New Interval Methodologies for Reliable Process Modeling

Chao-Yang Gau, Robert W. Maier
and Mark A. Stadtherr[1]
Department of Chemical Engineering
University of Notre Dame
Notre Dame, IN 46556 USA

AIChE Annual Meeting, Dallas, TX, Oct. 31– Nov. 5, 1999
Session Number 219: Design and Analysis II

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

Motivation

- In process modeling, chemical engineers frequently need to solve nonlinear equation systems in which the variables are constrained physically within upper and lower bounds; that is, to solve:

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}$$

$$\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U$$

- These problems may:
  - Have multiple solutions
  - Have no solution
  - Be difficult to converge to any solution

Motivation (continued)

- There is also frequent interest in globally minimizing a nonlinear function subject to nonlinear equality and/or inequality constraints; that is, to solve (globally):

$$\min_{\mathbf{x}} \phi(\mathbf{x})$$

subject to

$$\mathbf{h}(\mathbf{x}) = \mathbf{0}$$
$$\mathbf{g}(\mathbf{x}) \geq \mathbf{0}$$
$$\mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U$$

- These problems may:
  - Have multiple local minima (in some cases, it may be desirable to find them all)
  - Have no solution (infeasible NLP)
  - Be difficult to converge to any local minima

Motivation (continued)

- *One* approach for dealing with these issues is *interval analysis*.

- Interval analysis can
  - Provide the engineer with tools needed to solve modeling and optimization problems with complete certainty.
  - Provide problem-solving reliability not available when using standard local methods.
  - Deal automatically with rounding error, thus providing both mathematical and computational guarantees.

Motivation (continued)

- We have successfully applied interval Newton/generalized bisection (IN/GB) methods for
  - General process modeling problems (Schnepper and Stadtherr, 1996).
  - Phase stability and equilibrium problems using several different thermodynamic models (Stadtherr et al., 1994; Hua et al., 1996,1998,1999; Xu et al., 1998,1999).
  - Computation of azeotropes (homogeneous, reactive, heterogeneous) of multicomponent mixtures (Maier et al., 1998,1999).
  - Computation of mixture critical points (Stradi et al., 1998)
  - Parameter estimation in vapor-liquid equilibrium models (Gau and Stadtherr, 1998,1999).

- However, the IN/GB algorithm applied to date is very basic, and its performance is unacceptable on some problems.

Interval Method Used

- 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).
  - IN/GB can also be extended and employed as a deterministic approach for global optimization problems (e.g., Hansen, 1992).

- A general purpose approach; in general requires no simplifying assumptions or problem reformulations.

- Current implementation based on modifications of routines from INTBIS and INTLIB packages (Kearfott and coworkers)

Interval Method (Cont'd)

