Global Optimization for Parameter Estimation of Dynamic Systems

Youdong Lin and Mark A. Stadtherr

Department of Chemical and Biomolecular Engineering

University of Notre Dame

Notre Dame, IN 46556

AIChE Annual Meeting, Cincinnati, OH

October 31, 2005

Outline

- Background
- Interval Analysis
- Taylor Models
- Validated Solutions for Parametric ODEs
- Algorithm Summary
- Computational Studies
- Concluding Remarks and Acknowledgments

Background

- Parameter estimation is a key step in development of mathematical models
- Models of interest may be ODEs/DAEs
- Minimization of a weighted squared error

$$\min_{\boldsymbol{\theta}, \boldsymbol{z}_\mu} \phi = \sum_{m \in M} \sum_{\mu=1}^{r} (z_{\mu,m} - \bar{z}_{\mu,m})^2$$

$$\text{s.}t. \quad \dot{\boldsymbol{z}} = \boldsymbol{f}(\boldsymbol{z}, \boldsymbol{\theta}, t), \quad \boldsymbol{z}(t_0) = \boldsymbol{z}_0$$

$$\boldsymbol{\theta} \in \boldsymbol{\Theta}$$

$$\boldsymbol{z}_\mu = \boldsymbol{z}(t_\mu), \quad t_\mu \in [t_0, t_f]$$

- Sequential approach – eliminate $\boldsymbol{z}_\mu$ using parametric ODE solver
- Multiple local solutions – a need for global optimization

Deterministic Global Optimization with Dynamic Systems

- Much recent interest, e.g.
  - Esposito and Floudas (2000)
  - Chachuat and Latifi (2003)
  - Papamichail and Adjiman (2002, 2004)
  - Singer and Barton (2004)
- New approach: branch and reduce algorithm based on interval analysis
  - Construct Taylor models of the states using a new validated solver for parametric ODEs (VSPODE) (Lin and Stadtherr, 2005)
  - Compute the Taylor model T_ϕ of the objective function
  - Perform constraint propagation procedure using $T_\phi \leq \hat{\phi}$, to reduce the parameter domain

Interval Analysis

- A real interval

$$X = [\underline{X}, \overline{X}] = \{x \in \Re \mid \underline{X} \leq x \leq \overline{X}\}$$

- A real interval vector – a box

$$\boldsymbol{X} = (X_1, X_2, \cdots, X_n)^T$$

- Interval arithmetic – basic operations and elementary functions
- An interval extension of a function $f(\boldsymbol{x})$ over $\boldsymbol{X}$

$$F(\boldsymbol{X}) \supseteq \{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \boldsymbol{X}\}$$

- Natural interval extension – leads to overestimation (dependence problem)

Taylor Models

- Taylor Model T_f – an interval extension of a function over $\boldsymbol{X}$

$$T_f = (p_f, R_f)$$

$$p_f = \sum_{i=0}^{q} \frac{1}{i!} \left[(\boldsymbol{X} - \boldsymbol{x}_0) \cdot \nabla \right]^i f(\boldsymbol{x}_0)$$

$$R_f = \frac{1}{(q+1)!} \left[(\boldsymbol{X} - \boldsymbol{x}_0) \cdot \nabla \right]^{q+1} F\left[\boldsymbol{x}_0 + (\boldsymbol{X} - \boldsymbol{x}_0)\Xi \right]$$

where,

$$\boldsymbol{x}_0 \in \boldsymbol{X}; \quad \Xi = [0, 1]$$

$$[\boldsymbol{g} \cdot \nabla]^k = \sum_{\substack{j_1 + \cdots + j_m = k \\ 0 \le j_1, \cdots, j_m \le k}} \frac{k!}{j_1! \cdots j_m!} g_1^{j_1} \cdots g_m^{j_m} \frac{\partial^k}{\partial x_1^{j_1} \cdots \partial x_m^{j_m}}$$

- p_f is a polynomial function; store and operate on its coefficients only

Taylor Models - Remainder Differential Algebra (RDA)

- Basic operations

$$T_{f\pm g} = (p_f, R_f) \pm (p_g, R_g) = (p_f \pm p_g, R_f \pm R_g)$$

$$\begin{aligned} T_{f\times g} &= (p_f, R_f) \times (p_g, R_g) \\ &= p_f \times p_g + p_f \times R_g + p_g \times R_f + R_f \times R_g \\ &= (p_{f\times g}, R_{f\times g}) \end{aligned}$$

where,

$$p_{f\times g} = p_f \times p_g - p_e$$

$$R_{f\times g} = B(p_e) + B(p_f) \times R_g + B(p_g) \times R_f + R_f \times R_g$$

- $B(p)$ indicates an interval bound on the function p.
- Reciprocal operation and intrinsic functions can also be defined.
- It is possible to compute Taylor models of complex functions.

