 1 - \sum_{i \in C} x_i = 0 \]

- \( C \) is the set of all \( N \) components
- Ideal vapor phase
- \( P_i^{sat} \) and \( \gamma_i \) are functions of \( T \)
- All \( k \)-ary azeotropes (\( k \leq N \)) are solutions, as are all of the pure components (trivial roots)
- Need solution method guaranteed to find all solutions
Formulation: Sequential Approach

- If $x_i \neq 0$

$$\ln P - \ln P_i^{sat} - \ln \gamma_i = 0, \ i \in C_{nz}$$

$$1 - \sum_{i \in C_{nz}} x_i = 0$$

- $C_{nz}$ is a set of $k$ nonzero components

- All $k$-ary azeotropes ($k \leq N$) for the chosen $C_{nz}$ are solutions; there may be no solutions

- Solve (unordered) sequence of problems:

  For $k = 2 \rightarrow N$:

  For all combinations of $k$ nonzero components, solve for all $k$-ary azeotropes

- Need solution method guaranteed to find all solutions of all problems, and to determine with certainty when there are no solutions
Formulation: Other Issues

- $T$ dependence of $\gamma_i$
  - Treat explicitly using T-dependent parameters in $\gamma_i$ model
  - Guess a reference temperature $T_{ref}$ and treat T-dependent parameters as constants evaluated at $T_{ref}$ $\Rightarrow$ No guarantee all azeotropes will be found, even if equations solved correctly

- Solutions of equifugacity equations may not be stable phases (liquid may split)
  - Need to check stability of liquid phase at azeotropic composition and temperature
  - Interval analysis also provides guaranteed method to determine stability
Some Current Solution Methods

• Various local methods — Fast, but initialization dependent and hard to find all roots

• Fidkowski et al. (1993) use a homotopy-continuation method
  – Simultaneous approach with explicit $T$-dependence of $\gamma_i$
  – Improved reliability but no guarantee that all roots are found

• Harding et al. (1997) use a branch and bound method
  – Simultaneous and sequential approaches, but $T_{ref}$ approach for $T$-dependence of $\gamma_i$
  – Reformulation as a global optimization problem using convex underestimating functions
  – Mathematical guarantee that all roots are found
Example Problems

- Solved using both simultaneous and sequential approaches (with same results for azeotropic composition and temperature)

- Solved for case of T-dependent $\gamma_i$-model parameters and for the case of constant $\gamma_i$-model parameters

- Problem A1
  - Ethanol, Methyl Ethyl Ketone, Water
  - 1 atm, 10-100 °C, Wilson Equation

- Problem A2
  - Acetone, Chloroform, Methanol
  - 16.8 atm, 100-200 °C, NRTL Equation

- Problem A3
  - Acetone, Methyl Acetate, Methanol
  - 1 atm, 10-100 °C, Wilson Equation

- Have solved many other problems using Wilson, NRTL and UNIQUAC activity coefficient models with up to $N=5$
## Results - Problem A1

### Azeotropes

<table>
<thead>
<tr>
<th>$\gamma_i$ parameters</th>
<th>E</th>
<th>MEK</th>
<th>W</th>
<th>T (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant ($T_{ref}=73.7$ °C)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.49</td>
<td>0.51</td>
<td>0.00</td>
<td>74.1</td>
<td></td>
</tr>
<tr>
<td>0.90</td>
<td>0.00</td>
<td>0.10</td>
<td>78.1</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>0.68</td>
<td>0.32</td>
<td>73.7</td>
<td></td>
</tr>
<tr>
<td>0.23</td>
<td>0.54</td>
<td>0.23</td>
<td>72.8</td>
<td></td>
</tr>
<tr>
<td>T-dependent</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.49</td>
<td>0.51</td>
<td>0.00</td>
<td>74.1</td>
<td></td>
</tr>
<tr>
<td>0.91</td>
<td>0.00</td>
<td>0.09</td>
<td>78.2</td>
<td></td>
</tr>
<tr>
<td>0.00</td>
<td>0.68</td>
<td>0.32</td>
<td>73.7</td>
<td></td>
</tr>
<tr>
<td>0.23</td>
<td>0.54</td>
<td>0.23</td>
<td>72.8</td>
<td></td>
</tr>
</tbody>
</table>

**CPU Times (Sun Ultra 1/140, sec)**

<table>
<thead>
<tr>
<th></th>
<th>Constant</th>
<th>T dependent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential</td>
<td>0.21</td>
<td>0.34</td>
</tr>
<tr>
<td>Simultaneous</td>
<td>0.90</td>
<td>4.62</td>
</tr>
</tbody>
</table>

