A global search algorithm for solving systems of non linear polynomial equations

Oliver Salazar

January 4, 2004
Overview

- **Introduction**
  - define problem
  - methods
  - algorithm
  - introductory example

- **Formalization**
  - boxconsistency
  - interval extensions
  - pseudocode
Problems

• Applications in chemistry, economics, engineering,...
Problems

- Applications in chemistry, economics, engineering, ...
- Computationally Complex (NP-hard)
Problems

• Applications in chemistry, economics, engineering,…

• Computationally Complex (NP-hard)

• find all solutions?
Problems

- Applications in chemistry, economics, engineering, ...

- Computationally Complex (NP-hard)

- find all solutions?

- provide proof for
  - uniqueness of solutions?
Problems

• Applications in chemistry, economics, engineering,…

• Computationally Complex (NP-hard)

• find all solutions?

• provide proof for
  – uniqueness of solutions?
  – absence of solutions?
Methods
Methods

• Algebra
  – *Gröbner bases* ⇒ suffer poor scalability
Methods

• Algebra
  – Gröbner bases ⇒ suffer poor scalability
  – Continuation methods ⇒ restrictive application
Methods

- Algebra
  - \textit{Gröbner bases} $\Rightarrow$ suffer poor scalability
  - \textit{Continuation methods} $\Rightarrow$ restrictive application

- Iterative numerical techniques
  - \textit{Newton, bisection} $\Rightarrow$ what to do if no/multiple solution?
Methods

• Algebra
  – Gröbner bases \( \Rightarrow \) suffer poor scalability
  – Continuation methods \( \Rightarrow \) restrictive application

• Iterative numerical techniques
  – Newton, bisection \( \Rightarrow \) what to do if no/multiple solution?

• Interval techniques
  – Newton-like interval methods \( \Rightarrow \) how isolate single root?
Methods

• Algebra
  – Gröbner bases ⇒ suffer poor scalability
  – Continuation methods ⇒ restrictive application

• Iterative numerical techniques
  – Newton, bisection ⇒ what to do if no/multiple solution?

• Interval techniques
  – Newton-like interval methods ⇒ how isolate single root?
  – ⇒ too slow
Solution by Pascal Van Hentenryck

Combine

- consistency technique from AI
Solution by Pascal Van Hentenryck

Combine

• consistency technique from AI
  – discrete combinatorial problems (8-QUEENS)
Solution by Pascal Van Hentenryck

Combine

• consistency technique from AI
  – discrete combinatorial problems (8-QUEENS)
  – eliminate inconsistent values
Solution by Pascal Van Hentenryck

Combine

• consistency technique from AI
  – discrete combinatorial problems (8-QUEENS)
  – eliminate inconsistent values

• intervals for mathematical/numerical correctness
Solution by Pascal Van Hentenryck

Combine

- consistency technique from AI
  - discrete combinatorial problems (8-QUEENS)
  - eliminate inconsistent values

- intervals for mathematical/numerical correctness

Fast algorithm that provides proof for solutions/absence of solutions
Introductory example

\[ x_1^2 + x_2^2 - x_3 = 0 \]
\[ x_1^2 - x_2 = 0 \]
\[ 10x_2 - x_3 = 0 \]

Find values for \( x_1, x_2, x_3 \in \mathbb{R} \)
Introductory example

Transform into an Interval system

\[ X_1^2 + X_2^2 - X_3 = 0 \] \hspace{1cm} (1)
\[ X_1^2 - X_2 = 0 \] \hspace{1cm} (2)
\[ 10X_2 - X_3 = 0 \] \hspace{1cm} (3)

Find canonical intervals for \( X_1, X_2, X_3 \in [-10^8, 10^8] \) by *pruning*
How to prune intervals?

Definition 1. An interval projection constraint $< C, i >$ is the association of an interval constraint $C$ and of an index $i$ ($1 \leq i \leq n$)
How to prune intervals?

