Deterministic Global Optimization Algorithm and Nonlinear Dynamics

C. S. Adjiman and I. Papamichail

Centre for Process Systems Engineering
Department of Chemical Engineering and Chemical Technology
Imperial College London

Global Optimization Theory Institute, Argonne National Laboratory
September 8-10, 2003

Financial support: Engineering and Physical Sciences Research Council

Outline

- Motivation
- Dynamic Optimization: Problem and Methods
- Convex Relaxation of the Dynamic Information
- Deterministic Global Optimization Algorithm
- Convergence of the Algorithm
- Case Studies
- Conclusions and Perspectives

Applications of Dynamic Optimization

- Chemical systems
  - Determination of Kinetic Constants from Time Series Data (Parameter Estimation)
  - Optimal Control of Batch and semi-Batch Chemical Reactors
  - Safety Analysis of Industrial Processes
- Biological and ecological systems
- Economic and other dynamic systems

Dynamic Optimization Problem

$$\min_{p} J(x(t_i, p), p\,;\; i = 0, 1, ..., NP)$$

subject to:

$$g_i(x(t_i, p), p) \leq 0\;,\; i = 0, 1, ..., NP$$

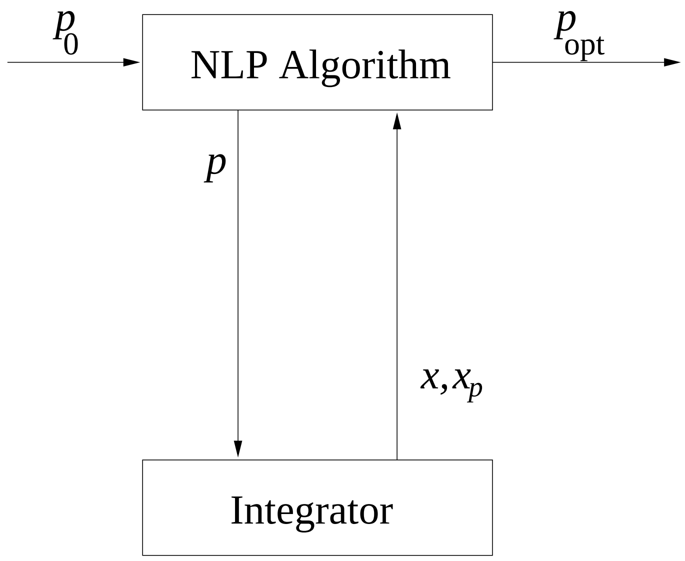
$$p^L \leq p \leq p^U$$

$$\dot{x} = f(t, x, p)\;,\; \forall t \in [t_0, t_{NP}]$$

$$x(t_0, p) = x_0(p)$$

Solution approaches

- Simultaneous approach: full discretization
- Sequential approach:



Usually *nonconvex* NLP Problems

- Multiple optimum solutions exist
- Commercially available numerical solvers guarantee *local optimum* solutions
- Poor economic performance

Algorithm classes (combined with simultaneous or sequential approach)

Stochastic algorithms

Deterministic algorithms

Global optimization methods
Simultaneous approaches

- Application of global optimization algorithms for NLPs

- Issues: problem size; quality of discretization.

- Smith and Pantelides (1996): spatial BB + reformulation

- Esposito and Floudas (2000): αBB algorithm

Global optimization methods: Sequential approaches

- Stochastic algorithms
  - Luus et al. (1990): direct search procedure.
  - Banga and Seider (1996), Banga et al. (1997): randomly directed search.
- Deterministic algorithms
  - New techniques for convex relaxation of time-dependent parts of problem
  - Lack of analytical forms for the constraints / objective function
  - Esposito and Floudas (2000): extension of αBB to handle nonlinear dynamics.
  - Singer and Barton (2002): convex relaxation of integral objective function with linear dynamics
  - Papamichail and Adjiman (2002): convex relaxations of nonlinear dynamics.

Outline

- Motivation
- Dynamic Optimization: Problem and Methods
- Convex Relaxation of the Dynamic Information
- Deterministic Global Optimization Algorithm
- Convergence of the Algorithm
- Case Studies
- Conclusions and Perspectives

Reformulated NLP Problem

$$\min_{\widehat{x},p} J(\widehat{x}_i, p\,;\; i = 0, 1, ..., NP)$$

subject to:

$$g_i(\widehat{x}_i, p) \leq 0\;,\; i = 0, 1, ..., NP$$

$$\widehat{x}_i = x(t_i, p)\;,\; i = 0, 1, ..., NP$$

$$p \in [p^L, p^U]$$

$$\dot{x} = f(t, x, p)\;,\; \forall t \in [t_0, t_{NP}]$$

$$x(t_0, p) = x_0(p)$$

Convex Relaxation of J and g_i (1)

$$f(z) = f_{CT}(z) + \sum_{i=1}^{bt} b_i z_{B_i,1} z_{B_i,2} + \sum_{i=1}^{ut} f_{UT,i}(z_i) + \sum_{i=1}^{nt} f_{NT,i}(z)$$

