RINO: Robust INner and Outer Approximated Reachability of Neural Networks Controlled Systems

Eric Goubault and Sylvie Putot

June 15, 2022

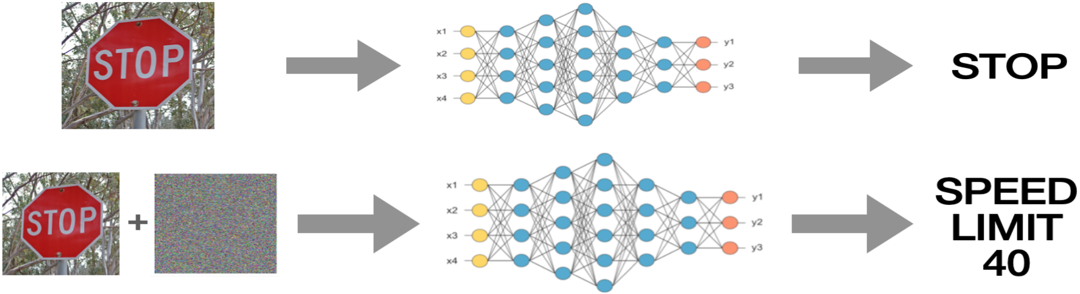
Overview

1. Introduction: safe learning in autonomous systems
2. Taylor expansion based approach for outer-approximation
3. AE extensions for (robust) inner and outer approximations
4. RINO: tool and Evaluation

Introduction: safe learning in autonomous systems

Safe learning in autonomous systems: perception

Perception: objects (obstacles, traffic sign, etc.) detection should be robust to change in lighting, physical attacks, adversarial noise

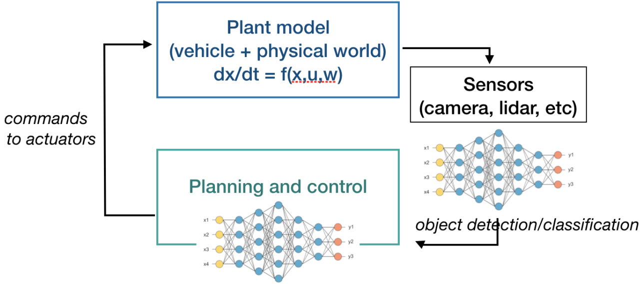


Robustness issues are amenable to (post-training) reachability-based verification

Safe learning in autonomous systems: planning and control

Planning and control:

- robots need to operate in unknown, uncertain and dynamic environments
- from offline to online planning and control, in learned environments

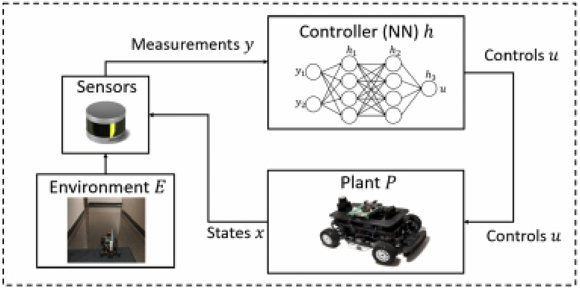


Reach-avoid or similar properties well suited to reachability verification

The closed-loop: a time-triggered hybrid system

Given

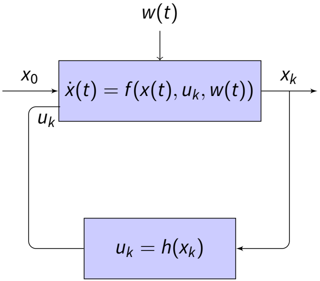
- plant dynamic f,
- state x, control u, disturbance $w \in W$
- NN controller h
- control period Δt_u

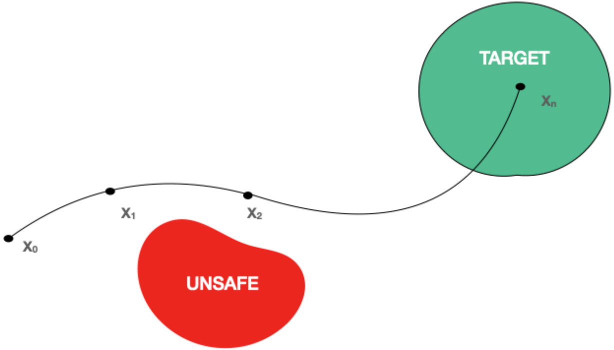


Time-triggered (u computed every $\Delta_u t$) dynamical system with non-linear feedback:

$$
\begin{aligned}
\dot{x}(t) &= f(x(t), u(t), w(t)) \\
x(t_0) &= x_0 \in X_0 \\
u(t) &= u_k = h(y(x(\tau_k))), \text{ for } t \in [\tau_k, \tau_{k+1}), \text{ with } \tau_k = t_0 + k\Delta t_u, \ \forall k \geq 0
\end{aligned}
$$

