Current Possibilities for Simulating Uncertain Non-Smooth Dynamic Systems

Ekaterina Auer

University of Technology, Business and Design Wismar

July 14, 2015
Non-smooth Models: Phenomena and Areas

**Friction**

**Impacts**

**Hysteresis**

Non-smooth Models: Further Details

Areas: Control, biology, economics, material science, ...

Phenomena: Saturation, “good” numerical behavior, switchings...

Formalisms: differential inclusions, Moreau’s sweeping process, ...

Similarities: \(m\ddot{x} + h \cdot \text{sign}(x) = 0\)

sliding pendulum

relay
Impact-like Problems

Impact

A high force applied over a short time period when two or more bodies collide

Mechanics: Collision of rigid bodies
Goal: model and simulate changes in the motion of two solid bodies following collision
Impact-like Problems

Impact

A high force applied over a short time period when two or more bodies collide

Mechanics: Collision of rigid bodies
Goal: model and simulate changes in the motion of two solid bodies following collision

Electrical engineering: Ideal diodes

The current $i(t)$ and the voltage $v(t)$ satisfy the complementarity conditions: $0 \leq i(t) \perp v(t) \geq 0$
Impact-like Problems

Impact

A high force applied over a short time period when two or more bodies collide

Mechanics: Collision of rigid bodies
Goal: model and simulate changes in the motion of two solid bodies following collision

Electrical engineering: Ideal diodes

The current $i(t)$ and the voltage $v(t)$ satisfy the complementarity conditions: $0 \leq i(t) \perp v(t) \geq 0$

Models: E.g., differential inclusions

$$
\psi_K(x) = \begin{cases} 
0 & \text{if } x \in K \\
+\infty & \text{if } x \notin K
\end{cases}, \quad \partial \psi_{\mathbb{R}^+}(x) = \begin{cases} 
\{0\} & \text{if } x > 0 \\
(-\infty, 0] & \text{if } x = 0
\end{cases}, \quad i(t) \in -\partial \psi_{\mathbb{R}^+}(v(t))
$$
Friction

The force resisting the relative motion of e.g. solid surfaces sliding against each other (\(\sim\) dry friction resists relative lateral motion of two solid surfaces in contact).

**Mechanics:** A block which slides or sticks on the table

**Non-smooth approach:**

\[
\begin{align*}
\text{slip} & \quad m\ddot{x} = F + \lambda_T \\
\lambda_T & = \pm \mu m g
\end{align*}
\]

\[
\begin{align*}
\text{stick} & \quad m\ddot{x} = F + \lambda_T \\
\dot{x} & = 0
\end{align*}
\]

\[-\lambda_T \in \mu m g \text{ Sgn}(\dot{x})\]
Friction

The force resisting the relative motion of e.g. solid surfaces sliding against each other (dry friction resists relative lateral motion of two solid surfaces in contact).

Mechanics: A block which slides or sticks on the table

Electrical engineering: Ideal Zener diode

Non-smooth approach:

Slip:
\[ m\ddot{x} = F + \lambda_T \]
\[ \lambda_T = \pm \mu m g \]

Stick:
\[ m\ddot{x} = F + \lambda_T \]
\[ \dot{x} = 0 \]

Allows current to flow in the forward direction, but also permits it to flow in the reverse direction when the voltage is above a certain value known as the breakdown voltage.
Friction

The force resisting the relative motion of e.g. solid surfaces sliding against each other (\(\rightsquigarrow\) dry friction resists relative lateral motion of two solid surfaces in contact).