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

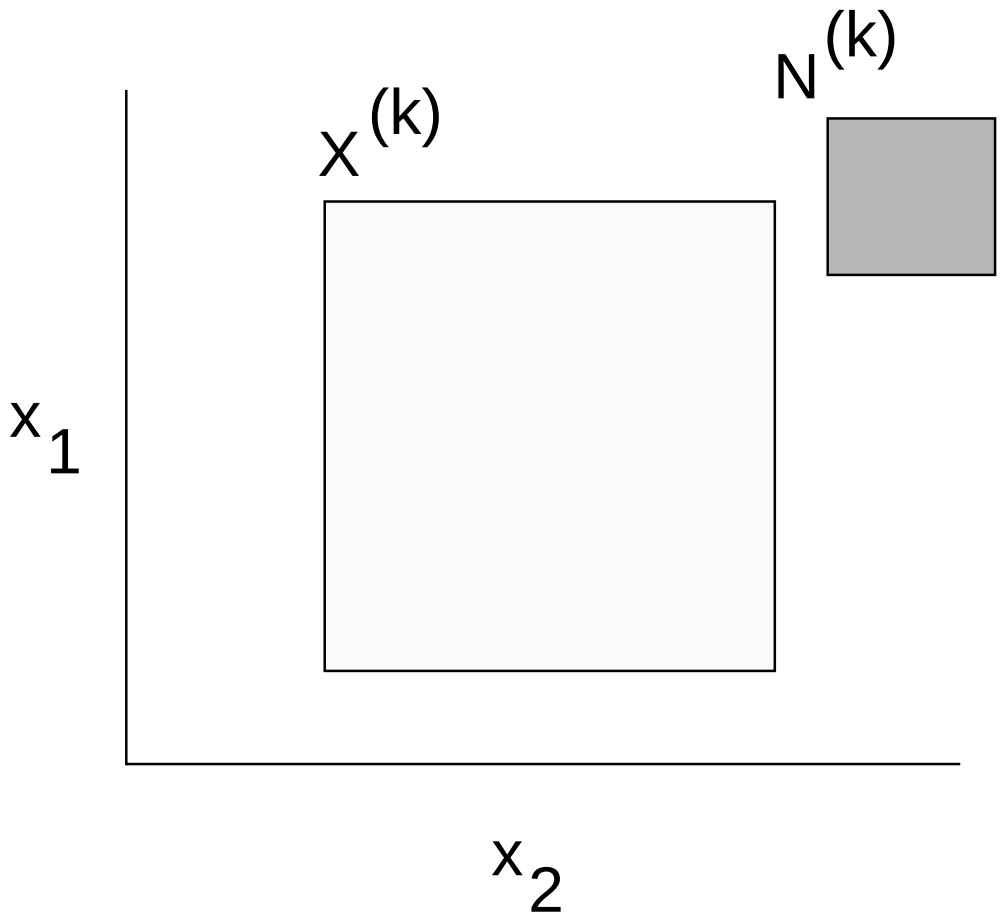
Basic iteration scheme: For a particular subinterval (box), $\mathbf{X}^{(k)}$, perform root inclusion test:

- (Range Test) Compute an interval extension of each function in the system.
  - If 0 is not an element of any interval extension, delete the box.
  - Otherwise,

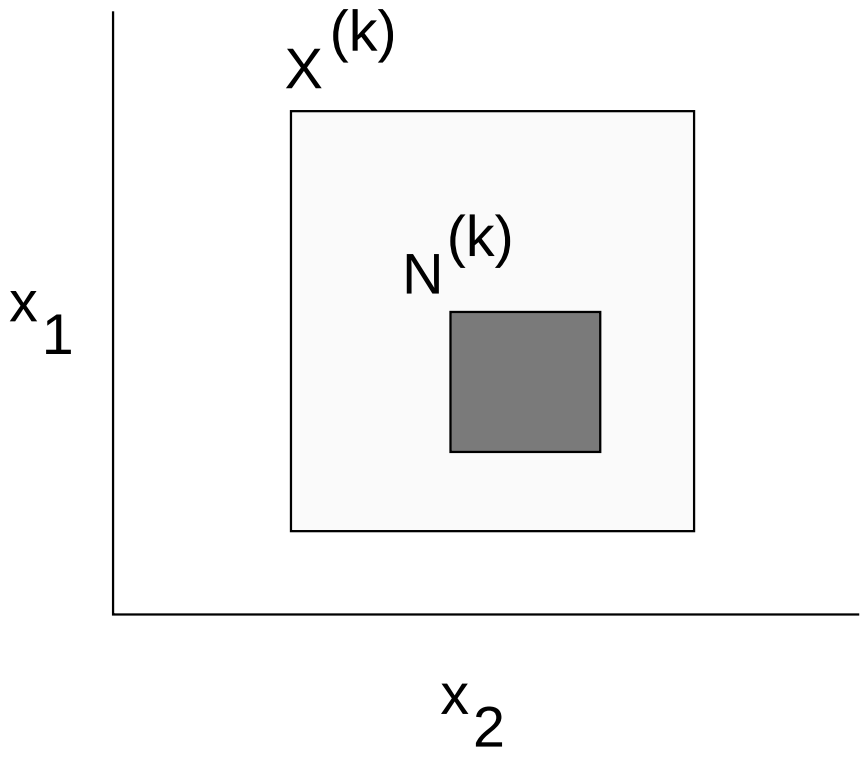
- (Interval Newton Test) Compute the $image$, $\mathbf{N}^{(k)}$, of the box by solving the linear interval equation system

$$F'(\mathbf{X}^{(k)})(\mathbf{N}^{(k)} - \mathbf{x}^{(k)}) = -\mathbf{f}(\mathbf{x}^{(k)})$$

  - $\mathbf{x}^{(k)}$ is some point in the interior of $\mathbf{X}^{(k)}$.
  - $F'\left(\mathbf{X}^{(k)}\right)$ is an interval extension of the Jacobian of $\mathbf{f}(\mathbf{x})$ over the box $\mathbf{X}^{(k)}$.