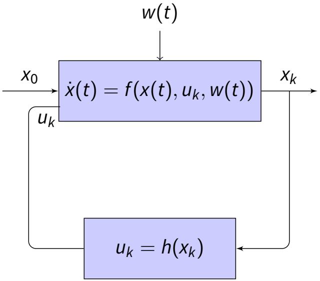
Reachability analysis for safety and robustness verification

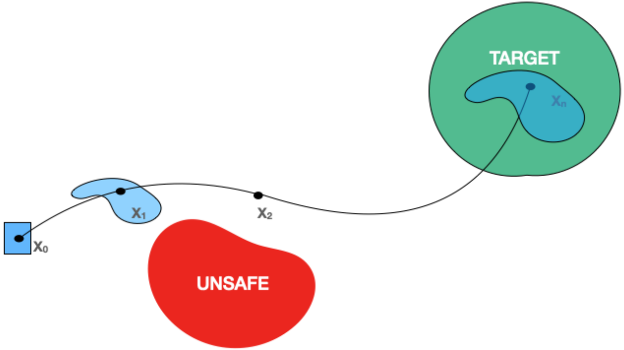




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions

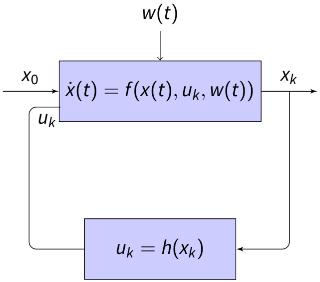
Reachability analysis for safety and robustness verification

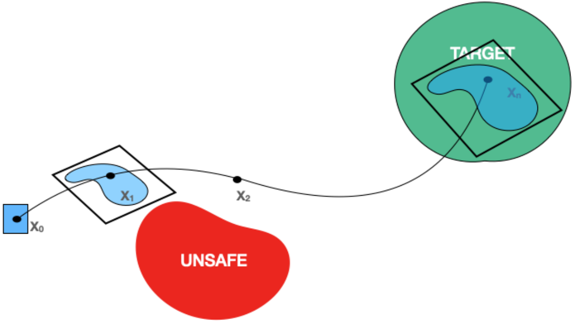




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- Also for noisy initial conditions x_0 (robustness)

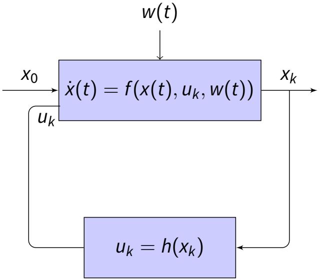
Reachability analysis for safety and robustness verification

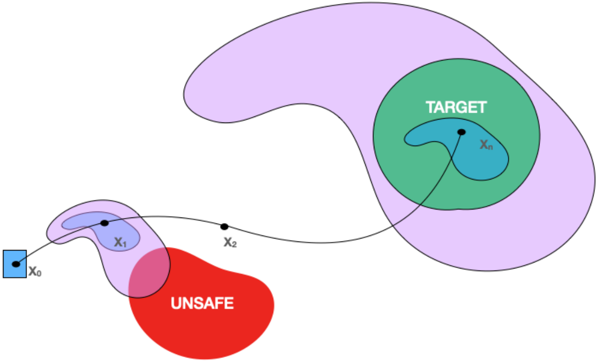




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- Also for noisy initial conditions x_0 (robustness)
  - Proven by over-approximated reachability

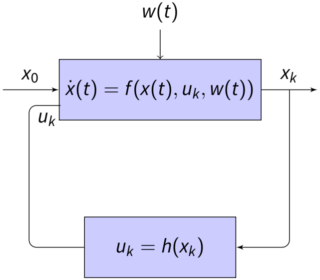
Reachability analysis for safety and robustness verification

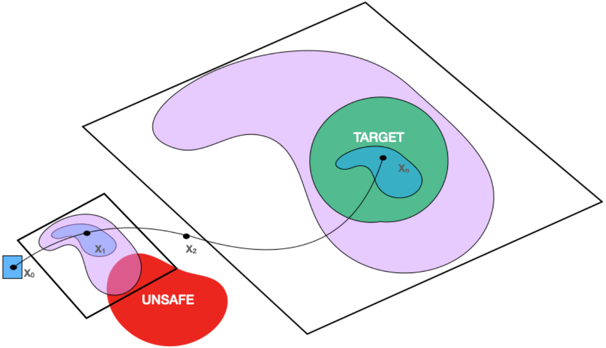




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- And external disturbances $w(t)$