Underestimating Bilinear Terms (McCormick, 1976)

$$w = z_1 z_2 \quad \text{over} \quad [z_1^L, z_1^U] \times [z_2^L, z_2^U]$$

$$\begin{aligned} w &\geq z_1^L z_2 + z_2^L z_1 - z_1^L z_2^L \\ w &\geq z_1^U z_2 + z_2^U z_1 - z_1^U z_2^U \\ w &\leq z_1^L z_2 + z_2^U z_1 - z_1^L z_2^U \\ w &\leq z_1^U z_2 + z_2^L z_1 - z_1^U z_2^L \end{aligned}$$

Underestimating Univariate Concave Terms

$$\breve{f}_{UT}(z) = f_{UT}(z^L) + \frac{f_{UT}(z^U) - f_{UT}(z^L)}{z^U - z^L}(z - z^L) \quad \text{over} \quad [z^L, z^U] \subset \Re$$

Convex Relaxation of J and g_i (2)

Underestimating General Nonconvex Terms in $\mathcal{C}^2$
(Maranas and Floudas, 1994; Androulakis *et al.*, 1995)

$$\bar{f}_{NT}(z) = f_{NT}(z) + \sum_{i=1}^{m} \alpha_i (z_i^L - z_i)(z_i^U - z_i) \quad \text{over} \quad [z^L, z^U] \subset \Re^m$$

- $\bar{f}_{NT}(z)$ is always less than f_{NT}
- $\bar{f}_{NT}(z)$ is convex if α_i is big enough

$$H_{\bar{f}_{NT}}(z) = H_{f_{NT}}(z) + 2\,\mathrm{diag}(\alpha_i)$$

Rigorous α calculations using the scaled Gerschgorin method
(Adjiman *et al.*, 1998)

$$\forall z \in [z^L, z^U] \; H_{f_{NT}}(z) \in [H_{f_{NT}}] = H_{f_{NT}}([z^L, z^U])$$

Convex Relaxation of $\widehat{x}_i = x(t_i, p)$

$$\widehat{x}_i - x(t_i, p) \leq 0$$
$$x(t_i, p) - \widehat{x}_i \leq 0$$

Constant bounds: $\underline{x}(t_i) \leq \widehat{x}_i \leq \overline{x}(t_i)$

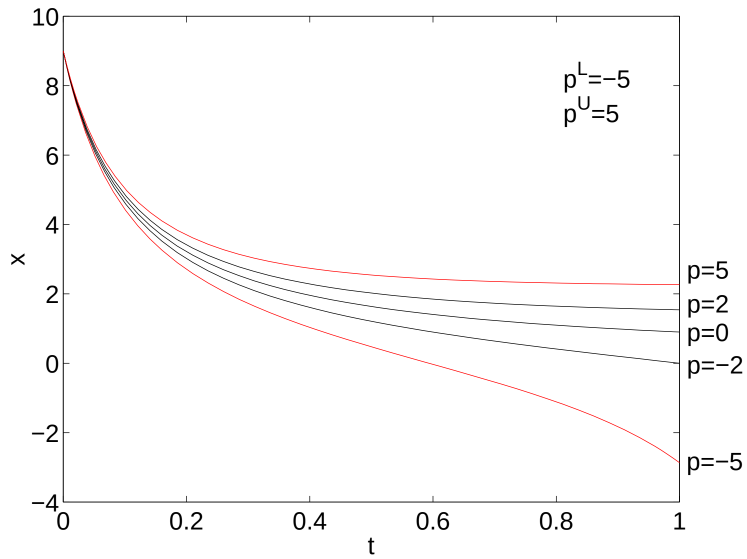
Affine bounds: $\underline{M}(t_i)p + \underline{N}(t_i) \leq \widehat{x}_i \leq \overline{M}(t_i)p + \overline{N}(t_i)$

α-*based bounds* (Esposito and Floudas, 2000):

$$\widehat{x}_{ik} - x_k(t_i, p) + \sum_{j=1}^{r} \alpha_{kij}^{-}(p_j^L - p_j)(p_j^U - p_j) \leq 0$$
$$x_k(t_i, p) + \sum_{j=1}^{r} \alpha_{kij}^{+}(p_j^L - p_j)(p_j^U - p_j) - \widehat{x}_{ik} \leq 0$$

Illustrative example

$$\dot{x}(t) = -x(t)^2 + p\,,\ \forall t \in [0,1]$$
$$x(0) = 9$$
$$p \in [-5,5]$$



How do we find bounding trajectories?

