Lagrangian Reachability

J. Cyranka\textsuperscript{1}, M. Islam\textsuperscript{2}, G. Byrne\textsuperscript{3}, P. Jones\textsuperscript{3}, S. Smolka\textsuperscript{4}, R. Grosu\textsuperscript{5}

\textsuperscript{1}Rutgers University
\textsuperscript{2}Carnegie Mellon University
\textsuperscript{3}FDA
\textsuperscript{4}Stony Brook University
\textsuperscript{5}TU Wien

*CyberCardia Annual project meeting*
UPenn, 2017
Problem statement

Given a continuous dynamical system (set of ODEs)
Compute continuous Reachtube overestimate bounding the ODE trajectories
Problem statement 2

Too wide Reachtube may result in false positives.
Existing approaches

1. Taylor-expansion in time, variational-expansion in space

2. Taylor-expansion in time and space of the solution set (Taylor models)

3. Bloating-factor-based and discrepancy-function-based approach

References:
Existing approaches

- Taylor-expansion in time, variational-expansion in space
- CAPD,
Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   1. CAPD,
   2. VNode,
Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   - CAPD,
   - VNode,
2. Taylor-expansion in time and space of the solution set (Taylor models)
Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   - CAPD,
   - VNode,

2. Taylor-expansion in time and space of the solution set (Taylor models)
   - Cosy Infinity,
Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   1. CAPD,
   2. VNode,
2. Taylor-expansion in time and space of the solution set (Taylor models)
   1. Cosy Infinity,
   2. Flow*. 

Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   1. CAPD,
   2. VNode,

2. Taylor-expansion in time and space of the solution set (Taylor models)
   1. Cosy Infinity,
   2. Flow*.

3. Bloating-factor-based and discrepancy-function-based approach
Existing approaches

1. Taylor-expansion in time, variational-expansion in space
   1. CAPD,
   2. VNode,

2. Taylor-expansion in time and space of the solution set (Taylor models)
   1. Cosy Infinity,
   2. Flow*.

3. Bloating-factor-based and discrepancy-function-based approach

Our approach:

Explore possibilities of computation of the Reachtube overestimate using the Lagrangian coordinates method and identify advantages of doing so.
Figure: general idea of Lagrangian (deformed) coordinates source: wikipedia.org
We work with time-variant ordinary differential equations (ODEs):

\[ x'(t) = F(t, x(t)), \quad (1a) \]
\[ x(t_0) = x_0, \quad (1b) \]

where \( x: \mathbb{R} \to \mathbb{R}^n \). We assume that \( F \) is a smooth function, which guarantees short-time existence of solutions.
For $t_0$, set of states $\mathcal{X} \subset \mathbb{R}^n$, and $T > t_0$, our goal is to compute a sequence of time-stamped sets of states $(R_1, t_1), \ldots, (R_k, t_k)$ satisfying (tight):

$$\text{Reach}((t_0, \mathcal{X}), [t_{i-1}, t_i]) \subset R_i \text{ for } i = 1, \ldots, k,$$

where $\text{Reach}((t_0, \mathcal{X}), [t_{i-1}, t_i])$ denotes the set of reachable states of (1) in the interval $[t_{i-1}, t_i]$. 
How our algorithm works?

**Figure:** Illustration of our algorithm
General idea of the algorithm

1. Compute over-approximation of the *gradient of the solution-flows*, and the *Cauchy-Green deformation tensor* (currently done by CAPD library),

\[ \text{Compute over-approximation of the gradient of the solution-flows, and the Cauchy-Green deformation tensor (currently done by CAPD library),} \]
General idea of the algorithm

1. Compute over-approximation of the gradient of the solution-flows, and the Cauchy-Green deformation tensor (currently done by CAPD library),

2. Optimize for positive-definite symmetric matrix $M_1$, defining the weighted norm minimizing the Streching Factor for a choice of gradient,
General idea of the algorithm