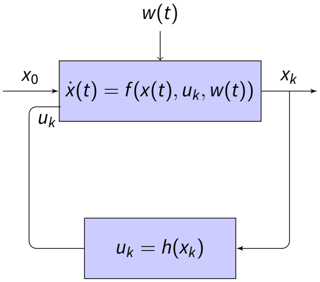
Reachability analysis for safety and robustness verification

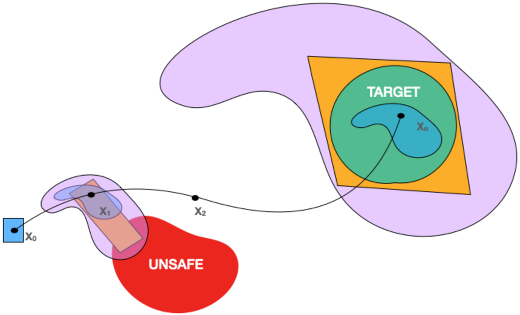




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- And external disturbances $w(t)$
  - (Maximal) over-approximation unconclusive

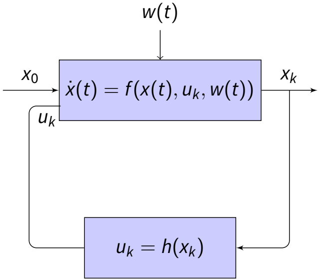
Reachability analysis for safety and robustness verification

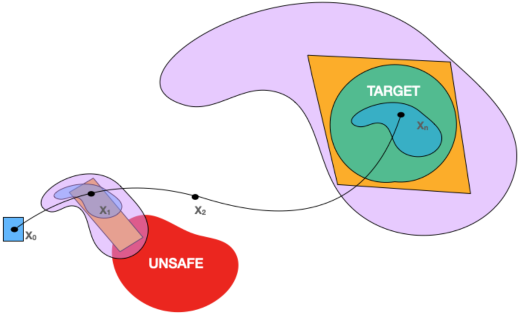




- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- And external disturbances $w(t)$
  - **Under-approximation**: $\exists x_0, \exists w(t)$ such that the trajectory is unsafe

Reachability analysis for safety and robustness verification





- Classical reach-avoid problem: reaching target region while avoiding unsafe regions
- And external disturbances $w(t)$
  - **Under-approximation**: $\exists x_0, \exists w(t)$ such that the trajectory is unsafe
  - **Under-approximation**: $\forall x$ in target, $\exists x_0, \exists w(t)$ s.t. x is reached (target covered) + some final states proven to be outside the target

Reachability problems with disturbances w

Compute inner and outer-approximating sets $I(t)$ and $O(t)$ such that:

- Maximal reachability

$$I_{\mathcal{E}}(t) \subseteq R_{\mathcal{E}}^{f,h}(t; \mathbb{X}_0, \mathbb{W}) = \{x \,|\, \exists w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)\} \subseteq O_{\mathcal{E}}(t)$$

- Minimal or robust reachability

$$I_{\mathcal{AE}}(t) \subseteq R_{\mathcal{AE}}^{f,h}(t; \mathbb{X}_0, \mathbb{W}) = \{x \,|\, \forall w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)\} \subseteq O_{\mathcal{AE}}(t)$$

We have:

$$R_{\mathcal{AE}}^{f,h}(t; \mathbb{X}_0, \mathbb{W}) \subseteq R_{\mathcal{E}}^{f,h}(t; \mathbb{X}_0, \mathbb{W})$$

Taylor expansion based approach for outer-approximation

Taylor expansions for ODEs reachability (Berz & Makino) I

For $f \in C^k$, over-approximate the solution of $\dot{x}(t) = f(x(t)),\ x(t_0) \in [x_0]$ on $[t_0, T]$:

- Time grid $t_0 < t_1 < \ldots < t_N = T$
- Taylor-Lagrange expansion in t of the solution on each time slice $[t_j, t_{j+1}]$

$$[\boldsymbol{x}](t, t_j, [\boldsymbol{x}_j]) = [\boldsymbol{x}_j] + \sum_{i=1}^{k-1} \frac{(t - t_j)^i}{i!} f^{[i]}([\boldsymbol{x}_j]) + \frac{(t - t_j)^k}{k!} f^{[k]}([\boldsymbol{r}_{j+1}])$$

- Evaluation of expansion at time t_{j+1} gives initial solution on next time slice

Set-valued computations: evaluation with intervals, *affine forms(or zonotopes)*, etc.