Differential Inequalities (1)

Consider the following parameter dependent ODE:

$$\dot{x} = f(t, x, p), \ \forall t \in [t_0, t_{NP}]$$

$$x(t_0, p) = x_0(p)$$

$$p \in [p^L, p^U] \subset \Re^r$$

where x and $\dot{x} \in \Re^n$, $f: (t_0, t_{NP}] \times \Re^n \times [p^L, p^U] \mapsto \Re^n$ and $x_0: [p^L, p^U] \mapsto \Re^n$.

Let $x = (x_1, x_2, ..., x_n)^T$ and $x_{k^-} = (x_1, x_2, ..., x_{k-1}, x_{k+1}, ..., x_n)^T$. The notation $f(t, x, p) = f(t, x_k, x_{k^-}, p)$ is used.

Theory on diff. inequalities (Walter, 1970) has been extended.

Differential Inequalities (2)

Bounds on the solutions of the parameter dependent ODE:

$$\dot{\underline{x}}_k = \inf f_k(t, \underline{x}_k, [\underline{x}_{k-}, \overline{x}_{k-}], [p^L, p^U])$$

$$\dot{\overline{x}}_k = \sup f_k(t, \overline{x}_k, [\underline{x}_{k-}, \overline{x}_{k-}], [p^L, p^U])$$

$$\forall t \in [t_0, t_{NP}] \text{ and } k = 1, ..., n$$

$$\underline{x}(t_0) = \inf x_0([p^L, p^U])$$

$$\overline{x}(t_0) = \sup x_0([p^L, p^U])$$

$\underline{x}(t)$ is a subfunction and $\overline{x}(t)$ is a superfunction for the solution of the ODE, i.e.,

$$\underline{x}(t) \le x(t, p) \le \overline{x}(t) \ , \ \forall p \in [p^L, p^U] \ , \ \forall t \in [t_0, t_{NP}]$$

Quasi-monotonicity

Definition 1: Let g(x) be a mapping $g : \mathcal{D} \mapsto \Re$ with $\mathcal{D} \subseteq \Re^n$. Again the notation $g(x) = g(x_k, x_{k-})$ is used. The function g is called unconditionally partially isotone (antitone) on $\mathcal{D}$ with respect to the variable x_k if

$$g(x_k, x_{k-}) \leq g(\tilde{x}_k, x_{k-}) \text{ for } x_k \leq \tilde{x}_k \ \ (x_k \geq \tilde{x}_k)$$

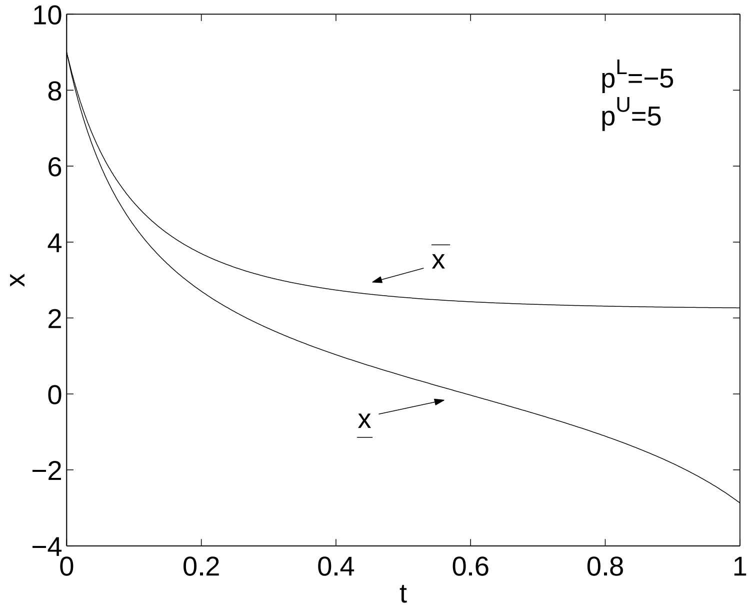
and for all $(x_k, x_{k-}), (\tilde{x}_k, x_{k-}) \in \mathcal{D}$.

Definition 2: Let $f(t,x) = (f_1(t,x), ..., f_2(t,x))^T$ and each $f_k(t, x_k, x_{k-})$ be unconditionally partially isotone on $\mathcal{I}_0 \times \Re \times \Re^{n-1}$ with respect to any component of x_{k-}, but not necessarily with respect to x_k. Then f is quasi-monotone increasing on $\mathcal{I}_0 \times \Re^n$ with respect to x (Walter, 1970)

Example: Constant bounds

$$\dot{\underline{x}}(t) = -\underline{x}(t)^2 - 5 \qquad \dot{\overline{x}}(t) = -\overline{x}(t)^2 + 5$$

$$\underline{x}(0) = 9 \qquad \overline{x}(0) = 9$$



Parameter Dependent Bounds

Let $\underline{f}(t, x, p) \le f(t, x, p)\ \forall x \in [\underline{x}(t), \overline{x}(t)],\ \forall p \in [p^L, p^U],\ \forall t \in [t_0, t_{NP}]$ and $\underline{\underline{x}}_0(p) \le x_0(p)\ \forall p \in [p^L, p^U]$, where $\underline{f} :\ [t_0, t_{NP}] \times \Re^n \times [p^L, p^U] \mapsto \Re^n$ and $\underline{\underline{x}}_0 :\ [p^L, p^U] \mapsto \Re^n$.