**Mechanics:** A block which slides or sticks on the table

\[
\begin{align*}
F &= mg - \lambda_T \\
\lambda_T &= \pm \mu mg
\end{align*}
\]

**Non-smooth approach:**

\[
\begin{align*}
-m \ddot{x} &= F + \lambda_T \\
\lambda_T &= \pm \mu mg
\end{align*}
\]

\(-\lambda_T \in \mu mg \text{ Sgn}(\dot{x})\)

**Electrical engineering:** Ideal Zener diode

Allows current to flow in the forward direction, but also permits it to flow in the reverse direction when the voltage is above a certain value known as the breakdown voltage

**Models:** E.g., Coulomb friction with \(\text{Sgn}(x) = \partial|x| = \begin{cases} 1 & \text{if } x > 0 \\ [ -1, 1] & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}\)
Hysteresis

The time-based dependence of a system’s output on current and past inputs ( reflux “loop”).

Mechanics: Rubber band

The behavior as a load is removed is not the same as that when the load is being increased.
Hysteresis

The time-based dependence of a system’s output on current and past inputs (\(\leadsto\) “loop”).

**Mechanics:** Rubber band

The behavior as a load is removed is not the same as that when the load is being increased.

**Electrical engineering:** Schmitt trigger

The output retains its value until the input changes sufficiently to trigger a change.
Hysteresis

The time-based dependence of a system’s output on current and past inputs (\(\sim\) “loop”).

**Mechanics:** Rubber band

The behavior as a load is removed is not the same as that when the load is being increased.

**Electrical engineering:** Schmitt trigger

The output retains its value until the input changes sufficiently to trigger a change.

**Models:** Specific to the area, e.g., Bouc-Wen model with the hysteretic displacement

\[
\dot{z}(t) = \dot{u}(t) \{ A - [\beta \text{sign}(z(t)\dot{u}(t)) + \gamma] |z(t)|^n \}
\]
Non-smooth Models: Code Angle

<table>
<thead>
<tr>
<th>Construct</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF-THEN-ELSE</td>
<td>Force: $F \leq 0$</td>
</tr>
<tr>
<td>SWITCH</td>
<td>Muscle activation function:</td>
</tr>
<tr>
<td></td>
<td>$0 \leq a(t) = A_1 e^{-c_1(t-t_1)} + A_2 e^{-c_2(t-t_2)} \leq 1$</td>
</tr>
<tr>
<td>$</td>
<td>x</td>
</tr>
</tbody>
</table>
|                 | $\dot{\omega}(t) = \rho \cdot \left( v(t) - \sigma \cdot |v(t)| \cdot |\omega(t)|^{\nu-1} \cdot \omega(t) \\
|                 | $+ (\sigma - 1) \cdot v(t) \cdot |\omega(t)|^{\nu} \right)$ |
| sign$x$         | Friction: $F(v) = \text{sign}(v) \cdot F + \mu \cdot v$ |
**Mathematical Formalisms**

**Automaton** (causal): a graph or an automaton with different ODEs as vertices and logical conditions for jumps as edges

**Closed-expression**: (non-causal): a single system of ordinary or implicit D(A)Es or inequalities that changes its right side in dependence on zeros of a certain $g$

---

**Necessary:** A study of relationships between modeling concepts

**Example:** a system with Coloumb and viscous friction

**Model:**\[ m\ddot{x}(t) + c\dot{x}(t) + kx(t) \in -\partial(\mu|\dot{x}|) \]

**Formalisms:** DI, variational inequality, interval, ...
Possible Interval Reformulations: Example

\[ \dot{x}(t) = x(t)u(t), \ x(0) = 0, \ u(t) \in [-1, 1] \text{ unknown, smooth} \]

### Possibility  
### Formulation  
### Solution

1. Solve the IVP  
   
   \[ x(t) = x_0 \cdot e^{\int_0^t u(s)ds} \]
   
   for \( x_0 \neq 0 \)
   
   or \( x(t) \equiv 0 \)

2. Consider a DI  
   \[ \dot{x}(t) \in [-x(t), x(t)] \]
   
   \[ x(t) = 0 \text{ and e.g.} \]

   \[ x(t) = \begin{cases} 
   0, & 0 \leq t \leq 2 \\
   t^2, & t \geq 2 
   \end{cases} \]