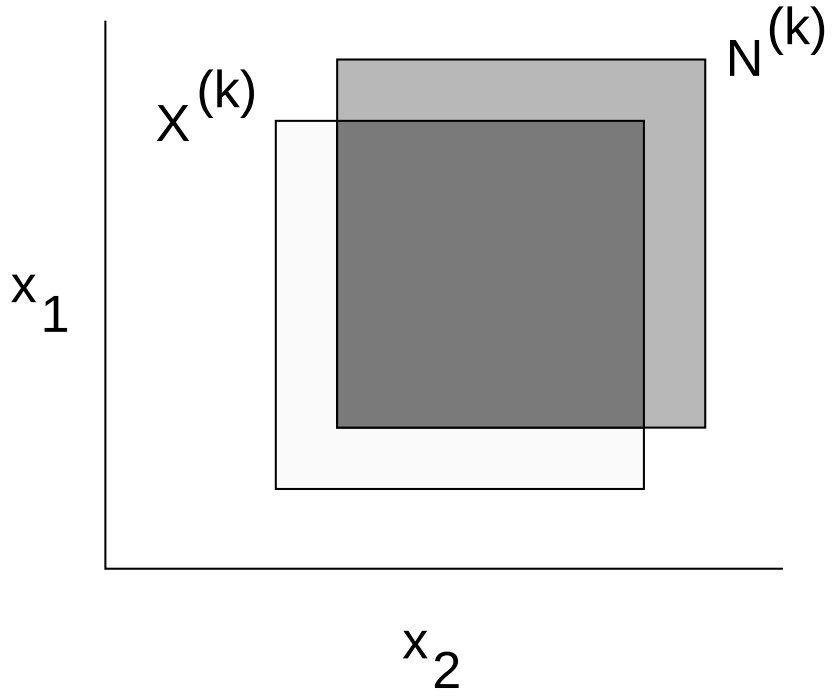
There was no solution in $X^{(k)}$



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.

Interval Method (Cont'd)

Some areas for potential algorithm improvement

- Tightening interval extensions of functions and Jacobian elements.
- Use of different tessellation schemes.
- Tighter bounds on the image $\mathbf{N}^{(k)}$ that encloses the solution set of the interval Newton equation.
  - **Preconditioning strategies (focus of this presentation).**

Solving the Interval Newton (IN) Equation

- Usually done by one iteration of preconditioned Gauss-Seidel scheme:
  - Solve

$$Y^{(k)} F'(\mathbf{X}^{(k)})(\mathbf{N}^{(k)} - \boldsymbol{x}^{(k)}) = -Y^{(k)} \mathbf{f}(\boldsymbol{x}^{(k)})$$

  - The scalar preconditioning matrix $Y^{(k)}$ is often chosen to be an inverse midpoint preconditioner Y^{inv}: inverse of the midpoint of the interval Jacobian matrix, or inverse of the Jacobian matrix at midpoint of the interval.

- One performance goal: Find smallest possible enclosure $\mathbf{N}$ of the solution set of the IN equation. The preconditioner used can have a strong effect on performance in this regard.

- Preconditioners that are optimal in some sense have been proposed by Kearfott (1990,1996) based on LP strategies

Preconditioning Strategies

- The preconditioner can be *designed* row by row during the Gauss-Seidel process try to achieve desired goals.

