SONIC
A Solver and Optimizer for Nonlinear Problems based on Interval Computations

Lars Balzer
University of Wuppertal

September 23, 2014
## Contents

1. SONIC
2. Taylor models
3. Inversion & Newton method
4. Outlook
What is SONIC

SONIC – A Solver and Optimizer for Nonlinear Problems based on Interval Computations

- rigorous solver for nonlinear systems
- developed at RWTH Aachen and University of Wuppertal
- initially for dynamic systems in chemical processes
Features

- written in C++
- portability and performance by different interval libraries: choice between C-XSC, filib++
- parallel versions for distributed and shared memory (MPI, OpenMP)
- solver for optimization problems
- handling of unbounded intervals
Branch and Bound

startbox $x^{(0)}$

- Investigate further
- Solution box
- Contains no solution
Subdivision

- subdivision using gaps caused by division
- choice of subdivision direction:
  - subdivide direction with largest width
  - search zeros near boundary
  - search direction with highest influence on function value
  - hybrid strategy
- choice of subdivision point:
  - bisect at midpoint
  - shift bisection point
- iterated subdivision
Contractors

Aim: Contract or discard a given box

- Constraint propagation
- Taylor refinement
- Interval Newton method
- Taylor models
Consider system

\[ f_1(x) = x_1^2 - x_2 = 0 \]
\[ f_2(x) = x_1^2 + x_2^2 - 1 = 0 \]

with \( x \in [-2, 2]^2 \)
Constraint propagation

Forward sweep corresponds to the natural interval evaluation
Constraint propagation

Backward sweep yields narrower bounds:

\[ 0 = f_1 = x_3 - x_2 \]
\[ \iff x_3 = f_1 + x_2 \in (0 + [-2, 2]) \cap [0, 4] = [0, 2] \]

\[ \ldots \]

\[ x_1 = \pm \sqrt{x_3} \in [-1, 1] \]
\[ x_2 = \pm \sqrt{x_4} \in [0, 1] \]
Taylor refinement

Gain sharper enclosure by Taylor expansion for $f : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}$:

$$f(x) \subseteq f(c) + \sum_{i=1}^{n} s_i \cdot (x_i - c_i)$$

$$x'_i := x_i \cap \left( c_i - \left( f(c) + \sum_{\substack{j=1\atop j \neq i}}^{n} s_j \cdot (x_j - c_j) \right) \oslash s_i \right), \quad i = 1 \ldots n$$

with some center $c \in x$ and slope vector $s$
Interval Newton method

Refines or discards box

\[ f(x) \subseteq f(c) + S(x - c) \]

\[-f(c) = S(x^* - c)\]

- solve with Gauß-Seidel-approach
- several preconditioners available
- can be used to verify the existence of a (unique) solution
Algorithm 1 Application scheme for contractors (box \( x \))

1: if CP is enabled then apply CP
2: if \( x \) cannot contain a solution then discard \( x \)
3: if Taylor refinement of first order is enabled then
4:    apply Taylor refinement of first order
5:    if CP is enabled then apply CP
6: end if
7: if \( x \) cannot contain a solution then discard \( x \)
8: if Taylor refinement of second order is enabled then
9:    apply Taylor refinement of second order
10: if CP is enabled then apply CP
11: end if
12: if \( x \) cannot contain a solution then discard \( x \)
13: if Newton method is enabled then
14:    apply Newton method
15: if CP is enabled then apply CP
16: end if
17: if \( x \) cannot contain a solution then discard \( x \)
Verification

Aim: Prove existence or uniqueness of a solution in a given box

- **Newton test**
  - apply one step of the interval Newton method
  - proofs uniqueness

- **Miranda test**
  - compare sign of function values on opposite facets

- **Borsuk test**

- **topological degree test**
Problem: Too many boxes...

- Many boxes → slow
- No discarding of boxes

Diagram: A tree structure with many levels and boxes, indicating the complexity and slow performance due to the large number of boxes.
Problem: Too many boxes...

- Taylor models
- discard boxes
- fewer boxes left to investigate
Taylor models

Idea: Enclosure of a function over its course
based on the works by Berz, Hoefkens and Makino

Consists of

- domain $\mathcal{D} \subseteq \mathbb{R}^v$
- reference point $x_0 \in \mathcal{D}$
- Taylor polynomial $P_n : \mathcal{D} \to \mathbb{R}^w$ of order $n$
- remainder interval $R \subseteq \mathbb{I}^w$

Inclusion property for all $x \in \mathcal{D}$:

$$f(x) \in P_n(x) + R$$
Function enclosures

Enclosure by a box ...  

... and a Taylor Model
Properties of Taylor models

- sharper enclosure over relatively large domain ⇒ avoids subdivisions
- the remainder interval scales with the \((n + 1)\)-st order of the magnitude of \(D\)
- reduced impact of the dependency problem
- number of coefficients for a polynomial of order \(n\) with \(v\) variables: \(\frac{(n+v)!}{n! \cdot v!}\)
Taylor models in SONIC

Usage of Taylor models as a contractor in SONIC:

Algorithm 2 Usage of Taylor models as a contractor (box $x$)

1: build Taylor model $T$ over $x$
2: compute evaluation $B(T)$ of $T$
3: if $0 \notin B(T)$ then
4: discard $x$
5: end if
Results

Observations for the naive usage of Taylor models:

- computation time increases significantly
- most time is consumed on the buildup of the TM
- very few boxes get discarded
- natural function evaluation of Taylor polynomial causes overestimations
  → linear dominated bounder
Inverse Taylor models

Taylor models \( T = (P_n, x_0, \mathcal{D}, R) \) and \( S = (G_n, y_0, \Delta, \Omega) \)

\( S \) left-inverse Taylor model for \( T \) if

- \( G_n \circ P_n = n \mathcal{I} \)
- \( P_n(x_0) = y_0 \)
- \( f \in T \Rightarrow f(\mathcal{D}) \subset \Delta \)
- \( B(G_n(T) - I) \subset \Omega \)
Theorem

Let $T = (P_n, x_0, D, R)$ and $S = (G_n, y_0, \Delta, \Omega)$ be given Taylor models such that $S$ is a left-inverse Taylor model for $T$. Assume that $f \in T$ is invertible over $D$.

Then there is $g \in S$ such that $g = f^{-1}$ on $f(D)$. 

Newton method with inverse Taylor models

Algorithm 3 Newton method with taylor models ()

1: \( k := 0 \)
2: repeat
3:   build \( n \)-th order Taylor model \( T^{(k)} \) over \( D^{(k)} \) with reference point \( \text{mid}(D^{(k)}) \)
4:   compute left-inverse Taylor model \( S^{(k)} \) for \( T^{(k)} \)
5:   \( D^{(k+1)} := D^{(k)} \cap S^{(k)}(0) \)
6:   if \( D^{(k+1)} = \emptyset \) then
7:     discard box
8:   end if
9:   \( k := k + 1 \)
10: until \( D^{(k)} \) accurate enough
Properties of the inverse TM Newton method:

- faster speed of convergence
- larger domain of convergence
- less domain splitting required
- less impact of dependency problem and cancellation errors
Future plans

- web interface
- graphical output
Contact

For questions write to

balzer@math.uni-wuppertal.de
Bibliography

Elke Just.  
*Adaptive use of extended systems for the efficient verified solution of nonlinear systems.*  
Dissertation, University of Wuppertal, 2013.

Jens Hoefkens.  
*Rigorous numerical analysis with high-order Taylor Models.*  