3. Use intervals  
   \[ \dot{x}(t) = x(t) \cdot [-1, 1] \]
   
   \( x(t) = x_0 \cdot e^{[-1,1]t} \) for \( x_0 \neq 0 \)
   
   or \( x(t) \equiv 0 \)

Intervals offer help in solving DE with convex and closed set-valued right sides

E. Auer  
University of Technology, Business and Design Wismar

Uncertain Non-Smooth Dynamic Systems
Let $\dot{x}(t) = x(t)u(t)$, $x(0) = 1$, $u(t) = \cos t \in [-1, 1]$

"Normal": $x(t) = 1 \cdot e^{\int_0^t \cos s \, ds} = e^{\sin t}$

DI $x(t) = \begin{cases} 1, & 0 \leq t \leq 2 \\ t^2, & t \geq 2 \end{cases}$

"Interval": $x(t) = x_0 \cdot e^{[-1,1]t} = [e^{-t}, e^t]$, because $e^t$ is monotone

Interval solution is wide but encloses both of the other solutions!

Systematize formalisms/applications, assign a verified method, introduce a simple way of analysis
Verified Methods for Non-smooth Systems

Description of a non-smooth IVP

Closed expressions

Automata

\[ x' = \begin{cases} 
  f^-(x), & h(x(t), t) < 0 \\
  f^+(x), & h(x(t), t) > 0 
\end{cases} \]

Rihm (1993), Mahmoud and Chen (2008)


Verified non-smooth optimization: Slopes, generalized gradients ...

Ratz (1995), Kearfott (1996), Schnurr (2007), ...

Rapid (1995), Kearfott (1996), Schnurr (2007), ...
Task: Solve the Non-smooth IVP

\[ \dot{x} = f(x, t), \ x(0) = x_0, \ \text{where } f(x, t) \ \text{is non-smooth in } x \ (\text{or in } t). \]

**Situation 1:** \( f \) is discontinuous only in \( t \)

\[ x(t) = x_0 + \int_0^t f(x(s), s) \, ds \]

**Situation 2:** \( f \) is discontinuous in \( x \): more difficult

- Problem reformulation
- Solution definition (allowed to be discontinuous?)
- Existence (uniqueness) of the solution
- Application areas
Rihm’s method

Reformulation
\[ \dot{x}(t) = f(x(t), t) = \begin{cases} f_1(t, x(t)), g(t, x(t)) < 0 \\ f_2(t, x(t)), g(t, x(t)) > 0 \end{cases} \]

Application
systems with friction, switchings

Solution
(a) a continuous function for which IVP holds except at isolated switching points
(b) Filippov’s convex definition for DIs if switching points are not isolated

Existence
(a) unique if transversality conditions hold
(b) exists if the function \( f_0 = \alpha \cdot f_1 + (1 - \alpha) f_2 \) is cont. in \( x \); unique if \( f_0 \) cont. differentiable
Rihm’s method: Enclosure over the Switching Point

Transversality
\[ \dot{g}_1(t, x) := \frac{\partial g}{\partial t} + \frac{\partial g}{\partial x} f_1(t, x) > 0 (< 0) \]
\[ \dot{g}_2(t, x) := \frac{\partial g}{\partial t} + \frac{\partial g}{\partial x} f_2(t, x) > 0 (< 0) \]

Endpoints
\[ \dot{g}_1(t, x) < 0 \text{ and } \dot{g}_2(t, x) > 0 \]

Sliding
\[ \dot{g}_1(t, x) > 0 \text{ and } \dot{g}_2(t, x) < 0 \]