Definition 1. An interval projection constraint \( < C, i > \) is the association of an interval constraint \( C \) and of an index \( i \) \((1 \leq i \leq n)\)

Illustration:
interval projection constraints of \([2]\) are

\[
< X_1^2 - X_2 = 0 \ , \ 1 > \\
< X_1^2 - X_2 = 0 \ , \ 2 >
\]
From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  \hspace{2cm} (6)
\[ X_2 = X_1^2. \]  \hspace{2cm} (7)
• From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  \hspace{1cm} (6)

\[ X_2 = X_1^2. \]  \hspace{1cm} (7)

\[ \Rightarrow X_2, X_3 \in [0, 10^8] \]
• From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  
\[ X_2 = X_1^2. \]  

\[ \Rightarrow X_2, X_3 \in [0, 10^8] \]

• From (2)

\[ X_2 = X_1^2. \]
• From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  \hspace{1cm} (6)

\[ X_2 = X_1^2. \]  \hspace{1cm} (7)

⇒ \( X_2, X_3 \in [0, 10^8] \)

• From (2)

\[ X_2 = X_1^2. \]  \hspace{1cm} (8)

⇒ \( X_1 \in [-10^4, 10^4] \)
• From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  \hspace{1cm} (6)
\[ X_2 = X_1^2. \]  \hspace{1cm} (7)

⇒ \( X_2, X_3 \in [0, 10^8] \)

• From (2)

\[ X_2 = X_1^2. \]  \hspace{1cm} (8)

⇒ \( X_1 \in [-10^4, 10^4] \)

• From (1)

\[ X_2^2 = X_3 - X_1^2. \]  \hspace{1cm} (9)
From (1) and (2)

\[ X_3 = X_1^2 + X_2^2 \]  
\[ X_2 = X_1^2. \]  
\[ \Rightarrow X_2, X_3 \in [0, 10^8] \]

From (2)

\[ X_2 = X_1^2. \]  
\[ \Rightarrow X_1 \in [-10^4, 10^4] \]

From (1)

\[ X_2^2 = X_3 - X_1^2. \]  
\[ \Rightarrow X_2 \in [0, 10^4] \]
• From (2)

\[ X_1 = \pm \sqrt{X_2}. \]  

(10)
From (2) we have

\[ X_1 = \pm \sqrt{X_2}. \]  

\[ \Rightarrow X_1 \in [-100, 100]. \]
• From (2)

\[ X_1 = \pm \sqrt{X_2}. \]  \hspace{1cm} (10)

⇒ \( X_1 \in [-100, 100]. \)

• From (3)

\[ X_3 = 10X_2. \]  \hspace{1cm} (11)
• From (2)

\[ X_1 = \pm \sqrt{X_2}. \]  \hspace{1cm} (10)

\[ \Rightarrow X_1 \in [-100, 100]. \]

• From (3)

\[ X_3 = 10X_2. \]  \hspace{1cm} (11)

\[ \Rightarrow X_3 \in [0, 10^5]. \]
• From (2) \[ X_1 = \pm \sqrt{X_2}. \quad (10) \]
\[ \Rightarrow X_1 \in [-100, 100]. \]

• From (3) \[ X_3 = 10X_2. \quad (11) \]
\[ \Rightarrow X_3 \in [0, 10^5]. \]

• From (1) \[ X_2^2 = X_3 - X_1^2 \quad (12) \]
• From (2) \[ X_1 = \pm \sqrt{X_2}. \] (10)
  \( \Rightarrow X_1 \in [-100, 100]. \)

• From (3) \[ X_3 = 10X_2. \] (11)
  \( \Rightarrow X_3 \in [0, 10^5]. \)

• From (1) \[ X_2^2 = X_3 - X_1^2 \] (12)
  \( \Rightarrow X_2 \in [0, \sqrt{10^5}] = [0, 316.227766016]. \)
• From (2) \[ X_1 = \pm \sqrt{X_2}. \] (10) \[ \Rightarrow X_1 \in [-100, 100]. \]

• From (3) \[ X_3 = 10X_2. \] (11) \[ \Rightarrow X_3 \in [0, 10^5]. \]

