Pitfalls of Computing Enclosures of Overdetermined Interval Linear Systems
Jaroslav Horáček

horacek@kam.mff.cuni.cz

[Charles University in Prague, Czech Republic]
[Department of Applied Mathematics]
+
[University of Economics in Prague, Czech Republic]
[Faculty of Informatics and Statistics]

supervisor: Mgr. Milan Hladík, Ph.D.
Basic notation
An interval linear system and its solution
Methods for solving IOLS
Toolbox LIME 1.0
Conclusions and future
Basic notation

- \( A, b \) indicate an interval matrix and vector respectively.
- \( A, b \) indicate a point real matrix and vector respectively.
- \( A = [\underline{A}, \overline{A}] \), where \( \underline{A} \) is called lower bound and \( \overline{A} \) is called upper bound.
- Also \( A = \langle A_c, A_\Delta \rangle \), where \( A_c \) is midpoint matrix and \( A_\Delta \) is radius matrix.
- It holds \( A_c = (\underline{A} + \overline{A})/2 \).
- It holds \( A_\Delta = (\overline{A} - \underline{A})/2 \).
General interval linear system

Definition

\[ Ax = b, \quad \text{where} \]

- \( A \in \mathbb{IR}^{m \times n} \) (interval matrix)
- \( b \in \mathbb{IR}^{m \times 1} \) (interval vector)
- \( \mathbb{IR} \) is the set of all real closed intervals
Definition

The solution set of $Ax = b$ is

$$\Sigma = \{x \mid Ax = b \text{ for some } A \in \mathcal{A}, b \in \mathcal{b}\}.$$
Solution of interval linear system

- It a polyhedral set
- Not necessarily convex
- But convex in each orthant
Description of a solution

- Difficult to describe
- ⇒ one possibility – a tight n-dimensional box \((\textit{interval hull})\)
- NP-hard (even approximation)
- ⇒ We are looking for a box as narrow as possible containing the hull \((\textit{interval enclosure})\)
Definition

\[ Ax = b, \text{ where} \]

- \( A \in \mathbb{IR}^{m \times n} \), where \( m > n \)
- \( b \in \mathbb{IR}^{m \times 1} \)
- \( \mathbb{IR} \) is the set of all real closed intervals

"More equations than variables."

- More difficult than a square system.
We can adapt many numerical methods to be suitable for computing with intervals.

We usually cannot just replace real variables with interval variables!

The methods will follow now.

The first thing that can come to our mind is the elimination.
We can use elimination
Gaussian elimination (GE)

- Hansen 2006
- Adapted Gaussian elimination - we eliminate to the shape
We have \( m - n + 1 \) equations in the shape \( x_n = [a_i, b_i] \)
We provide intersection
Empty intersection - means no solution
Infinite intersection means infinite solution or overestimation
Then the backward substitution
Ok only for small systems \( n \sim 4 \)
For larger systems we get great overestimation
(Variable|Midpoint|Radius) for a random system $15 \times 13$ with random radii $\leq 10^{-3}$

<table>
<thead>
<tr>
<th>Variable</th>
<th>Midpoint</th>
<th>Radius</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>0.1479</td>
<td>227.6698</td>
</tr>
<tr>
<td>$x_2$</td>
<td>15.2091</td>
<td>172.5929</td>
</tr>
<tr>
<td>$x_3$</td>
<td>11.1031</td>
<td>68.4653</td>
</tr>
<tr>
<td>$x_4$</td>
<td>9.7809</td>
<td>64.1056</td>
</tr>
<tr>
<td>$x_5$</td>
<td>-8.8168</td>
<td>27.2234</td>
</tr>
<tr>
<td>$x_6$</td>
<td>25.8164</td>
<td>25.8398</td>
</tr>
<tr>
<td>$x_7$</td>
<td>-19.0444</td>
<td>30.4596</td>
</tr>
<tr>
<td>$x_8$</td>
<td>-22.0799</td>
<td>11.0313</td>
</tr>
<tr>
<td>$x_9$</td>
<td>1.9649</td>
<td>12.1172</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>-19.1817</td>
<td>11.6841</td>
</tr>
<tr>
<td>$x_{11}$</td>
<td>-20.9670</td>
<td>1.9153</td>
</tr>
<tr>
<td>$x_{12}$</td>
<td>-4.6988</td>
<td>3.5407</td>
</tr>
<tr>
<td>$x_{13}$</td>
<td>3.1223</td>
<td>4.3894</td>
</tr>
</tbody>
</table>
Gaussian elimination (GE) - preconditioning