Let \( t^* \) be the switching point, \( x^* := x(t^*), \)
\[ f^- := f_1(t^-, x(t^-)), \]
\[ f^+ := f_2(t^*, x^*), \]
\[ h^- := t^* - t^-, \]
\[ h^+ := t^+ - t^*, \]
\[ s = t^+ - t^- \]
\[ z^- \in \mathbf{z}^- \text{ and } z^+ \in \mathbf{z}^+ \text{ local errors, then} \]
\[ x(t^+) = x(t^-) + h^+(f^+ - f^-) + sf^- + z^- + z^+ \]
\[ \in x^- + sf^- + [0, s](f^+ - f^-) + z^- + z^+ =: \mathbf{x}^+ \]
Rihm’s method: Algorithm

1. Prepare: \(0 \in g(t_0, [x_0])\)? (Y) Check transversality, reformulate (N) Proceed

2. Enclose in the area of continuity: Define a grid \(t_0, t_1, \ldots, t_{r_s}\)
   
   Compute a rough enclosure \(x_{j-1,j}\) over \([t_{j-1}, t_j]\)
   
   Refine into \(x_j\) of \(x_1(t_j)\) until \(j = r_s\)
   
   or \((t^*, x^*)\) is reached (check with \([t_{j-1}, t_j], x_{j-1,j}\))

3. Enclose \(t^*\): Use the interval Newton method: \(t^* \in I^* := [t^-, t^+], h = \text{wid}(I^*)\)

4. Enclose \(x^*\): Compute \(x(t)\) over \([t_{j-1}, t_j]\) (e.g. from Taylor coeff.)
   
   Compute a rough enclosure
   
   \(x^* = x_{j-1,j} \cap (x(t^-) + [0, h]f_1(I^*, x_{j-1,j})) \cap (x(t^+) - [0, h]f_1(I^*, x_{j-1,j}))\)
   
   Refine if possible, e.g. using \(g(t, x)\); check if \((t^*, u^*)\) is an end point

5. Continue into the next cont. area: Compute rough enclosure \(x^+\) and local errors \(z^-, z^+\)
   
   Compute the refined enclosure of \(x(t^+)\) (previous slide)
   
   Reformulate and go to 2
An Automaton Based Method (Rauh et al.)

Problem: Smooth models \( \{S_i\}_{i=1}^{l} : \dot{x}(t) = f_{S_i}(x(t), p, u(t), t) \)

Transition \( S_i \rightarrow S_j \) if the condition \( T_{ij}^j(x, u) \) holds true

Stage 1: Calculate a bounding box

\[
b_{b_k}^a = \bigcup_{i \in I_a} (x_0 + [0, h] \cdot f_{S_i}(x_k, p, u(t_k), t_k))
\]

Stage 2: Activate additional transitions \( T_{ij}^j(b_{b_k}^a, u([t_k, t_{k+1}])) \)

\[
\tilde{b}_{b_k}^a := b_{b_k}^a \bigcup_{i \in \tilde{I}_a \setminus I_a} (x_0 + [0, h] \cdot f_{S_i}(b_{b_k}^a, p, u([t_k, t_{k+1}]), [t_k, t_{k+1}]))
\]

Stage 3: Calculate \( x_{k+1} \) at \( t_{k+1} \) (\( \approx \) refinement of \( \tilde{b}_{b_k}^a \))

Stage 4: Deactivate transition conditions depending on \( x_{k+1} \)
A “Continuous” Method: Problem Definition


Simplicity is a major advantage of treating non-smooth problems with the same techniques as smooth problems.

Interval IVP: \[
\begin{aligned}
\dot{x} &= f(x), \\
 x(0) &\in [x_0],
\end{aligned}
\]

where \( f: \mathcal{D} \subset \mathbb{R}^n \to \mathbb{R}^n \)

or \( f: \mathcal{D} \subset \mathbb{I}^{n} \to \mathbb{I}^{n} \).