Taylor expansions for ODEs reachability (Berz & Makino) II

- The $f^{[i]}$ are defined inductively; can be computed by automatic differentiation:

$$
\begin{aligned}
f_k^{[1]} &= f_k \\
f_k^{[i+1]} &= \sum_{j=1}^{n} \frac{\partial f_k^{[i]}}{\partial x_j} f_j
\end{aligned}
$$

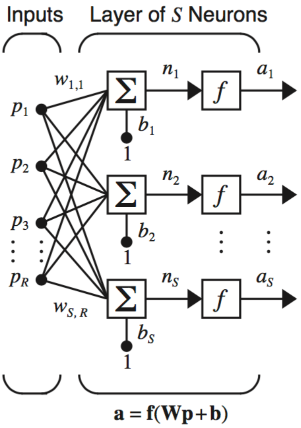
- Bounding the remainder supposes to first compute an enclosure $[\boldsymbol{r}_{j+1}]$ of solution $x(t, z_0)$ on $[t_j, t_{j+1}]$, classical by Picard iteration: find h_{j+1}, $[r_{j+1}]$ such that

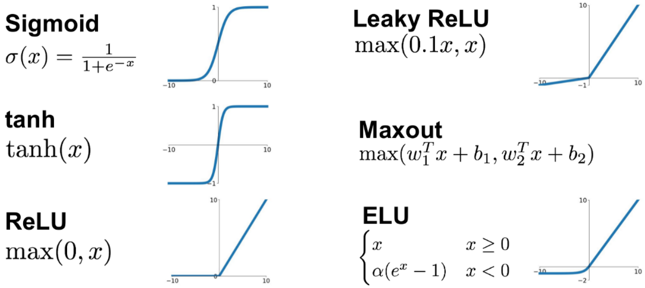
$$[x_j] + [0, h_{j+1}] f([r_{j+1}]) \subseteq [r_{j+1}]$$

- Initialization of next iterate $[\boldsymbol{x}_{j+1}] = [\boldsymbol{x}](t_{j+1}, t_j, [\boldsymbol{x}_j])$

Feedforward neural network controlled system

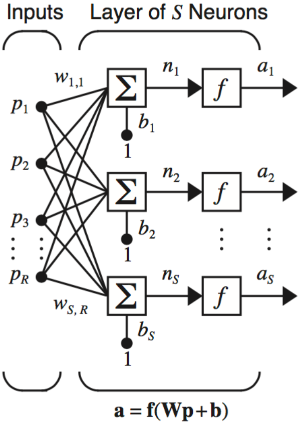
Each layer consists in a linear transform followed by a non linear activation function:

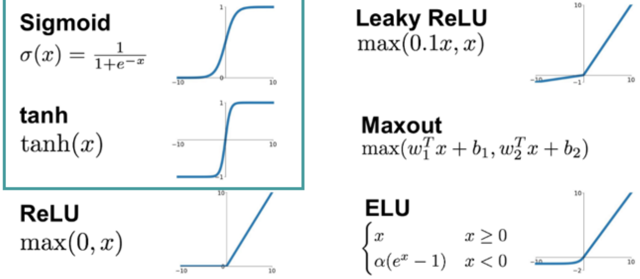




Feedforward neural network controlled system

Each layer consists in a linear transform followed by a non linear activation function:





We focus on differentiable activation functions (needed for inner-approximations)

Taylor expansions for neural network controlled system

Straightforward extension for outer-approximations in the case of a time-triggered feedforward neural network controller:

$$\begin{aligned} \dot{x}(t) &= f(x(t), h(x(\tau_k))) \text{ for } t \in [\tau_k, \tau_{k+1}), \text{ with } \tau_k = t_0 + k\Delta t_u, \ \forall k \geq 0 \\ x(t_0) &= x_0 \in X_0 \end{aligned}$$

- evaluation of $h(x(\tau_k))$ for set-valued $x(\tau_k)$ for instance with intervals or zonotopes as for any nonlinear function
- requires $\{\tau_k, k \geq 0, \tau_k < T\} \subseteq \{t_1, \ldots, t_N\}$: the stepwise constant control changes values at a subset of the points of the time grid of the Taylor expansions.

AE extensions for (robust) inner and outer approximations

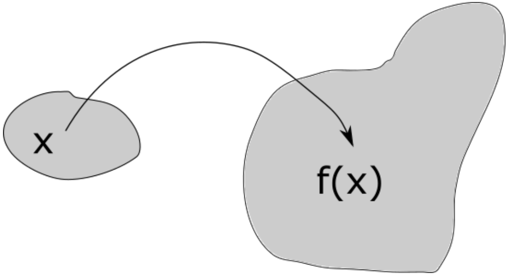
AE extensions for function image computation

Given

- $f : \mathbb{R}^m \to \mathbb{R}^n$
- a set $\boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$

we want:

range$(f, \boldsymbol{x}) = \{f(x), x \in \boldsymbol{x}\}$.