- Consider the i-th step of Gauss-Seidel and the i-th preconditioner row, $\mathbf{y}_i$,

$$\begin{aligned} N_i &= x_i - \frac{Q_i(\mathbf{y}_i)}{D_i(\mathbf{y}_i)} \\ &= x_i - \frac{\mathbf{y}_i\mathbf{f}(\mathbf{x}) + \sum_{\substack{j=1 \\ j\neq i}}^{n} \mathbf{y}_i\mathbf{A_j}(X_j - x_j)}{\mathbf{y}_i\mathbf{A_i}} \end{aligned}$$

  then take $N_i \cap X_i$. ($\mathbf{A_i}$ is the i-th column of the $F'(\mathbf{X})$ matrix.)

- Elements of $\mathbf{y}_i$ can be chosen to try to meet a desired goal.

Preconditioning Strategies

- Practical optimality criteria for preconditioner row $\mathbf{y}_i$:
  - Width-optimal preconditioner row :
    minimize width of $N_i \cap X_i$.
  - Endpoint-optimal preconditioner row:
    maximize the lower bound of N_i or minimize the upper bound of N_i.

- Optimality can be approached by a scheme in which the preconditioner row contains only one nonzero element. This can be called a pivoting preconditioner Y^P.

- We use a new hybrid scheme in which one or more of Y^{INV}, width-optimal Y^P, or endpoint-optimal Y^P are used, depending on the situation, and following heuristic rules.

Numerical Experiments

- Both equation-solving and global optimization problems were selected to illustrate the improvements that can be achieved using the new hybrid preconditioner.
  - Problem 1: Error-in-variables parameter estimation.
  - Problem 2 : Phase stability analysis for LLE system.
  - Problem 3 : Computation of critical points of mixtures.
  - Problem 4 : Computation of heterogeneous azeotropes.

- We compared use of Y^{INV} alone to use of the new hybrid preconditioner on a Sun Ultra 2/1300 workstation.

Results and Discussion

- Problem 1: Error-in-variables parameter estimation
  - Global optimization with 2 parameter variables and 10 state variables.
  - Point evaluations of objective function done at the midpoint of current box used for bounding in objective range test.
  - Use Van Laar equation to model experimental vapor-liquid equilibrium data.
  - Using Y^{inv} alone took $>$ 4 CPU days.
  - Using new hybrid preconditioner took 1504 CPU seconds.

Results and Discussion (cont.)

- Problem 2 : Phase stability analysis for LLE system
  - Equation-solving problem with 6 independent variables.
  - Use UNIQUAC model to for computing excess Gibbs energy.
  - Using Y^{inv} alone took 50217 CPU seconds
  - Using new hybrid preconditioner took 152 CPU seconds.

Results and Discussion (cont.)

- Problem 3 : Computation of critical points of mixtures
  - Equation-solving problem with 6 variables (four component mixture).
  - Use Peng-Robinson equation of state to model both the liquid and gas phases.
  - Using Y^{inv} alone took 2094 CPU seconds.
  - Using new hybrid preconditioner took 658 CPU seconds.

Results and Discussion (cont.)

- Problem 4 : Computation of heterogeneous azeotropes
  - Equation-solving problem with 11 variables (3 components).
  - Use NRTL activity coefficient model.
  - Using Y^{inv} alone took $>$ 1 CPU days
  - Using new hybrid preconditioner took 270 CPU seconds.

Concluding Remarks

- Use of the new hybrid pivoting preconditioner scheme provides an approach to manipulate the interval Gauss-Seidel process to achieve greater efficiency.

- This has led to large reductions in CPU time for all problems tested, and in some cases, reductions of 2 or more orders of magnitude.

- For difficult problems, the additional work required to construct the preconditioner is easily overcome by a large reduction in the number of intervals that must be processed.

- For more details, please see the Poster 213c in the High Performance Computing Poster Session, Wednesday, 7pm, Khmer Pavilion.

- These slides will be available next week at http://www.nd.edu/~markst/presentations.html

Acknowledgments

- ACS PRF 30421-AC9
- NSF CTS95-22835, NSF DMI96-96110, NSF EEC97-00537-CRCD
- EPA R824731-01-0, R826734-01-0
- DOE DE-FG07-96ER14691
- Sun Microsystems, Inc.