• From (1) \[ X_2^2 = X_3 - X_1^2 \] (12) \[ \Rightarrow X_2 \in [0, \sqrt{10^5}] = [0, 316.227766016]. \]
• From (2)

\[ X_1 = \pm \sqrt{X_2} \]  \hspace{1cm} (13)
• From (2)

\[ X_1 = \pm \sqrt{X_2} \]  
\[ \Rightarrow X_1 \in \left[ -\sqrt[4]{10^5}, \sqrt[4]{10^5} \right] \]
\[ = [-17.78279410038923, +17.78279410038923] \]

• ...
• From (2)

\[ X_1 = \pm \sqrt{X_2} \]  \hspace{1cm} (13)

\[ \Rightarrow X_1 \in [-\sqrt[4]{10^5}, \sqrt[4]{10^5}] \]

\[ = [-17.78279410038923, +17.78279410038923] \]

• ...
\[ X_1 \in [-3.24876838337, +3.24876838337] \quad (14) \]
\[ X_2 \in [0, 10.27350768179303] \quad (15) \]
\[ X_3 \in [0, 105.5449600878603] \quad (16) \]
Solutions are \((X_1, X_2, X_3) \in \{(0, 0, 0), (-3, 9, 90), (3, 9, 90)\}\).

**Observations**

- no solutions are lost!
Solutions are \((X_1, X_2, X_3) \in \{(0, 0, 0), (-3, 9, 90), (3, 9, 90)\}\). 

**Observations**

- no solutions are lost!
- boundaries are close to solutions!
Key Idea of algorithm

1. preprocess the system until a stable state is reached (Boxconsistency)
Key Idea of algorithm

1. preprocess the system until a stable state is reached (Boxconsistency)

2. if intervals are small enough $\Rightarrow$ solution is found
Key Idea of algorithm

1. preprocess the system until a stable state is reached (Boxconsistency)

2. if intervals are small enough $\Rightarrow$ solution is found

3. otherwise branch
\( x \in [-3.249, +3.249] \)
\( y \in [0, 10.28] \)
\( z \in [0, 105.545] \)

\( x \in [-3.249, 0] \)
\( x \in [0, +3.249] \)

\( x \in [-3, -3] \)
\( y \in [9, 9] \)
\( z \in [89.99999999, 90.00000001] \)
Overview

• Introduction
  – define problem
  – methods
  – algorithm
  – introductory example

• Formalization
  – boxconsistency
  – interval extensions
  – pseudocode
Boxconsistency

First introduced by *Benhamou et al*
Boxconsistency

First introduced by Benhamou et al

**Definition 2.** An interval projection constraint $< C, i >$ is boxconsistent with respect to $\vec{I} = (I_1, \ldots, I_n)$ iff

$$0 \in C(I_1, \ldots, I_{i-1}, \bar{l}, I_{i+1}, \ldots, I_n) \land 0 \in C(I_1, \ldots, I_{i-1}, \bar{r}, I_{i+1}, \ldots, I_n).$$

with $\bar{l}$ the smallest interval enclosing $\text{left}(I_i)$.
and $\bar{r}$ the smallest interval enclosing $\text{right}(I_i)$
How ensure Boxconsistency?

- For each projection constraint
  - project on one variable
How ensure Boxconsistency?

• For each projection constraint
  – project on one variable
  – replace all other variables by their range
How ensure Boxconsistency?

• For each projection constraint
  – project on one variable
  – replace all other variables by their range
  – solve $\exists x_i \in I_i \mid 0 \in F(I_1, \ldots, I_{i-1}, x_i, I_{i+1}, \ldots, I_n)$
  – find leftmost/rightmost zeros
Transformation to interval system

**Definition 3.** $F : \mathcal{I}^n \rightarrow \mathcal{I}$ is an interval extension of $f : \mathbb{R}^n \rightarrow \mathbb{R}$ iff

$$\forall I_1, \ldots, I_n \in \mathcal{I} : r_1 \in I_1, \ldots, r_n \in I_n \Rightarrow f(r_1, \ldots, r_n) \in F(I_1, \ldots, I_n)$$