Taylor Models - Range Bounding

- Exact range bounding of the interval polynomials – NP hard
- Direct evaluation of the interval polynomials – inefficient
- Focus on bounding the dominant part (1st and 2nd order terms)
- Exact range bounding of a general interval quadratic - computationally expensive
- A compromise approach – 1st order and diagonal elements of 2nd order

$$B(p) = \sum_{i=1}^{m} \left[a_i \left(X_i - x_{i0} \right)^2 + b_i (X_i - x_{i0}) \right] + S$$

$$= \sum_{i=1}^{m} \left[a_i \left(X_i - x_{i0} + \frac{b_i}{2a_i} \right)^2 - \frac{b_i^2}{4a_i} \right] + S,$$

where, S is the interval bound of other terms by direct evaluation

Taylor Models - Constraint Propagation

- Goal – to reduce part of domain not satisfying $c(\boldsymbol{x}) \leq 0$
- For some $i = 1, 2 \cdots, m$

$$
\begin{aligned}
B(T_c) &= B(p_c) + R_c = a_i \left(X_i - x_{i0} + \frac{b_i}{2a_i} \right)^2 - \frac{b_i^2}{4a_i} + S_i \leq 0 \\
\Longrightarrow \quad & a_i U_i^2 \leq V_i, \quad \text{with } U_i = X_i - x_{i0} + \frac{b_i}{2a_i} \text{ and } V_i = \frac{b_i^2}{4a_i} - S_i \\
\Longrightarrow \quad & U_i = \begin{cases} \emptyset & \text{if } a_i > 0 \text{ and } \overline{V_i} < 0 \\ \left[-\sqrt{\frac{\overline{V_i}}{a_i}}, \sqrt{\frac{\overline{V_i}}{a_i}} \right] & \text{if } a_i > 0 \text{ and } \overline{V_i} \geq 0 \\ [-\infty, \infty] & \text{if } a_i < 0 \text{ and } \overline{V_i} \geq 0 \\ \left[-\infty, -\sqrt{\frac{\overline{V_i}}{a_i}} \right] \cup \left[\sqrt{\frac{\overline{V_i}}{a_i}}, \infty \right] & \text{if } a_i < 0 \text{ and } \overline{V_i} < 0 \end{cases} \\
\Longrightarrow \quad & X_i = X_i \cap \left(U_i + x_{i0} - \frac{b_i}{2a_i} \right)
\end{aligned}
$$

Validated Solutions for Parametric ODEs

- Consider the IVP for the parametric ODEs

$$\dot{\boldsymbol{y}} = \boldsymbol{f}(\boldsymbol{y}, \boldsymbol{\theta}), \quad \boldsymbol{y}(t_0) = \boldsymbol{y}_0, \quad \boldsymbol{\theta} \in \boldsymbol{\Theta}$$

- Validated methods:
  - Guarantee there exists a unique solution $\boldsymbol{y}$ in the interval $[t_0, t_f]$, for each $\boldsymbol{\theta} \in \boldsymbol{\Theta}$
  - Compute the interval $\boldsymbol{Y}_{t_f}$ that encloses all solutions of the ODEs at t_f.
- Tools – AWA, VNODE, COSY VI, VSPODE, etc.

Validated Solutions for Parametric ODEs (Cont'd)

- VSPODE (Lin and Stadtherr, 2005) – novel use of Taylor model approach for dependency problem in solving ODEs with interval valued parameters
- Phase 1 – Validate existence and uniqueness (h_j and $\tilde{\boldsymbol{Y}}_j$) – like in VNODE