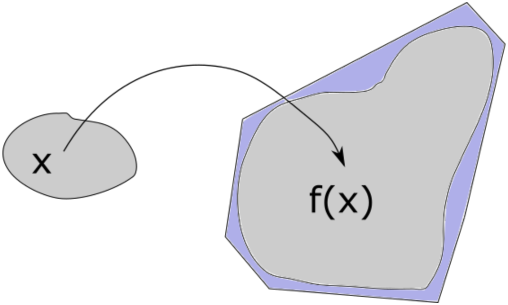
AE extensions for function image computation

Given

- $f : \mathbb{R}^m \to \mathbb{R}^n$
- a set $\boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$

we want:

range$(f, \boldsymbol{x}) = \{f(x), x \in \boldsymbol{x}\}$.



- Over-approximating extension of f (or inclusion function):
  $\boldsymbol{f}_o : \mathcal{P}(\mathbb{R}^m) \to \mathcal{P}(\mathbb{R}^n)$ such that $\forall \boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$, range$(f, \boldsymbol{x}) \subseteq \boldsymbol{f}_o(\boldsymbol{x})$

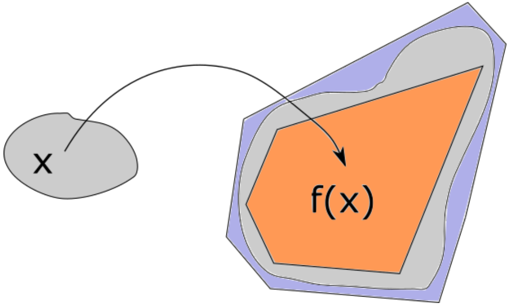
AE extensions for function image computation

Given

- $f : \mathbb{R}^m \to \mathbb{R}^n$
- a set $\boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$

we want:

range$(f, \boldsymbol{x}) = \{f(x), x \in \boldsymbol{x}\}$.



- Over-approximating extension of f (or inclusion function):
  $\boldsymbol{f}_o : \mathcal{P}(\mathbb{R}^m) \to \mathcal{P}(\mathbb{R}^n)$ such that $\forall \boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$, range$(f, \boldsymbol{x}) \subseteq \boldsymbol{f}_o(\boldsymbol{x})$
- Under-approximating extension of f:
  $\boldsymbol{f}_u : \mathcal{P}(\mathbb{R}^m) \to \mathcal{P}(\mathbb{R}^n)$ such that $\forall \boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$, $\boldsymbol{f}_u(\boldsymbol{x}) \subseteq$ range$(f, \boldsymbol{x})$

AE extensions for function image computation

- Over-approximating extension of f (or inclusion function):
  $\boldsymbol{f}_o : \mathcal{P}(\mathbb{R}^m) \to \mathcal{P}(\mathbb{R}^n)$ such that $\forall \boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$, $\text{range}(f, \boldsymbol{x}) \subseteq \boldsymbol{f}_o(\boldsymbol{x})$
- Under-approximating extension of f:
  $\boldsymbol{f}_u : \mathcal{P}(\mathbb{R}^m) \to \mathcal{P}(\mathbb{R}^n)$ such that $\forall \boldsymbol{x}$ in $\mathcal{P}(\mathbb{R}^m)$, $\boldsymbol{f}_u(\boldsymbol{x}) \subseteq \text{range}(f, \boldsymbol{x})$

Can be interpreted as AE propositions = quantified propositions where universal quantifiers (A) precede existential quantifiers (E)

$$\text{range}(f, \boldsymbol{x}) \subseteq \boldsymbol{z} = \boldsymbol{f}_o(\boldsymbol{x}) \Leftrightarrow \forall x \in \boldsymbol{x},\ \exists z \in \boldsymbol{z},\ f(x) = z$$

$$\boldsymbol{f}_u(\boldsymbol{x}) = \boldsymbol{z} \subseteq \text{range}(f, \boldsymbol{x}) \Leftrightarrow \forall z \in \boldsymbol{z},\ \exists x \in \boldsymbol{x},\ f(x) = z$$

Mean-Value AE extensions (scalar-valued function)

Theorem (Generalized Interval Mean-Value Theorem,Goldsztejn 2012)

- $f : \mathbb{R}^m \to \mathbb{R}$ *be a continuously differentiable function,* $\boldsymbol{x}$ *an initial box of* $\mathbb{R}^m$,
- $x_0 = mid(\boldsymbol{x})$ *the center of the box x,* $\boldsymbol{f}_0 = [\underline{f_0}, \overline{f_0}]$ *such that* $f(x_0) \in \boldsymbol{f}_0$
- $\Delta_i = [\underline{\Delta_i}, \overline{\Delta_i}]$ *such that* $\{|f_i'(x_{0,1}, \ldots, x_{0,i-1}, x_i, \ldots, x_m)|, x \in \boldsymbol{x}\} \subseteq \Delta_i$