If $\underline{f}$ is quasi-monotone increasing w.r.t. x and $\underline{\underline{x}}(t, p)$ is the solution of the ODE:

$$\dot{\underline{\underline{x}}} = \underline{f}(t, \underline{\underline{x}}, p),\ \forall t \in [t_0, t_{NP}]$$

$$\underline{\underline{x}}(t_0, p) = \underline{\underline{x}}_0(p)$$

$$p \in [p^L, p^U]$$

then $\quad \underline{\underline{x}}(t, p) \le x(t, p)\ ,\ \forall p \in [p^L, p^U]\ ,\ \forall t \in [t_0, t_{NP}]$.

Affine Bounds

Let $\underline{f}(t,\underline{\underline{x}},p) = A(t)\underline{\underline{x}} + B(t)p + C(t)$ and $\underline{\underline{x}}_0(p) = Dp + E$, where $A(t)$, $B(t)$ and $C(t)$ are continuous on $[t_0, t_{NP}]$. Then the analytical solution is (Zadeh and Desoer, 1963):

$$\begin{aligned}\underline{\underline{x}}(t,p) \;&=\; \left\{\Phi(t,t_0)D + \int_{t_0}^{t} \Phi(t,\tau)B(\tau)d\tau\right\}p \\ &\quad +\Phi(t,t_0)E + \int_{t_0}^{t} \Phi(t,\tau)C(\tau)d\tau,\end{aligned}$$

where $\Phi(t,t_0)$ is the transition matrix:

$$\begin{aligned}&\dot{\Phi}(t,t_0) = \underline{A}(t)\Phi(t,t_0) \;\; \forall t \in [t_0, t_{NP}] \\ &\Phi(t_0,t_0) = I\end{aligned}$$

and I is the identity matrix. $\underline{\underline{x}}(t,p) = \underline{M}(t)p + \underline{N}(t)$.

$\underline{M}(t_i)$, $\underline{N}(t_i)$ calculation

1. Apply $\underline{\underline{x}}(t_i, p) = \underline{M}(t_i)p + \underline{N}(t_i)$ for $r+1$ values of p
2. Calculate $\underline{\underline{x}}(t_i, p)$ for the $r+1$ values of p from the integration of the linear ODE
3. Solve n linear systems to find the $r+1$ unknowns for each one of the n dimensions of x

Example: Affine bounds

Underestimating IVP

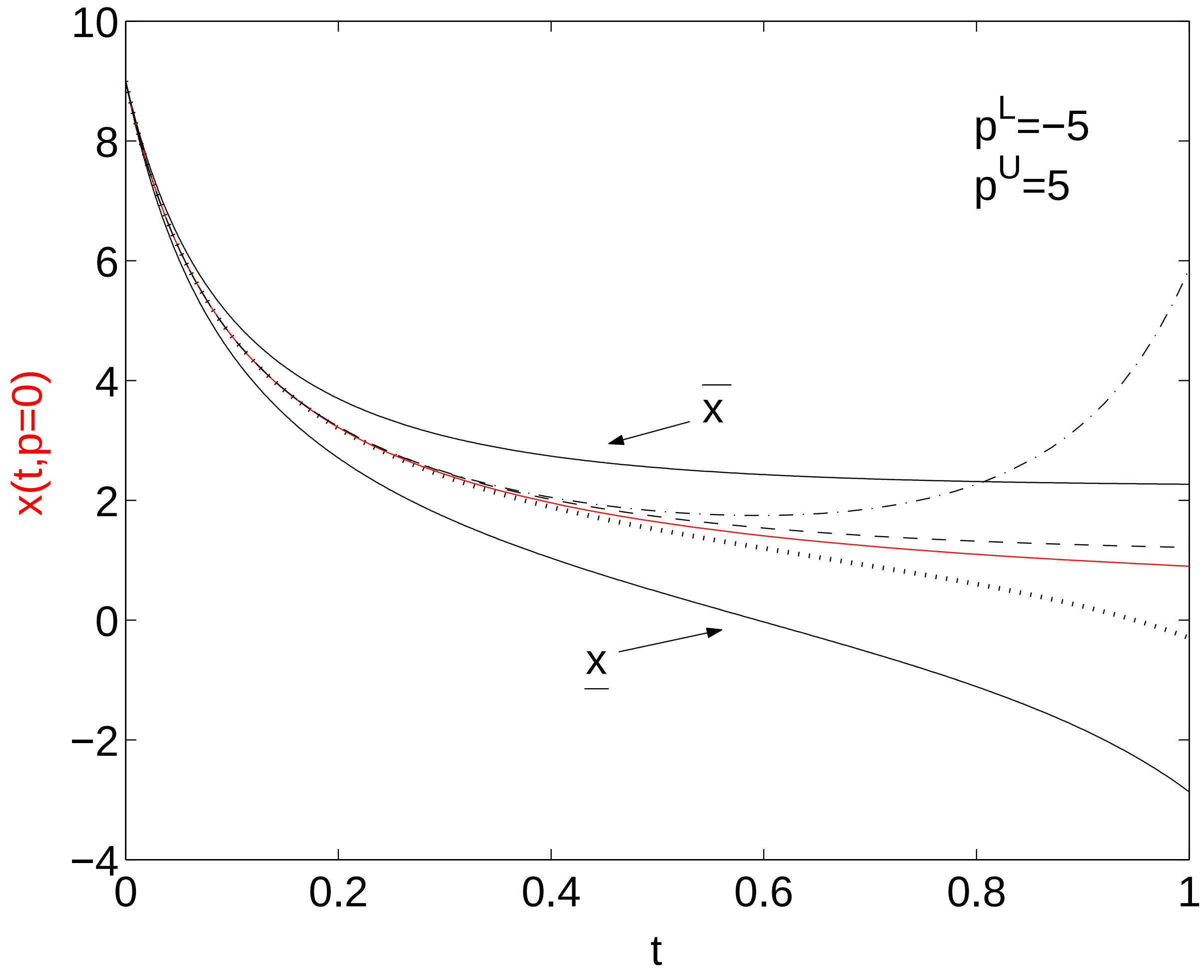
$$\dot{\underline{\underline{x}}} = -(\underline{x} + \overline{x})\underline{\underline{x}} + \underline{x}\overline{x} + v \quad \forall t \in [0, 1]$$
$$\underline{\underline{x}}(0, v) = 9$$