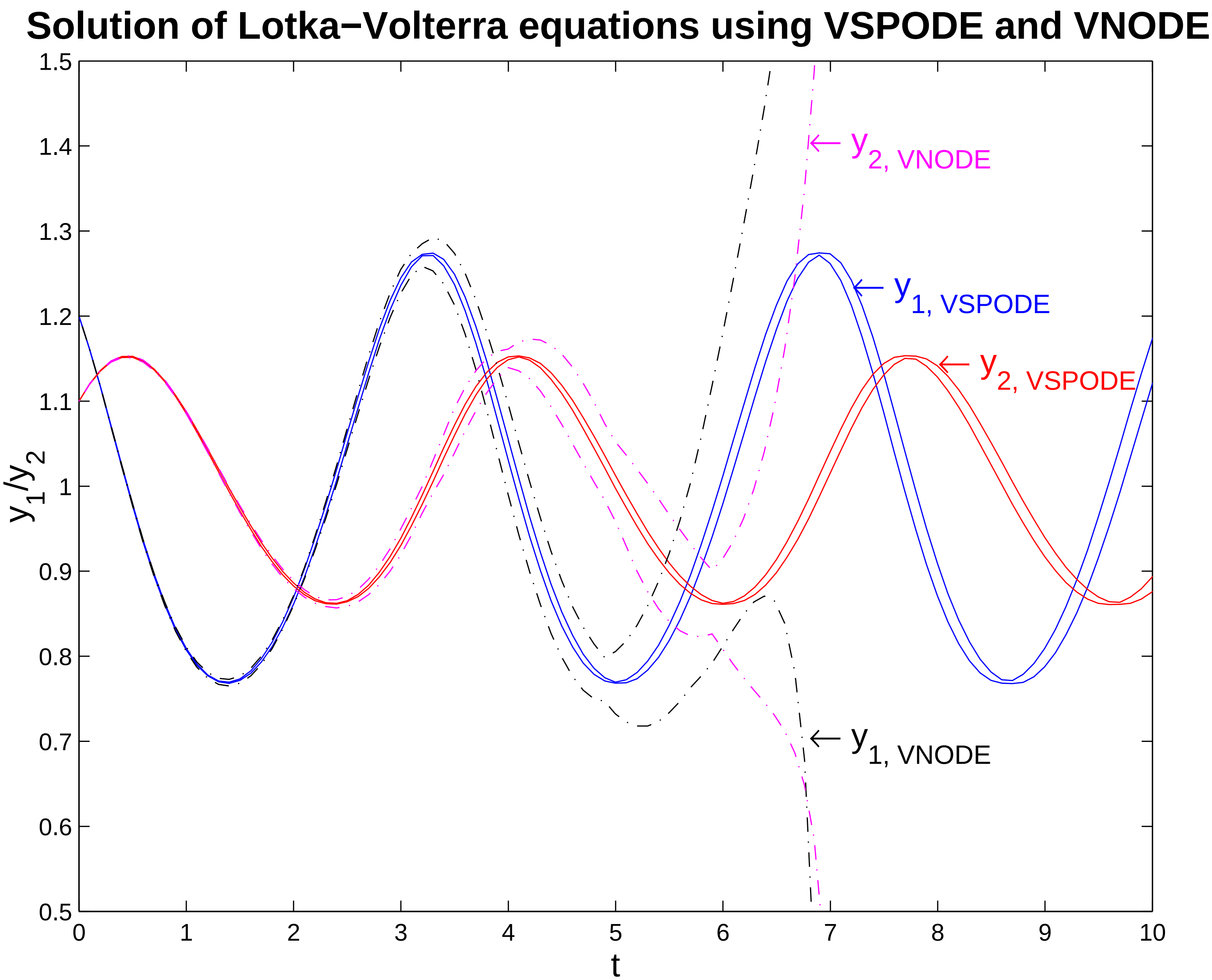
$$\tilde{\boldsymbol{Y}}_j = \sum_{i=0}^{k-1} [0, h_j]^i \boldsymbol{F}^{[i]}(\boldsymbol{Y}_j, \boldsymbol{\Theta}) + [0, h_j]^k \boldsymbol{F}^{[k]}(\tilde{\boldsymbol{Y}}_j^0, \boldsymbol{\Theta}) \subseteq \tilde{\boldsymbol{Y}}_j^0$$

- Phase 2 – Compute tighter enclosure
  - Dependence problem – Taylor model
  - Wrapping effect – QR factorization
  - Solutions: $\boldsymbol{T}_{y_{j+1}} = p_{y_{j+1}} + A_{j+1}\boldsymbol{V}_{j+1}$

Validated Solutions for Parametric ODEs (Cont'd)

- Example – Lotka-Volterra equations

$$\dot{y}_1 = \theta_1 y_1 (1 - y_2)$$
$$\dot{y}_2 = \theta_2 y_2 (y_1 - 1)$$
$$t \in [0, 10]$$
$$y_1(0) = 1.2$$
$$y_2(0) = 1.1$$
$$\theta_1 \in 3 + [-0.01, 0.01]$$
$$\theta_2 \in 1 + [-0.01, 0.01]$$