\( f \) is given in its algorithmic representation (inductive):

\[
\begin{cases}
\tau_i(x) = g_i(x) = x_i, \quad i = 1 \ldots n \\
\tau_i(x) = g_i(\tau_1(x), \ldots, \tau_{i-1}(x)), \quad i = n + 1 \ldots l, \\
g_i \in S_{EO} \cup S_{PW}
\end{cases}
\]

\( S_{EO} = \{ c, +, -, *, /, \sin, \cos, \ldots \} \) and \( S_{PW} \) are piecewise cont.

Goal: Find a derivative generalization to use with the usual theory.
Definition of Piecewise Functions $\phi(y)$ in $S_{PW}$

$$y = \tau_{\nu}(x), \: \phi_j(y), \: j = 0, \ldots, L \text{ smooth}$$

$$\phi(y) =
\begin{align*}
\phi_0(y) & \quad \text{for } c_{-1} = -\infty < y < c_0, \\
\phi_1(y) & \quad \text{for } c_0 < y < c_1, \\
& \ldots \\
\phi_{L-1}(y) & \quad \text{for } c_{L-2} < y < c_{L-1}, \\
\phi_L(y) & \quad \text{for } c_{L-1} < y < c_L = +\infty.
\end{align*}$$

An interval extension of $\phi$ over $x$ ($\phi(x)$):

$$\begin{align*}
\left\{ \begin{array}{ll}
\phi_i(x), & \text{if } x \subseteq (c_{i-1}, c_i), \\
\bigcup_{k=i+1}^{j-1} \phi_k([c_{k-1}, c_k]) \cup \phi_i([x, c_i]) \cup \phi_j([c_{j-1}, x]), & \text{if } x \subseteq (c_{i-1}, c_j)
\end{array} \right.
\end{align*}$$
Definition of the Derivative: Top-Down Approach

An interval extension of $\phi'$ over $x$ ($\phi'(x)$)

\[
\begin{cases}
\phi_i'(x), & \text{if } x \subseteq (c_{i-1}, c_i), \\
\bigcup_{k=i+1}^{j-1} \phi_k'([c_{k-1}, c_k]) \cup \phi_i'([x, c_i]) \cup \phi_j'([c_{j-1}, \overline{x}]) & \text{if } x \subseteq (c_{i-1}, c_j), \\
\bigcup \text{REST}, & \text{if we want the mean value theorem to hold.}
\end{cases}
\]

where REST depends on:

- how many switching points $x$ contains,
- whether $\phi$ is continuous,
Derivative for IF-THEN-ELSE (One Switching Point)

\[ \phi(x) = \begin{cases} 
\phi_0(x), & x < c_0, \\
\phi_1(x), & x > c_0.
\end{cases} \]

- If \( \phi \) is continuous, there is no REST

- If \( \phi \) is discontinuous

\[
\begin{cases}
\phi'_0([x, c_0]) \cup \left( \frac{\phi_1(c_0) - \phi_0(c_0)}{[c_0, \overline{x}] - x_0} \right) + \phi'_0([x, c_0]) \cup \phi'_1([c_0, \overline{x}]) & \text{if } x_0 \in [x, c_0), \\
\phi'_1([c_0, \overline{x}]) \cup \left( \frac{\phi_0(c_0) - \phi_1(c_0)}{[x, c_0] - x_0} \right) + \phi'_0([x, c_0]) \cup \phi'_1([c_0, \overline{x}]) & \text{if } x_0 \in (c_0, \overline{x}], \\
\phi'_0([x, c_0]) \cup \phi'_1([c_0, \overline{x}]) & \text{if } x_0 = c_0.
\end{cases}
\]

\( x_0 \) needed to avoid overconservative enclosures \([[-\infty, +\infty]]\)
Generalization

This definition can be generalized for the arbitrary number of \( c_i \in \mathbf{x}, \) but:
\( \mathbf{x} \) containing many \( c_i \) might be simply too wide.