- For square systems typical preconditioning is $A_c$ the midpoint matrix.
- We tested the preconditioning proposed by Hansen:
  
  $$
  B = \begin{bmatrix}
  A^c_1 & 0 \\
  A^c_2 & I
  \end{bmatrix}.
  $$

- Preconditioning widens the solution $\Rightarrow$ insolvable system becomes solvable.
- When we want to test insolvability of a system we can not use preconditioning (works only for $n \sim 10$).
- After the elimination our mind must be chased by the idea of iteration.
Iterative methods

- For square systems – Jacobi, Gauss-Seidel, Krawczyk, etc. cannot be used for IOLS

- Without preconditioning these methods often have no sense when computing with intervals
(1) After Hansen preconditioning we get almost this shape

\[
\begin{bmatrix}
\sim 1 & \sim 0 & \ldots & \sim 0 \\
\sim 0 & \sim 1 & \ldots & \sim 0 \\
\vdots & \vdots & \ddots & \vdots \\
\sim 0 & \sim 0 & \ldots & \sim 1 \\
\sim 0 & \sim 0 & \ldots & \sim 0 \\
\vdots & \vdots & \ddots & \vdots \\
\sim 0 & \sim 0 & \ldots & \sim 0 \\
\end{bmatrix}
\]

We can use only the upper \( n \times n \) subsquare

Solution enclosure is still rigorous but with loss of information

For matrices with radii close to \( 10^{-2} \) preconditioned matrix often contains zeros on diagonal
(2) do some simple tricks (later)

(3) Rohn method (later)

(4) Transform the known system to the new square one
transform the system to a new square system
The least squares

- Classical $A^T A x = A^T b$ does not work!
- Even $(CA)^T (CA) x = (CA)^T b$ does not work for some preconditioner $C$
- But what works is
  $$
  \begin{pmatrix}
  I & A \\
  A^T & 0
  \end{pmatrix}
  \begin{pmatrix}
  y \\
  x
  \end{pmatrix}
  =
  \begin{pmatrix}
  b \\
  0
  \end{pmatrix},
  $$
- Then we can apply our favourite iterative method
- Problem - for system $50 \times 3$ we have to solve a system $53 \times 53$
- This approach always returns solution!
- Implemented e.g. in Intlab
Subsquare methods

- Simple tricks to enable the use of square iterative methods

- \((1)\) solve some (random?) square subsystems of the overdetermined one and intersect the solutions

- If we check all of them we get something really close to the interval hull and for small systems it is cheaper than linear programming

- Also this way allows to detect insolvability very quickly after few trials random system \(200 \times 170\) with \(r \leq 10^{-2} \sim 5\) steps; with \(r \leq 10^{-4} \sim 2\) steps
Subsquare methods

- **(2)** Use some distinct overlapping subsystems covering the whole IOLS and apply some iterative method to them (Jacobi)

- Intersect all the partial solutions after each iteration - this works quite well

- The more systems we choose, the better
Rohn theorem

Let $Ax = b$ be an IOLS with a solution $S$. Let $R$ be arbitrary real $n \times m$ matrix and let $x_0$ and $d > 0$ are arbitrary $n$-dimensional real vector such that

$$Gd + g < d,$$

where

$$G = |I - RA_c| + |R|A_\Delta,$$

and

$$g = |R(A_c x_0 - b_c)| + |R|(A_\Delta|x_0| + b_\Delta).$$

Then

$$S \subseteq [x_0 - d, x_0 + d].$$
Iterative computing of $d$

\[ x_0 \approx Rb_c, \quad R \approx (A_c^T A_c)^{-1} A_c^T \]

We do not have to use $A_c$ we can use $A \in A \Rightarrow$ iteration

This method works well

However sometimes we cannot find $d$ when some radii are $\leq 0.1$
From wind energy to Linear Programming

Jaroslav Horáček

Pitfalls of Computing Enclosures of OILS
Linear programming

Oettli-Prager theorem

Vector $x \in \mathbb{R}^n$ is a (weak) solution of an interval system if and only if

$$|A_c x - b_c| \leq A_\Delta |x| + b_\Delta.$$

- First absolute value can be rewritten
- Second one with the knowledge of current orthant we are at
- Then we have a system of point real inequalities $\Rightarrow$ Linear programming - we get interval hull
- Problem - we have to solve $2^n \times (2n)$ linear programming problems (a system $15 \times 9 \sim 28$ min)
- Solution - we can solve with some worse method and then compute LP only in the orthant returned
Library of Interval Methods (LIME)

- Matlab / Intlab / Versoft
- Toolbox LIME 1.0
- Documentation (html + sourcecode)
- Free for non-commercial use
Conclusions and future

- Solving of overdetermined interval linear systems is sometimes needed
- Still much can be done in this area - derandomization, theoretical links, new effective methods
- We are currently working on new efficient iterative methods


Thank you very much for your attention.