Overestimating IVPs

$$\dot{\overline{\overline{x}}}_1 = -2\underline{x}\overline{\overline{x}}_1 + \underline{x}^2 + v \quad \forall t \in [0, 1]$$
$$\overline{\overline{x}}_1(0, v) = 9$$

$$\dot{\overline{\overline{x}}}_2 = -2\overline{x}\overline{\overline{x}}_2 + \overline{x}^2 + v \quad \forall t \in [0, 1]$$
$$\overline{\overline{x}}_2(0, v) = 9$$

Example: Affine bounds for $p = 0$

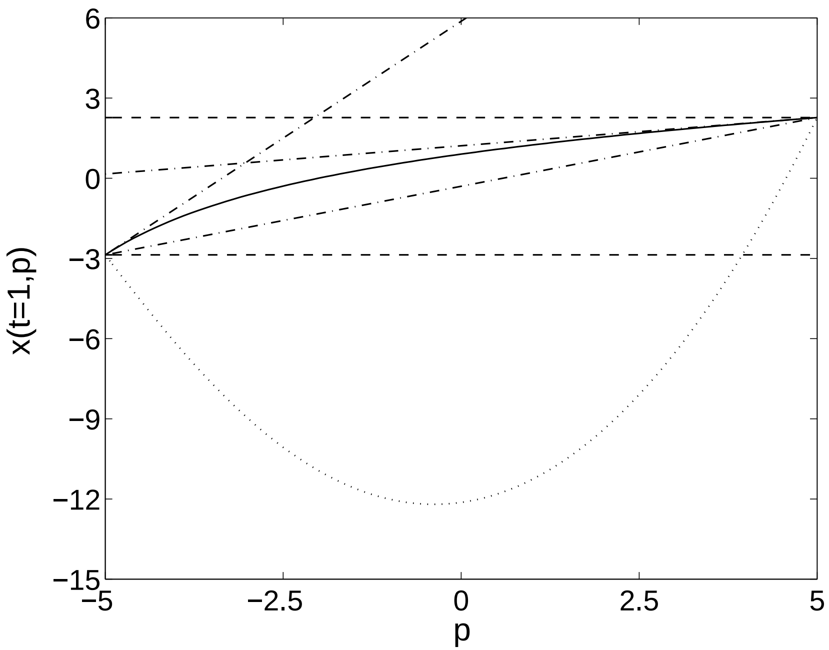


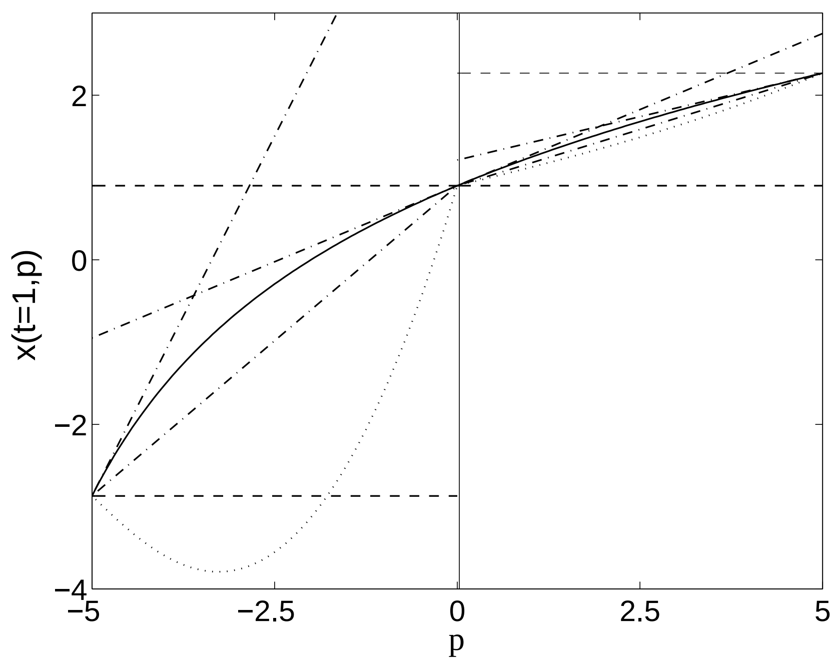
α calculation for $x_k(t_i, p)$

$$[H_{x_k(t_i)}] \ni H_{x_k(t_i)}(p) = \nabla^2 x_k(t_i, p),\ \forall p \in [p^L, p^U] \subset \Re^r$$
$$i = 0, 1, ..., NS,\ k = 1, 2, ..., n$$

1. 1st and 2nd order sensitivity equations