If \( \phi(\cdot) \) contains several switching points \( c_i \), but \( \mathbf{x} \) contains only one:

\[
\phi'(\mathbf{x}) = \begin{cases} 
\phi_i'(\mathbf{x}) & \text{for } \mathbf{x} \subset (c_{i-1}, c_i), \\
\phi'_{\text{cont}}(\mathbf{x}) & \text{for } \mathbf{x} \subset (c_{i-1}, c_{i+1}), \text{ if } \phi \text{ is cont. in } c_i, \\
\phi'_{\text{dis}}(\mathbf{x}) & \text{for } \mathbf{x} \subset (c_{i-1}, c_{i+1}), \text{ if } \phi \text{ is discont. in } c_i.
\end{cases}
\]
Features

+ Right sides with several variables represented

\[ f(x_1, x_2) = |x_1| + x_1 \cdot \text{sign} (x_2), \]
\[ g(x_1, x_2) = x_2 - x_1, \]
\[ f_1 = 1 \quad f_2 = 2 \]

− No PW operations like

\[ f(x_1, x_2) = \begin{cases} 
1, & x_2 < x_1 \\
2, & x_2 > x_1 
\end{cases} . \]

Covered by Rihm

+ Better coverage for

\[ f(x) = \begin{cases} 
-h, & x < -x_+ \\
0, & -x_+ < x < x_+ \\
h, & x_+ < x 
\end{cases} . \]
Another Derivative Definitions

Slopes (implementation in Schnurr (2007), also of order 2)

\[ f : D \subseteq \mathbb{R} \mapsto \mathbb{R} \text{ continuous, } x_0 \in D, \text{ then } f(x) = f(x_0) + \delta f(x, x_0)(x - x_0) \]

slope in 1D

Interval slope of \( f \) over \( x \in D \):

\[ \delta f(x, x_0) \supseteq \{ \delta f(x, x_0) | x \in x \} \]

\[ f(x) \in f(x_0) + \delta f(x, x_0)(x - x_0) \]

\[ \delta f(x, x_0) = \left[ \inf_{x \in [x], x \neq x_0} \frac{f(x) - f(x_0)}{x - x_0}, \sup_{x \in [x], x \neq x_0} \frac{f(x) - f(x_0)}{x - x_0} \right] \]

A triple \((F_x, F_{x_0}, \delta F)\) with \( f(x) \in F_x, f(x_0) \in F_{x_0}, f(x) - f(x_0) \in \delta F(x - x_0) \)

(≈ suitable for a bottom-up approach)

Usage

Approach
Plug the definition into **ValEncIA-IVP**

**ValEncIA**
- an a posteriori method
- for smooth problems
- \( x(t) \in [x(t)] := x_{app}(t) + [R(t)] \)
  - verified enclosure
  - approximation
  - error bounds
- uses MVT, a fixed point theorem
- Jacobians only \( \rightarrow \) easy to adapt

Non-smooth
- upper semi-continuous right sides
- Kakutani’s fixed point theorem
- \( f' \) satisfies MVT

+ for Lipschitz cont. right sides or isolated switching points
- overestimation for sliding solutions
## Remarks on Software

⚠️ Although descriptions of methods are easily available the corresponding software is **not easy to obtain** (not maintained, etc.)

**Automaton-based**

(→ all rely on a kind of meta-language for defining the system)

<table>
<thead>
<tr>
<th>Software</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SPACEEx</strong></td>
<td><a href="http://spaceex.imag.fr/">http://spaceex.imag.fr/</a></td>
<td>(linear systems),</td>
</tr>
<tr>
<td><strong>dREACH</strong></td>
<td><a href="http://dreal.github.io/">http://dreal.github.io/</a></td>
<td>(the satisfiability modulo theories solver for the nonlinear theories of the reals),</td>
</tr>
<tr>
<td><strong>Ariadne</strong></td>
<td><a href="http://trac.parades.rm.cnr.it/ariadne/">http://trac.parades.rm.cnr.it/ariadne/</a></td>
<td>(extendable)</td>
</tr>
<tr>
<td><strong>FLOW</strong></td>
<td><a href="http://systems.cs.colorado.edu/research/cyberphysical/taylormodels/">http://systems.cs.colorado.edu/research/cyberphysical/taylormodels/</a></td>
<td>(open source, Taylor models)</td>
</tr>
</tbody>
</table>
Several Mentioned Articles