Solution of Lotka−Volterra equations using VSPODE and VNODE
y_1/y_2
t
0
1
2
3
4
5
6
7
8
9
10
0.5
0.6
0.7
0.8
0.9
1
1.1
1.2
1.3
1.4
1.5
$y_{2,\ \mathrm{VNODE}}$
$y_{1,\ \mathrm{VSPODE}}$
$y_{2,\ \mathrm{VSPODE}}$
$y_{1,\ \mathrm{VNODE}}$

Branch and Reduce Algorithm Summary

Beginning with initial parameter interval $\mathbf{\Theta}^{(0)}$

- Establish $\hat{\phi}$, the upper bound on global minimum using p^2 local minimizations
- Iterate: for subinterval $\mathbf{\Theta}^{(k)}$
  1. Compute Taylor models of the states using VSPODE, and then obtain T_ϕ
  2. Perform constraint propagation using $T_\phi \leq \hat{\phi}$ to reduce $\mathbf{\Theta}^{(k)}$
  3. If $\mathbf{\Theta}^{(k)} = \emptyset$, go to next subinterval
  4. If $(\hat{\phi} - \underline{B(T_\phi)})/|\hat{\phi}| \leq \epsilon$, discard $\mathbf{\Theta}^{(k)}$ and go to next subinterval
  5. If $\overline{B(T_\phi)} < \hat{\phi}$, update $\hat{\phi}$ with local minimization, go to step 2
  6. If $\mathbf{\Theta}^{(k)}$ is sufficiently reduced, go to step 1
  7. Otherwise, bisect $\mathbf{\Theta}^{(k)}$ and go to next subinterval

Computational Studies - Example 1

- First-order irreversible series reaction (Esposito and Floudas, 2000)

$$A \xrightarrow{\theta_1} B \xrightarrow{\theta_2} C$$

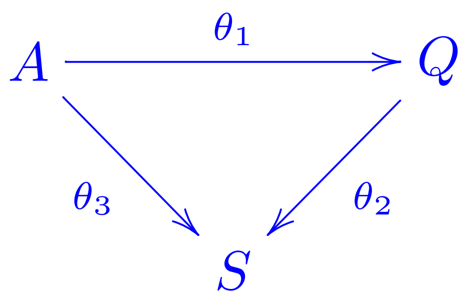
- The differential equation model

$$\begin{aligned} \dot{z}_A &= -\theta_1 z_A \\ \dot{z}_B &= \theta_1 z_A - \theta_2 z_B \\ \boldsymbol{z}_0 &= [1, 0] \\ \boldsymbol{\theta} &\in [0, 10] \times [0, 10] \end{aligned}$$

- Solution: $\boldsymbol{\theta}^* = (5.0035, 1.0000)$ and $\phi^* = 1.1858 \times 10^{-6}$
- Results: 4 iterations and < 0.1 CPU seconds

Computational Studies - Example 2

- Catalytic Cracking of Gas Oil (Esposito and Floudas, 2000)



- The differential equation model

$$
\begin{aligned}
\dot{z}_A &= -(\theta_1 + \theta_3) z_A^2 \\
\dot{z}_Q &= \theta_1 z_A^2 - \theta_2 z_Q \\
\boldsymbol{z}_0 &= [1, 0] \\
\boldsymbol{\theta} &\in [0, 20] \times [0, 20] \times [0, 20]
\end{aligned}
$$

- Solution: $\boldsymbol{\theta}^* = (12.2139, 7.9798, 2.2217)$ and $\phi^* = 2.6557 \times 10^{-3}$
- Results: 359 iterations and 14.3 CPU seconds

Computational Performance Comparison (CPU seconds)

	Example 1		Example 2	
Method	Reported	Adjusted	Reported	Adjusted
This work (Intel P4 3.2GHz)	< 0.1	< 0.1	14.3	14.3
Papamichail and Adjiman (SUN UltraSPARC-II 360MHz)	801	102.5	35478	4541
Chachuat and Latifi (Machine not reported)	280	-	10400	-
Esposito and Floudas* (HP 9000 model J2240)	13.30	1.53	100.21	11.5

Adjusted = Approximate CPU time adjusted for machine used based on SPEC benchmarks

* Does not provide rigorous guarantee of global optimality.

Concluding Remarks and Acknowledgments

- A deterministic global optimization approach based on interval analysis can be used to estimate the parameters of dynamic systems
- A validated solver for parametric ODEs is used to construct bounds on the states of dynamic systems
- An efficient constraint propagation procedure is used to reduce the incompatible parameter domain
- This approach can be combined with the interval-Newton method (Lin and Stadtherr, 2005)
  - True global optimum instead of ϵ-convergence
  - May or may not reduce CPU time required
- Acknowledgments

  - Indiana 21st Century Research & Technology Fund
  - Department of Energy