1. Compute over-approximation of the gradient of the solution-flows, and the Cauchy-Green deformation tensor (currently done by CAPD library),

2. Optimize for positive-definite symmetric matrix $M_1$, defining the weighted norm minimizing the Streching Factor for a choice of gradient,

3. Compute an upper bound for the Streching Factor $\Lambda$, then the ball-overestimate is

$$B_{M_1}(\phi_{t_0}^{t_1}(x_0), \Lambda \cdot \delta_0).$$
Cauchy-Green deformation tensor:

\[ CGDT = (\nabla X x)^T \cdot \nabla X x. \] (2)

Stretching factor

\[ \sqrt{\lambda_{max}(CGDT)} > 0 \] (3)
Definition

Given positive-definite symmetric matrix $M \in \mathbb{R}^{n \times n}$ we define the $M$-norm of $\mathbb{R}^n$ vectors by

$$\|y\|_M = \sqrt{y^T M y}.$$  \hfill (4)

Given the decomposition

$$M = C^T C,$$

the matrix norm induced by (4) is

$$\|A\|_M = \sqrt{\lambda_{max} \left( (C^T)^{-1} \cdot A^T \cdot M \cdot A \cdot C^{-1} \right)},$$  \hfill (5)
Notation

Let $\phi_{t_0}^{t_1}(x)$ be the solution of (1) with the initial condition $(t_0, x)$ at time $t_1$, let $D_x \phi_{t_0}^{t_1}$ be the gradient of the flow. Let $M_0, M_1 \in \mathbb{R}^{n \times n}$ be positive-definite symmetric matrices, and $C_0^T C_0 = M_0, C_1^T C_1 = M_1$ be their decompositions respectively.
Let $\mathcal{X} = B_{M_0}(x_0, \delta_0) \subset \mathbb{R}^n$ be a set of initial states. Assume that there exists a compact, conservative enclosure $\mathcal{D} \subset \mathbb{R}^{n \times n}$ for the gradients, such that:

$$D_x \phi_{t_0}^{t_1}(x) \in \mathcal{D} \text{ for all } x \in \mathcal{X}.$$  \hspace{1cm} (6)

Suppose $\Lambda > 0$ is such that:

$$\Lambda \geq \sqrt{\lambda_{\max} \left( (C_0^T)^{-1} D^T M_1 D C_0^{-1} \right)}, \text{ for all } D \in \mathcal{D}.$$  

Then it holds that:

$$\phi_{t_0}^{t_1}(x) \in B_{M_1}(\phi_{t_0}^{t_1}(x_0), \Lambda \cdot \delta_0).$$
Theorem (LRT-Conservativity)

Assume that the rigorous tool used in the Lagrangian Reachtube Algorithm (LRT) produces conservative gradient enclosures for system (1), and it guarantees existence of the solutions within time intervals. Assume also that the LRT terminates on the provided inputs.

Then, the output of the LRT is a conservative reachtube over-approximation of (1) at times \( \{t_j\}_{j=0}^k \), that is:

\[
(t_0, x) \subset B_{M_j}([x_j], \delta_j), \text{ for } j = 1, \ldots, k,
\]

bounded solutions exists for all intermediate times \( t \in (t_j, t_{j+1}) \).
To compute an over-approximation of the gradient we use CAPD library ($C^1$ Lohner algorithm),
Implementation

1. To compute an over-approximation of the gradient we use CAPD library ($C^1$ Lohner algorithm),
2. We implemented an interface to communicate MATLAB ↔ C++,
   
   we optimize for positive definite matrix $M_1$ using a Matlab package, and compute bound for the eigenvalues of the deformation tensor matrices using Intlab,
### Algorithm instability