Z. Galias, Rigorous Study of the Chua’s Circuit Spiral Attractor, 2012
Th. A. Henzinger et al., Beyond HYTECH: Hybrid Systems Analysis Using Interval Numerical Methods, 2000
D. Ishii, Simulation and Verification of Hybrid Systems Based on Interval Analysis..., 2010
Known Exact Solution: Tacoma Narrows Suspension Bridge

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{1}{m} \left( \sin (4t) - q(x_1) \right)
\end{align*}
\]

\[
x_1(0) = 0 \\
x_2(0) = 1 \\
q(x_1) = \begin{cases} 
  x_1, & x_1 < 0 \\
  4x_1, & x_1 > 0 
\end{cases}
\]
Known Exact Solution: Oscillator

\[ m\ddot{x} = -f_i(x), \quad h = m = x_t = 1, \quad x(0) = 2, \quad v(0) = 0 \]

\[ f_1(x_1) = \begin{cases} 
- h, & x < 0 \\
+ h, & x > 0 
\end{cases} \quad f_2(x_1) = \begin{cases} 
- h, & x < -x_t \\
0, & -x_t < x < x_t \\
+ h, & x > x_t 
\end{cases} \]
Comparisons with Other Methods: Water Level

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= 0.5u(x_1)
\end{align*}
\]

\[x_1(0) = 5, \quad x_2(0) = 1\]

\[u(x_1) = \begin{cases} 
1, & x_1 < 3 \\
-1, & x_1 > 7 \\
0, & \text{otherwise}
\end{cases}\]

Width at \(t = 35\): \(\text{wid}(x_1) = 0.28\) as opposed to Nedilkov/von Mohrenschildt \(\text{wid}(x_1) = 10^{-7}\)
Example without a Classical Solution

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -0.2x_2 - x_1 + 2 \cos(\pi t) - u(x_2) \\
\end{align*}
\]

\[u(x_1) = \begin{cases} 
-4, & x_1 < 0 \\
+4, & x_1 > 0 
\end{cases}\]

\[x_1(0) = 3, \ x_2(0) = 4\]

The first switching point at \(t \approx 0.5\), the second at \(t \approx 2.03\), the solution leaves the switching surface \(v = 0\) after \(t \approx 2.6\)
Summary

Considered:
→ Different formulations and reformulations of non-smooth problems
→ Several approaches with result verification:
  → ODEs with switchings
  → Automaton-based
  → “Continuous”

Interesting topics:
− Systematize reformulations, find equivalencies, supply interval (etc.) reformulation
− Software-related issues, e.g. how to implement in order to easily integrate into available simulation packages?
− Reduction of overestimation (→ slopes? combination with non-verified methods?)
Properties of $\phi'(x)$

1. If the derivative of $\phi$ exists for $x \in \mathbf{x}$, then $\phi'(x) \in \phi'(\mathbf{x})$

2. The slope $\delta \phi(x, x_0) \subseteq \phi'(\mathbf{x})$

3. The mean value theorem holds:

   $$\phi(x) = \phi(x_0) + \phi'(\xi)(x - x_0) \in \phi(x_0) + \phi'(\mathbf{x})(\mathbf{x} - x_0)$$

4. If $\phi$ is continuous ($\phi_j(c_j) = \phi_{j+1}(c_j)$, $0 \leq j < L$), then $f(x)$ is continuous if all operations in $S_{EO}$ are continuous.
Non-smooth IVPs