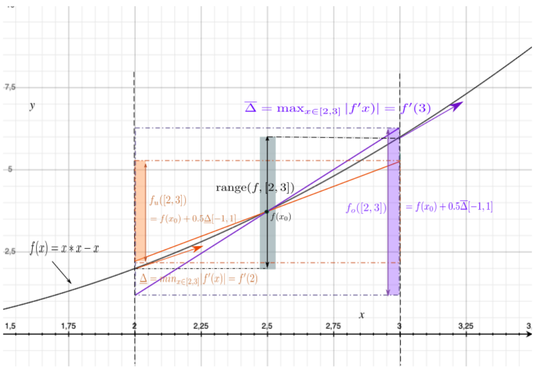
$$range(f, \boldsymbol{x}) \subseteq [\underline{f_0}, \overline{f_0}] + \sum_{i=1}^{m} \overline{\Delta_i} radius(\boldsymbol{x}_i)[-1, 1]$$

$$[\overline{f_0} - \sum_{i=1}^{m} \underline{\Delta_i} radius(\boldsymbol{x}_i), \underline{f_0} + \sum_{i=1}^{m} \underline{\Delta_i} radius(\boldsymbol{x}_i)] \subseteq range(f, \boldsymbol{x})$$

- Interval abstractions over $\boldsymbol{x}$ of $f(x) = f(x_0) + \int_{x_0}^{x} f'(x)dx, \ x \in \boldsymbol{x}$
- For over-approximation, first proposed by Moore (as centered interval form)

Example

- $f(x) = x^2 - x$ over $\boldsymbol{x} = [2, 3]$
- $f(2.5) = 3.75$
- $|\boldsymbol{f}'([2, 3])| \subseteq [3, 5] = [\underline{\Delta}, \overline{\Delta}]$.



Then,

$$3.75 + 0.5 * 3 * [-1, 1] \subseteq \text{range}(f, [2, 3]) \subseteq 3.75 + 0.5 * 5 * [-1, 1]$$

from which we deduce $[2.25, 5.25] \subseteq \text{range}(f, [2, 3]) \subseteq [1.25, 6.25]$.

AE extensions when f is the flow $\varphi^{f,h}(t; x_0, w)$ of the system

- Maximal reachability

$$I_{\mathcal{E}}(t) \subseteq R^{f,h}_{\mathcal{E}}(t; \mathbb{X}_0, \mathbb{W}) = \{x \mid \exists w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)\} \subseteq O_{\mathcal{E}}(t)$$

$$\forall x \in I_{\mathcal{E}}(t), \exists w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)$$

$$\forall w \in \mathbb{W}, \forall x_0 \in \mathbb{X}_0,\ \exists x \in O_{\mathcal{E}}(t), x = \varphi^{f,h}(t; x_0, w)$$

- Minimal or robust reachability

$$I_{\mathcal{AE}}(t) \subseteq R^{f,h}_{\mathcal{AE}}(t; \mathbb{X}_0, \mathbb{W}) = \{x \mid \forall w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)\} \subseteq O_{\mathcal{AE}}(t)$$

$$\forall x \in I_{\mathcal{AE}}(t), \exists w \in \mathbb{W}, \exists x_0 \in \mathbb{X}_0,\ x = \varphi^{f,h}(t; x_0, w)$$

$$\forall x_0 \in \mathbb{X}_0,\ \exists w \in \mathbb{W},\ \exists x \in O_{\mathcal{AE}}(t), x = \varphi^{f,h}(t; x_0, w)$$

AE extensions when f is the flow $\varphi^{f,h}(t; x_0, w)$ of the system

In order to use the generalized mean-value theorem on

$$\begin{aligned} \dot{x}(t) &= f(x(t), h(x(\tau_k))) \text{ for } t \in [\tau_k, \tau_{k+1}), \text{ with } \tau_k = t_0 + k\Delta t_u, \ \forall k \geq 0 \\ x(t_0) &= x_0 \in X_0 \end{aligned}$$

- Need bounds on the solutions of the system (trajectories)
- Need bounds on the solution of the variational equations (Jacobian of trajectories wrt initial states and uncertainties)
- Taylor expansions in time for vector field $f(x(t), h(x(\tau_k)))$ and its Jacobian: implies differentiating h, using $tanh'(x) = 1.0 - tanh(x)^2$ and $sig'(x) = sig(x)(1 - sig(x))$.