Not uniquely defined!
Interval Extensions (2)

Example: function $f$

\[
f_1(x_1; x_2) = \frac{x_1 x_2}{1 - x_1} \quad (17)
\]

\[
f_2(x_1; x_2) = \frac{x_2}{\frac{1}{x_1} - 1} \quad (18)
\]
Interval Extensions(2)

Example: function $f$

\[
\begin{align*}
  f_1(x_1; x_2) & = \frac{x_1 x_2}{1 - x_1} \quad (17) \\
  f_2(x_1; x_2) & = \frac{x_2}{1 - x_1} \quad (18)
\end{align*}
\]

Evaluations

\[
F_1([2, 3]; [0, 1]) = \frac{[2,3][0,1]}{1-[2,3]} = [-3, 0] \quad (19)
\]
Interval Extensions\(^2\)

Example: function \( f \)

\[
f_1(x_1; x_2) = \frac{x_1 x_2}{1 - x_1} \quad (17)
\]

\[
f_2(x_1; x_2) = \frac{x_2}{\frac{1}{x_1} - 1} \quad (18)
\]

Evaluations

\[
F_1([2, 3]; [0, 1]) = \frac{[2, 3][0, 1]}{1 - [2, 3]} = [-3, 0] \quad (19)
\]

\[
F_2([2, 3]; [0, 1]) = \frac{[0, 1]}{[2, 3] - 1} = [-2, 0] \neq F_1([2, 3]; [0, 1]) \quad (20)
\]
Interval Extensions(3)
Interval Extensions (4)

Computation of Boxconsistency depends on interval extension

- Natural Interval Extension
Interval Extensions(4)

Computation of Boxconsistency depends on interval extension

- Natural Interval Extension
- Distributed Interval Extension
Interval Extensions (4)

Computation of Boxconsistency depends on interval extension

- Natural Interval Extension
- Distributed Interval Extension
- Taylor Interval Extension
Interval Extensions (4)

Computation of Boxconsistency depends on interval extension

- Natural Interval Extension
- Distributed Interval Extension
- Taylor Interval Extension

project onto one variable
solve \[ \exists x_i \in I_i \mid 0 \in F(I_1, \ldots, I_{i-1}, x_i, I_{i+1}, \ldots, I_n) \]
Natural Interval Extension

Example:

\[ f(x_1, x_2) = x_1^3 + x_2 \iff F(X_1, X_2) = X_1^3 + X_2 \]
Natural Interval Extension

Example:

\[ f(x_1, x_2) = x_1^3 + x_2 \Leftrightarrow F(X_1, X_2) = X_1^3 + X_2 \]  \hspace{1cm} (21)

Boxconsistency:

- project onto one variable
Natural Interval Extension

Example:

\[ f(x_1, x_2) = x_1^3 + x_2 \iff F(X_1, X_2) = X_1^3 + X_2 \] (21)

Boxconsistency:

- project onto one variable
  - Apply Interval Newton method for finding zeros
Natural Interval Extension

Example:

\[ f(x_1, x_2) = x_1^3 + x_2 \Leftrightarrow F(X_1, X_2) = X_1^3 + X_2 \]  \hspace{1cm} (21)

Boxconsistency:

- project onto one variable
  - Apply Interval Newton method for finding zeros
  - combine with bisection
Distributed Interval Extension (1)

Example:

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]
Distributed Interval Extension (1)

Example:

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  

(22)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1X_2 - 4 \]  

(23)
Distributed Interval Extension (1)

Example:

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (22)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1X_2 - 4 \]  \hspace{1cm} (23)

Boxconsistency:

- project on one variable
Distributed Interval Extension (1)

Example:

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (22)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1X_2 - 4 \]  \hspace{1cm} (23)

Boxconsistency:

- project on one variable
- sandwich \( f \) between upper/lower function \((f_u, f_l)\)
Distributed Interval Extension (1)

Example:

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (22)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1 X_2 - 4 \]  \hspace{1cm} (23)

Boxconsistency:

- project on one variable

- sandwich \( f \) between upper/lower function \((f_u, f_l)\)

- find leftmost/rightmost zeros of these functions
Distributed Interval Extension (2)

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]
Distributed Interval Extension (2)

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (24)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1X_2 - 4 \]  \hspace{1cm} (25)
Distributed Interval Extension (2)

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (24)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1X_2 - 4 \]  \hspace{1cm} (25)

with \( X_1, X_2 = [0, 1] \)
Distributed Interval Extension (2)

\[ f(x_1, x_2) = x_1(x_1 + x_2) - 4 \]  \hspace{1cm} (24)

Transform into

\[ F(X_1, X_2) = X_1^2 + X_1 X_2 - 4 \]  \hspace{1cm} (25)

with \( X_1, X_2 = [0, 1] \)

projecting \( F \) onto \( X_1 \)

\[ F_p(X) = X^2 + [0, 1] X - 4 \]  \hspace{1cm} (26)
Distributed Interval Extension (3)
from (26) the functions $f_l$ and $f_u$ constructed

\[ f_u(x) = x^2 - 4 \]
\[ f_l(x) = x^2 + x - 4 \]
from (26) the functions \( f_l \) and \( f_u \) constructed

\[
\begin{align*}
  f_u(x) &= x^2 - 4 \\
  f_l(x) &= x^2 + x - 4
\end{align*}
\]  \tag{27} \tag{28}

with their Natural Interval Extensions

\[
\begin{align*}
  F_u(X) &= X^2 - 4 \\
  F_l(X) &= X^2 + X - 4
\end{align*}
\]  \tag{29} \tag{30}
Distributed Interval Extension (4)

Advantages

• easy to calculate these upper and lowerbound functions
Distributed Interval Extension (4)

Advantages

• easy to calculate these upper and lowerbound functions

• effective pruning (numbers, no intervals)
Distributed Interval Extension (4)

Advantages

• easy to calculate these upper and lower bound functions

• effective pruning (numbers, no intervals)

• increase precision
Taylor Interval Extension

The Taylor interval extension transforms the function into Taylor form.

**Definition 4.** Let \( f : \mathbb{R}^n \to \mathbb{R} \) be a function of the form \( f = 0 \) and have continue partial derivatives. Let \( \vec{I} \) be an interval vector \((I_1, \ldots, I_n)\) and \( m_i \) be the center of \( I_i \). The Taylor interval extension of \( f \) developed around \( \vec{C} = (m_1, \ldots, m_n) \) is

\[
F(m_1, \ldots, m_n) + \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)(X_i - m_i) = 0.
\]
Taylor Interval Extension: Boxconsistency

projection of

\[ F(\bar{m}_1, \ldots, \bar{m}_n) + \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)(X_i - \bar{m}_i) \]  

onto \( X_i \)
Taylor Interval Extension: Boxconsistency

projection of

\[ F(\overline{m}_1, \ldots, \overline{m}_n) + \sum_{i=1}^{n} \frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)(X_i - \overline{m}_i) \]  \hspace{1cm} (31)

onto \( X_i \)

\[ F(\overline{m}_1, \ldots, \overline{m}_n) + \sum_{j=1}^{i-1} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \overline{m}_j) + \]
\[ \frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)(X_i - \overline{m}_i) + \sum_{j=i+1}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \overline{m}_j) \]  \hspace{1cm} (32)
Solve to $X_i$

$$X_i = m_i - \frac{1}{\partial F/\partial x_i(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - m_j)$$

$$+ F(m_i, \ldots, m_n) \quad (33)$$
Solve to $X_i$

$$X_i = m_i - \frac{1}{\frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - m_j)$$

$$+ F(m_i, \ldots, m_n) \quad (33)$$

- no overestimation (centered form)
Solve to $X_i$

$$X_i = \overline{m}_i - \frac{1}{\frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \overline{m}_j)$$

$$+ F(\overline{m}_i, \ldots, \overline{m}_n) \quad (33)$$

- no overestimation (centered form)
- weak pruning on large intervals
Solve to $X_i$

\[
X_i = \bar{m}_i - \frac{1}{\frac{\partial F}{\partial x_i}(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \bar{m}_j) \]

\[+ F(\bar{m}_i, \ldots, \bar{m}_n) \tag{33} \]