2. Create bounds using Differential Inequalities
3. Construct the interval Hessian matrix
4. Calculate α using the scaled Gerschgorin method

Example: All bounds

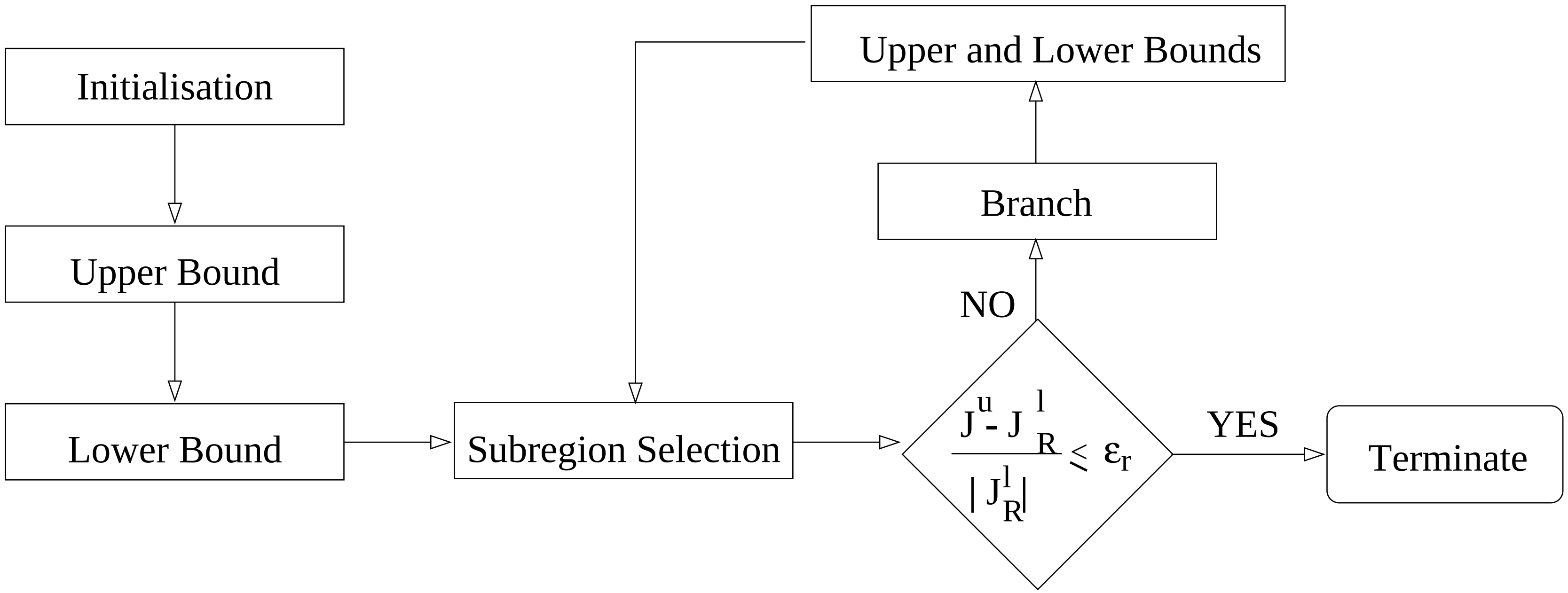




Outline

- Motivation
- Dynamic Optimization: Problem and Methods
- Convex Relaxation of the Dynamic Information
- Deterministic Global Optimization Algorithm
- Convergence of the Algorithm
- Case Studies
- Conclusions and Perspectives

Spatial BB Algorithm (Horst and Tuy, 1996)



Global optimization algorithm

Step 1 Initialization: empty list of subregions, bounds on solution

Step 2 First upper bound calculation (local optimization)

Step 3 First lower bound calculation, including relaxation. Add subregions to list.