RINO: tool and Evaluation

RINO (Robust INner and Outer reachability)

Available from

`https://github.com/cosynus-lix/RINO/`

Computes Inner and Outer Approximations of Robust and Maximal Reachable sets:

- Continuous-time (possibly delayed) or discrete-time uncertain dynamical systems
- Possibly controlled by a neural network (with differentiable activation functions)
- Guaranteed computations using Taylor Expansions in time and Zonotopes in space

Relies on

- FILIB++ for interval arithmetic,
- aaflib for affine arithmetic,
- FADBAD++ for automatic differentiation.

Examples and Comparison to existing work

- Verisig and Verisig 2.0 [4, 3]: sigmoid/tanh is the solution to a differential equation: transform the neural network into an equivalent hybrid system (solved with Taylor Model based reachability Flowstar)
- ReachNN and ReachNNstar: [2, 1] : Bernstein polynomials + Taylor models (Flowstar)

J. Fan, C. Huang, X. Chen, W. Li, and Q. Zhu.
Reachnn*: A tool for reachability analysis of neural-network controlled systems.
In *ATVA 2020,*. Springer, 2020.

C. Huang, J. Fan, W. Li, X. Chen, and Q. Zhu.
Reachnn: Reachability analysis of neural-network controlled systems.
ACM Trans. Embed. Comput. Syst., 18, 2019.

R. Ivanov, T. Carpenter, J. Weimer, R. Alur, G. Pappas, and I. Lee.
Verisig 2.0: Verification of neural network controllers using taylor model preconditioning.
In *Computer Aided Verification*, pages 249–262. Springer International Publishing, 2021.

R. Ivanov, J. Weimer, R. Alur, G. J. Pappas, and I. Lee.
Verisig: verifying safety properties of hybrid systems with neural network controllers.
2019.

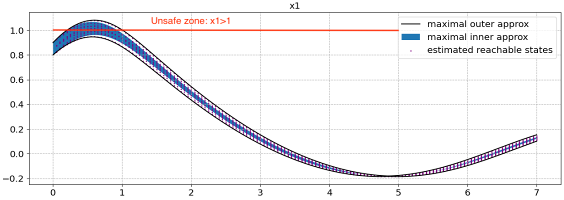
Benchmark examples

Name	Dynamics	Initial set	Horizon	Control step
Mountain Car sigmoid 2×200	$\dot{x}_1 = x_2$ $\dot{x}_2 = 0.0015u - 0.0025\cos(3x_1)$	$[-0.5, -0.48]$ $[0, 0.001]$	$T = 75$	1
discrete MC (stepsize 1) sigmoid 2×200	$x_1^{n+1} = x_1^n + x_2^n$ $x_2^{n+1} = x_2^n + 0.0015u^n$ $-0.0025\cos(3x_1^n)$	$[-0.5, -0.48]$ $[0, 0.001]$	$T = 75$	1
TORA tanh 3×20	$\dot{x}_1 = x_2$ $\dot{x}_2 = -x_1 + 0.1 * \sin(x_3)$ $\dot{x}_3 = x_4$ $\dot{x}_4 = u$	$[-0.77, -0.75]$ $[-0.45, -0.43]$ $[0.51, 0.54]$ $[-0.3, -0.28]$	$T = 5$	0.1
ACC ?	$\dot{x}_1 = x_2, \quad \dot{x}_4 = x_5$ $\dot{x}_2 = x_3, \quad \dot{x}_5 = x_6$ $\dot{x}_3 = -4 - 0.0001x_2^2 - 2x_3$ $\dot{x}_6 = 2u - 0.0001x_5^2 - 2x_6$	$x_1 = [90, 91]$ $x_2 = [32, 32.05]$ $x_4 = [10, 11]$ $x_5 = [30, 30.05]$	$T = 5$	0.1
B1 sigmoid 3×20	$\dot{x}_1 = x_2$ $\dot{x}_2 = ux_2^2 - x_1$	$[0.8, 0.9]$ $[0.5, 0.6]$	$T = 7$	0.2
B2 sigmoid 3×20	$\dot{x}_1 = x_2 - x_1^3$ $\dot{x}_2 = u$	$[0.7, 0.9]$ $[0.7, 0.9]$	$T = 1.8$	0.2

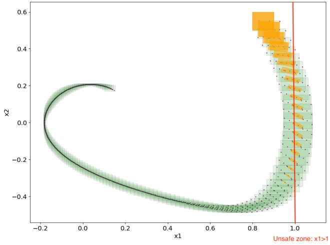
Comparison results (faster for comparable precision)