**ValEncIA-IVP**

For Non-smooth IVPs

**General approach in ValEncIA: A posteriori**

\[ x(t) \in [x(t)] := \underbrace{x_{app}(t)}_{\text{non-verified approximation}} + \underbrace{[R(t)]}_{\text{error bounds}} \]

**Conditions for the right side:**

1. continuous
2. Lipschitz

---

\(^{1}\) VALIDation of state ENClosures using Interval Arithmetic for Initial Value Problems

---

E. Auer

University of Technology, Business and Design Wismar

Uncertain Non-Smooth Dynamic Systems
The algorithm for $0 \leq t \leq T$:

1. Start with $[x(0)]$, $x_{app}(t)$, $[R(0)]$

2. $k = 1 \ldots k_{max}$ or while $[\dot{R}^{(k+1)}([0, T])] \neq [\dot{R}^{(k)}([0, T])]$

Compute $[\dot{R}^{(k+1)}([0, T])] := \dot{x}_{app} + f([x^{(k)}])$, (MVT)

where $[x^{(k)}] := [x^{(k)}([0, T])]$

If $[\dot{R}^{(k+1)}([0, T])] \subseteq [\dot{R}^{(k)}([0, T])]$ then

$$
\begin{align*}
[R^{(k+1)}([0, T])] &:= [R(0)] + [\dot{R}^{(k+1)}([0, T])][0, T] \\
x^{(k+1)}([0, T])] &:= x_{app} + [R^{(k+1)}([0, T])]
\end{align*}
$$

Differences ((non-)smooth): Derivative definition, the fixed point theorem

To-do-list: Discontinuities in $x$ for the right side
Remarks on $f(x)$

- $f'(x)$ is obtained with `pwFunc`
- `pwFunc` uses `FADBAD++` and overloads `hull`, `d()`
- $f'(x)$ encloses both left and right derivatives
- `pwFunc` is plugged into `VALENCIA`

Class declaration

```cpp
template<class T>
class pwFunc{
public:
typedef T (*ptrFct)(const T & x);
pwFunc(const vector<interval> & p,
       const vector<ptrFct> & f);
T operator()(const T & x)
    { return getValueAtX(x); }
private:
    vector< ptrFct > functions;
    vector<interval> points;
    vector<T> subintervals;
    T getValueAtX(const T & x);
    void generateSubintervals(const T & x);
};
```
Implementation Example: A Discontinuous Function

```cpp
template <class T> T f1(const T& x) { return -1 + x; }
template <class T> T f2(const T& x) { return 1 + x; }
template<class T> T ff(const T& a) {
    vector<INTERVAL> p; p.push_back(0);
    vector<pwFunc<T>::ptrFct> functions;
    functions.push_back(&f1<T>); functions.push_back(&f2<T>);
    pwFunc<T> fp(p, functions); return fp(a); }
ff([-1,2]);
```

**Equation:**

\[
F_f (v) = \begin{cases} 
-1.0 + x & x < 0 \\
+1.0 + x & x > 0 
\end{cases}
\]

**Result:**

\([-2,3] ([1,6])\)
System for the SOFC temperature

\[ \dot{\vartheta}_{FC} = \frac{1}{c_{FC} m_{FC}} \left[ \frac{1}{R_A} (\vartheta_A - \vartheta_{FC}) - (c_{N_2} \zeta_{N_2,C} + c_{O_2} \zeta_{O_2}) u(t) \right. \\
- (c_{H_2} \dot{m}_{H_2} + c_{H_2O} \dot{m}_{H_2O} + c_{N_2} \dot{m}_{N_2,A}) (\vartheta_{FC} - \vartheta_{AG,in}) \left. + \frac{\Delta H_m(\vartheta_{FC}) \dot{m}}{M_{H_2}} \right] \\
= a(\vartheta_{FC}(t), p) + b(\vartheta_{FC}(t), p, d) \cdot u(t) \]