Step 4 Subregion selection

- Terminate if list is empty
- Choose region with lowest lower bound otherwise

Step 5 Check for convergence (relative tolerance, max iter)

Step 6 Branch with standard rule (least reduced axis)

Step 7 Upper bound for new regions (not needed at every iteration)

Step 8 Lower bound calculation for each region. Go to Step 4.

Lower bound calculation for region R (Step 8)

Let J^L be the lower bound on the parent region of R.
Let J^U be the best known upper bound.

- Obtain bounds on the differential variables
- If affine bounds are used, obtain necessary matrices
- Form convex relaxation of problem for region R
- If a feasible solution with objective function J_R^L is obtained, then
  - If affine bounds are used and $J_R^L < J^L$, set $J_R^L = J^L$
  - If $J_R^L \leq J^U$, add R to the list of subregions

Outline of proof of convergence

Three main properties are needed:

- Bound improvement after branching
  - Constant bounds improve after branching
  - α-based bounds improve after branching
  - Affine bounds do not improve after branching.
    Ensure improvement through test in Step 8: if $J_R^L < J^L$, set $J_R^L = J^L$
- Bound improving selection operation
  - Consequence of region selection criterion (Step 6)
- Consistent bounding operation
  - Maximum distance between objective function and its relaxation converges to zero.
  - Maximum distance between any constraint and its relaxation converges to zero.

Key elements of proof

Bounds on the solutions of the ODE are such that:

$$\dot{\underline{x}}_k = \text{inf}\, f_k(t, \underline{x}_k, [\underline{x}_{k-}, \overline{x}_{k-}], [p^L, p^U]) \geq \text{inf}\, f_k(t, [\underline{x}, \overline{x}], [p^L, p^U])$$

$$\dot{\overline{x}}_k = \text{sup}\, f_k(t, \overline{x}_k, [\underline{x}_{k-}, \overline{x}_{k-}], [p^L, p^U]) \leq \text{sup}\, f_k(t, [\underline{x}, \overline{x}], [p^L, p^U])$$

$$\forall t \in [t_0, t_{NP}] \text{ and } k = 1, ..., n$$

Inclusion monotonicity of interval operations ensures consistency of bounding operation with constant bounds.

Similar approach can be taken to show α-based underestimators yield a consistent bounding operation:

- Interval Hessian matrix obtained through differential inequalities has desired properties.
- Hence, α and α-based bounds have desired properties.

Implementation of algorithm

- MATLAB 5.3 implementation
- NLPs: Use fmincon function (Optimization Toolbox)
- IVP solution: ode45 (Runge-Kutta based on Dormand-Prince pair)
- Interval calculations: INTLAB with directed outward rounding.
- Runs performed on an Ultra 60 workstation.

Case Study I: *Parameter estimation for* 1^{st} *order reaction* (Tjoa and Biegler, 1991) $A \xrightarrow{k_1} B \xrightarrow{k_2} C$

$$\min_{k_1,k_2} \sum_{j=1}^{10} \sum_{i=1}^{2} (x_i(t_j) - x_i^{exp}(t_j))^2$$

subject to:

$$\begin{aligned} &\dot{x}_1 = -k_1 x_1 \\ &\dot{x}_2 = k_1 x_1 - k_2 x_2 \end{aligned} \quad \forall t \in [0,1]$$
$$\begin{aligned} &x_1(0) = 1 \\ &x_2(0) = 0 \\ &0 \le k_1 \le 10 \\ &0 \le k_2 \le 10 \end{aligned}$$

- Up to 8 affine underestimators and 8 affine overestimators can be constructed.
- α-values ≤ 0.5. Convexity is identified.

Results for case study I

Obj. fun. = 1.1856e-06; $k_1 = 5.0035$; $k_2 = 1.0000$

Underestimation scheme	Branching strategy	ϵ_r	Iter.	CPU time (sec)
C	1	1.00e-02	3,501	2,828
C	1	1.00e-03	34,508	22,959
C & A	1	1.00e-02	37	767
C & A	1	1.00e-03	39	801
C & α	1	1.00e-02	31	396
C & α	1	1.00e-03	35	420
C & α	2	1.00e-02	27	366
C & α	2	1.00e-03	31	407
C & A & α	1	1.00e-02	31	959
C & A & α	2	1.00e-02	27	875