- no overestimation (centered form)
- weak pruning on large intervals
- powerful pruning on small intervals
Solve to $X_i$

$$X_i = \bar{m}_i - \frac{1}{\partial F/\partial x_i(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \bar{m}_j)$$

$$+ F(\bar{m}_i, \ldots, \bar{m}_n) \quad (33)$$

- no overestimation (centered form)
- weak pruning on large intervals
- powerful pruning on small intervals
- exact range $\Rightarrow$ proof for solutions!
Solve to $X_i$

\[
X_i = \bar{m}_i - \frac{1}{\partial F/\partial x_i(I_1, \ldots, I_n)} + \sum_{j=1, j \neq i}^{n} \frac{\partial F}{\partial x_j}(I_1, \ldots, I_n)(I_j - \bar{m}_j) + F(\bar{m}_i, \ldots, \bar{m}_n)
\]  

(33)

- no overestimation (centered form)
- weak pruning on large intervals
- powerful pruning on small intervals
- exact range $\Rightarrow$ proof for solutions!
Pseudocode

```
Func. Search(S: Set of Constraints; \vec{I}_0 : intervals ∈ \mathcal{I}^n): Set of \mathcal{I}^n
Begin
    \vec{I} := PRUNE(S, \vec{I}_0);
    if ¬ IsEmpty(\vec{I});
        if IsSmallEnough(\vec{I}) then
            return \{\vec{I}\};
        else
            < \vec{I}_1, \vec{I}_2 > := BRANCH(\vec{I});
            return Search(S, \vec{I}_1) ∪ Search(S, \vec{I}_2)
        endif
    else
        return ∅
End
```
Func. PRUNE($S$: Set of Constraints; $\vec{I}$: intervals $\in \mathcal{I}^n$)

Begin

repeat

$\vec{I}_p = \vec{I}$

BOXPRUNE(NE($S_E$) $\cup$ DE($S_E$), $\vec{I}$);
BOXPRUNE(TE($S_E$), $\vec{I}$);

until $\vec{I} = \vec{I}_p$

End

with $S_E = \{(c, i) | c \in S \text{ and } 1 \leq i \leq n\}$
Conclusion

• Benchmarks show fast results
Conclusion

• Benchmarks show fast results
  – competes well with state of the art continuing methods
  – outperforms traditional interval methods
Conclusion

- Benchmarks show fast results
  - competes well with state of the art continuing methods
  - outperforms traditional interval methods
  - Broyden Banded functions
    for 320 variables are solved in 150 seconds (linear!)

\[
f_i(x_1, \ldots, x_n) = x_i(2 + 5x_i^2) + 1 - \sum_{j \in J_i} x_j(1 + x_j) \quad (1 \leq i \leq n)
\]

with \( J_i = \{j \mid j \neq i \text{ and } \max(1, i - 5) \leq j \leq \min(n, j + 1)\} \)

and \( x_i = [-10^8, 10^8] \)
• proof for solution
• proof for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
• **proof** for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
  – Taylor interval extension: close to solution
• **proof** for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
  – Taylor interval extension: close to solution

• further active interval research
  – how solve overestimation of range?
• **proof** for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
  – Taylor interval extension: close to solution

• further active interval research
  – how solve overestimation of range?
  – does there exist an extension that combines
    * powerfull pruning?
• **proof** for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
  – Taylor interval extension: close to solution

• further active interval research
  – how solve overestimation of range?
  – does there exist an extension that combines
    * powerful pruning?
    * no overestimation?
• **proof** for solution

• combination of extensions seem to provide substantial pruning
  – distributed interval extension: far from solution
  – Taylor interval extension: close to solution

• further active interval research
  – how solve overestimation of range?
  – does there exist an extension that combines
    * powerfull pruning?
    * no overestimation?
References