<table>
<thead>
<tr>
<th>BM</th>
<th>dt</th>
<th>T</th>
<th>ID</th>
<th>LRT (F/I)</th>
<th>Flow* (F/I)</th>
<th>(direct) CAPD</th>
<th>nLRT (F/I)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(A/I)</td>
<td>(A/I)</td>
<td>(A/I)</td>
<td>(A/I)</td>
</tr>
<tr>
<td>B(2)</td>
<td>0.01</td>
<td>20</td>
<td>0.02</td>
<td>7.7e-5</td>
<td>0.09</td>
<td>Fail</td>
<td>Fail</td>
</tr>
<tr>
<td>I(2)</td>
<td>0.01</td>
<td>20</td>
<td>0.02</td>
<td>4.3e-9</td>
<td>0.09</td>
<td>7.4e-9</td>
<td>1.45</td>
</tr>
<tr>
<td>L(3)</td>
<td>0.001</td>
<td>2</td>
<td>1.4e-6</td>
<td>1.6e5</td>
<td>9.0e3</td>
<td>4.5e4</td>
<td>1.2e22</td>
</tr>
<tr>
<td>F(2)</td>
<td>0.01</td>
<td>40</td>
<td>2e-3</td>
<td>1.2e-42</td>
<td>1.01</td>
<td>NA</td>
<td>5.64e7</td>
</tr>
<tr>
<td>M(2)</td>
<td>1e-3</td>
<td>4</td>
<td>2e-3</td>
<td>0.006</td>
<td>0.22</td>
<td>0.29</td>
<td>4.01</td>
</tr>
<tr>
<td>R(4)</td>
<td>0.01</td>
<td>20</td>
<td>1e-2</td>
<td>2.2e-19</td>
<td>0.07</td>
<td>2.9e-11</td>
<td>Fail</td>
</tr>
<tr>
<td>O(7)</td>
<td>0.01</td>
<td>4</td>
<td>2e-4</td>
<td>71.08</td>
<td>34.35</td>
<td>620</td>
<td>Fail</td>
</tr>
<tr>
<td>P(12)</td>
<td>5e-4</td>
<td>0.1</td>
<td>0.01</td>
<td>5.25</td>
<td>4.76</td>
<td>62.4</td>
<td>19.4</td>
</tr>
</tbody>
</table>

**Figure:** Set of benchmarks from [1] + Lorenz equations L(3), Mitchell Schaeffer cardiac cell model M(2)
Figure: (a) A ball in the weighted norm given by $M$ of radius 1 (the ellipsoidal set). (b) The ellipsoidal set in its eigen-coordinates (unrotated). (c) Volume computation by wrapping the ellipsoidal set in a box: blue rectangle in eigen-coordinates and green square in canonical-coordinates.
Figure: Volume comparison for forced Van der Pol Oscillator

(a) time horizon = [0, 10]

(b) time horizon = [3.5, 10]
Conclusions

- For most of the used benchmarks LRT compares more favorably to other tools (Flow*, CAPD) in the sense of (F/I)V and (A/I)V metrics (adapted from [1]),
Conclusions

- For most of the used benchmarks LRT compares more favorably to other tools (Flow*, CAPD) in the sense of (F/I)V and (A/I)V metrics (adapted from [1]),

- Apparently, for ODEs exhibiting asymptotically stable behaviour, LRT results in tighter Reachtube over-approximation,
Conclusions

- For most of the used benchmarks LRT compares more favorably to other tools (Flow*, CAPD) in the sense of (F/I)V and (A/I)V metrics (adapted from [1]),
- Apparently, for ODEs exhibiting asymptotically stable behaviour, LRT results in tighter Reachtube over-approximation,
- Some improvements of the algorithm are still possible,
Conclusions

- For most of the used benchmarks LRT compares more favorably to other tools (Flow*, CAPD) in the sense of (F/I)V and (A/I)V metrics (adapted from [1]),
- Apparently, for ODEs exhibiting asymptotically stable behaviour, LRT results in tighter Reachtube over-approximation,
- Some improvements of the algorithm are still possible,
- Future goal – Reachtube enclosures for PDEs.