Case Study II: *Parameter estimation for catalytic cracking of gas oil* (Tjoa and Biegler, 1991)

$$
\begin{array}{ccc}
A & \xrightarrow{k_1} & Q \\
k_3 \searrow & & \swarrow k_2 \\
& S &
\end{array}
$$

$$\min_{k_1,k_2,k_3} \sum_{j=1}^{20} \sum_{i=1}^{2} (x_i(t_j) - x_i^{exp}(t_j))^2$$

subject to:

$$
\left.\begin{array}{l}
\dot{x}_1 = -(k_1 + k_3)x_1^2 \\
\dot{x}_2 = k_1 x_1^2 - k_2 x_2
\end{array}\right. \forall t \in [0, 0.95]
$$

$$
\begin{array}{l}
x_1(0) = 1 \\
x_2(0) = 0 \\
0 \leq k_1 \leq 20 \\
0 \leq k_2 \leq 20 \\
0 \leq k_3 \leq 20
\end{array}
$$

Results for case study II

Obj. fun. $= 2.6557e-03$; $k = (12.2141, 7.9799, 2.2215)^T$

Underestimation scheme	Branching strategy	ϵ_r	Iter.	CPU time (sec)
C	1	6.41e-02	10,000	16,729
C	1	1.33e-02	100,000	152,816
C & A	1	1.00e-02	67	26,597
C & A	1	1.00e-03	94	35,478
C & α	1	1.00e-02	73	11,415
C & α	1	1.00e-03	88	13,524
C & α	2	1.00e-02	65	10,116
C & α	2	1.00e-03	81	12,300

32 affine underestimators + 64 affine overestimators

Case Study III: *Parameter estimation for reversible gas phase reaction* (Bellman, 1967):

$$2NO + O_2 \rightleftharpoons 2NO_2$$

$$\begin{array}{ll}
\min\limits_{k_1,k_2} & \displaystyle\sum_{j=1}^{14} (x(t=t_j,k_1,k_2) - x^{exp}(t_j))^2 \\
\text{s.t.} & \dot{x} = k_1(126.2 - x)(91.9 - x)^2 - k_2 x^2 \ \ \forall t \in [0, 39] \\
& x(t=0,k_1,k_2) = 0 \\
& 0 \le k_1 \le 0.1 \\
& 0 \le k_2 \le 0.1
\end{array}$$

x is the difference of the pressure of the system from the initial pressure.

k_1 and k_2 are the rate constants of the forward and reverse reactions.

Results for case study III

Obj. fun.=21.86671; $k_1 = 4.5771\text{e-}06$; $k_2 = 2.7962\text{e-}04$

Underestimation scheme	Branching strategy	ϵ_r	Iter.	CPU time (sec)
constant	1	1.00e-02	75,441	58,513

- only constant bounds used
- 32 affine underestimators + 128 affine overestimators
  - stiff systems, expensive to integrate
  - quality of bounds not better than constant bounds
- quality of α-based bounds poor due to wrapping effect

Case Study IV: *Optimal control with end-point constraint* (Goh and Teo, 1988)

Problem formulation using control vector parameterization:

$$
\begin{array}{rll}
\min\limits_{u_1,u_2} & x_2(t=1,u_1,u_2) & \\
\text{s.t.} & \dot{x}_1 = u_1(1-t) + u_2 t & \forall t \in \\
 & \dot{x}_2 = x_1^2 + (u_1(1-t) + u_2 t)^2 & [0,1] \\
 & x_1(t=0,u_1,u_2) = 1 & \\
 & x_2(t=0,u_1,u_2) = 0 & \\
 & x_1(t=1,u_1,u_2) \geq 1 & \\
 & x_1(t=1,u_1,u_2) \leq 1 & \\
 & -1 \leq u_1 \leq 1 & \\
 & -1 \leq u_2 \leq 1 &
\end{array}
$$

16 affine underestimators + 2 affine overestimators

Results for case study IV

Obj. fun.=9.24242e-01; $u_1 = -0.4545$; $u_2 = 0.4545$

Underestimation scheme	Branching strategy	ϵ_r	Iter.	CPU time (sec)
C	1	1.00e-02	302	317
C	1	1.00e-03	1,062	1,106
C & A	1	1.00e-02	150	2787
C & A	1	1.00e-03	527	9922
C & α	1 or 2	1.12e-13	0	8

α-based bounds recognize convexity of problem at root node.

Conclusions

- *Three types of rigorous convex relaxations* have been developed
- Convergence of the algorithm has been proved
- A BB global optimization algorithm has been applied successfully to case studies in parameter estimation and optimal control
- References:
  - Papamichail and Adjiman, J. Glob Opt, 2002.
  - Papamichail and Adjiman, Comp Chem Eng, 2003.

Perspectives

- Basic theoretical developments of recent years make global optimization of problems with nonlinear IVPs in the constraints possible.
- Practical applicability limited by
  - cost of constructing underestimators and overestimators,
  - quality of estimators for highly nonlinear systems (e.g. oscillatory) and long time horizons.
- Further research needed to
  - identify classes of IVPs for which current estimators are effective,
  - develop new estimators for other problem classes,
  - establish basic theory for DAE systems.