$E =$ Ethanol; $\text{MEK} =$ Methyl Ethyl Ketone

$W =$ Water
## Results - Problem A2

### Azeotropes

<table>
<thead>
<tr>
<th>$\gamma_i$ parameters</th>
<th>A</th>
<th>C</th>
<th>M</th>
<th>$T$ (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant ((T_{ref}=152.4°C))</td>
<td>0.33</td>
<td>0.67</td>
<td>0.00</td>
<td>181.8</td>
</tr>
<tr>
<td></td>
<td>0.29</td>
<td>0.00</td>
<td>0.71</td>
<td>155.3</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.41</td>
<td>0.59</td>
<td>151.6</td>
</tr>
<tr>
<td>T-dependent</td>
<td>0.32</td>
<td>0.68</td>
<td>0.00</td>
<td>181.2</td>
</tr>
<tr>
<td></td>
<td>0.29</td>
<td>0.00</td>
<td>0.71</td>
<td>155.4</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.41</td>
<td>0.59</td>
<td>151.6</td>
</tr>
</tbody>
</table>

### CPU Times (sec)

<table>
<thead>
<tr>
<th></th>
<th>Constant</th>
<th>T dependent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential</td>
<td>0.51</td>
<td>0.82</td>
</tr>
<tr>
<td>Simultaneous</td>
<td>0.94</td>
<td>6.15</td>
</tr>
</tbody>
</table>

A = Acetone; C = Chloroform; M = Methanol
## Results - Problem A3

### Azeotropes

<table>
<thead>
<tr>
<th>$\gamma_i$ parameters</th>
<th>A</th>
<th>MA</th>
<th>M</th>
<th>T (°C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant*</td>
<td>0.53</td>
<td>0.47</td>
<td>0.00</td>
<td>55.7</td>
</tr>
<tr>
<td></td>
<td>0.75</td>
<td>0.00</td>
<td>0.25</td>
<td>54.5</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.68</td>
<td>0.32</td>
<td>54.4</td>
</tr>
<tr>
<td></td>
<td>0.27</td>
<td>0.47</td>
<td>0.26</td>
<td>54.3</td>
</tr>
<tr>
<td>T dependent</td>
<td>0.66</td>
<td>0.34</td>
<td>0.00</td>
<td>55.6</td>
</tr>
<tr>
<td></td>
<td>0.79</td>
<td>0.00</td>
<td>0.21</td>
<td>55.4</td>
</tr>
<tr>
<td></td>
<td>0.00</td>
<td>0.66</td>
<td>0.34</td>
<td>53.6</td>
</tr>
</tbody>
</table>

### CPU Times (sec)

<table>
<thead>
<tr>
<th></th>
<th>Constant</th>
<th>T dependent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential</td>
<td>0.40</td>
<td>1.04</td>
</tr>
<tr>
<td>Simultaneous</td>
<td>1.63</td>
<td>6.36</td>
</tr>
</tbody>
</table>

$A =$ Acetone(1); $MA =$ Methyl Acetate(2)
$M =$ Methanol(3)

*A* $\Lambda_{12} = 0.480, \Lambda_{21} = 1.550, \Lambda_{13} = 0.768$
$\Lambda_{31} = 0.566, \Lambda_{23} = 0.544, \Lambda_{32} = 0.650$
## Other Process Modeling Problems

1. Simple ethylene plant  
2. Mixer/divider network  
3. Adiabatic CSTR  
4. Flash with recycle  
5. Ammonia plant

<table>
<thead>
<tr>
<th>Problem</th>
<th>Number of Equations</th>
<th>Number of Roots</th>
<th>CPU time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>163</td>
<td>1</td>
<td>3.1</td>
</tr>
<tr>
<td>2</td>
<td>146</td>
<td>1</td>
<td>21.8</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>3</td>
<td>12.9</td>
</tr>
<tr>
<td>4</td>
<td>50</td>
<td></td>
<td>MAX</td>
</tr>
<tr>
<td>5</td>
<td>177</td>
<td>1</td>
<td>1003</td>
</tr>
</tbody>
</table>

CPU times on BBN TC2000 (1 processor).
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
  - Phase equilibrium (split)
  - Homogeneous azeotropes

- 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

• Students: Carol Schnepper (Air Liquide), James Hua (Shell Chemical), Bill Rooney (CMU), Rob Maier (Notre Dame), Gang Xu (Notre Dame)

• Funding:
  – ACS Petroleum Research Fund
  – National Science Foundation
  – Environmental Protection Agency
  – Department of Energy
  – Sun Microsystems, Inc.