Example	% width Verisig2 over RINO	Ratio time Verisig2/RINO	% width ReachNN* over RINO	Ratio time ReachNN*/RINO
TORA (tanh)	117,6 %	38,6	Mem full	Mem full
	98,4 %			
	106,7 %			
	128,0 %			
TORA (sig)	115,7 %	43,4	Mem full	Mem full
	68,0 %			
	110,1 %			
	133,3 %			
ACC (tanh)	101,9 %	500,8	Time out	Time out
	105,6 %			
	103,3 %			
	110,1 %			
	105,1 %			
	65,8 %			
B1 (tanh)	84,9 %	88,8	96,7 %	85,1
	287,8 %		245,0 %	
B1 (sig)	112,1 %	105,4	227,8 %	86,8
	140,6 %		441,9 %	
B2 (sig)	263,2 %	77,6	408,8 %	121,9
	60,4 %		513,7 %	

B1: sampling (purple dots) and inner/outer-approximations



(a) x_1 as function of time



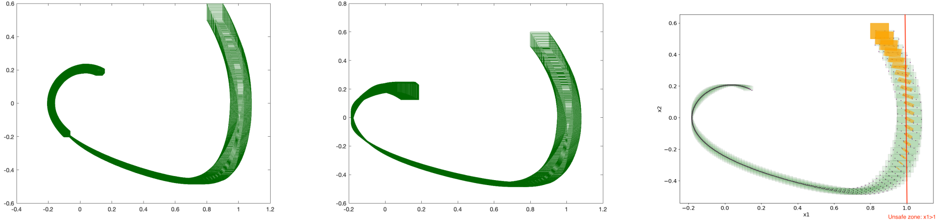
(b) Joint range (x_1, x_2)

- Over-approximation is very tight
- Samples show $(x1, x2)$ becomes almost a 1-dim curve: inner-approx difficult!
- N-dim inner-approximation more difficult and imprecise than 1-dim

Safety property $x_1 < 1$ (red line):

- over-approx raises a potential alarm
- under-approx proves falsification

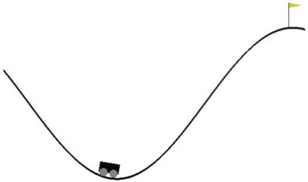
B1: comparison to Verisig 2.0 and ReachNNstar



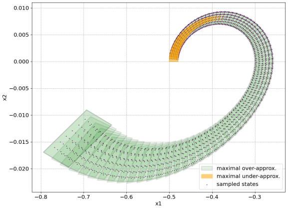
(a) Verisig 2.0 **(b)** ReachNNStar **(c)** RINO

Figure 2: B1 sigmoid

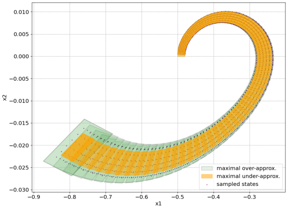
Mountain Car



(a) Mountain car problem



(b) Continuous-time MC



(c) Discrete-time MC

Loss of accuracy for under-approximation in the continuous-time case to be investigated...

Backup Slides

From range projection to joint inner range

Product of 1-dim approximations as n-dim approximation?

- Products of 1-dim over-approx. are (possibly imprecise) n-dim over-approx.
- Generally false for under-approximations! Take $(z_1, z_2) = (f_1(x_1, x_2), f_2(x_1, x_2))$ and

$$\forall z_1 \in \boldsymbol{z}_1, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_1 = f_1(x)$$

$$\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_2 = f_2(x)$$

  Does not imply $\forall z_1 \in \boldsymbol{z}_1$ and $\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1$ and $\exists x_2 \in \boldsymbol{x}_2$ such that $z = f(x)$.

From range projection to joint inner range

Product of 1-dim approximations as n-dim approximation?

- Products of 1-dim over-approx. are (possibly imprecise) n-dim over-approx.
- Generally false for under-approximations! Take $(z_1, z_2) = (f_1(x_1, x_2), f_2(x_1, x_2))$ and

$$\forall z_1 \in \boldsymbol{z}_1, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_1 = f_1(x)$$

$$\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_2 = f_2(x)$$

Does not imply $\forall z_1 \in \boldsymbol{z}_1$ and $\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1$ and $\exists x_2 \in \boldsymbol{x}_2$ such that $z = f(x)$.

A solution (can be generalized to n-dim)

Suppose we can compute $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ with continuous selections x_2 and x_1 such that

$$\forall z_1 \in \boldsymbol{z}_1, \forall x_1 \in \boldsymbol{x}_1, \exists x_2 \in \boldsymbol{x}_2,\ z_1 = f_1(x)$$

$$\forall z_2 \in \boldsymbol{z}_2, \forall x_2 \in \boldsymbol{x}_2, \exists x_1 \in \boldsymbol{x}_1,\ z_2 = f_2(x)$$

From range projection to joint inner range

Product of 1-dim approximations as n-dim approximation?

- Products of 1-dim over-approx. are (possibly imprecise) n-dim over-approx.
- Generally false for under-approximations! Take $(z_1, z_2) = (f_1(x_1, x_2), f_2(x_1, x_2))$ and

$$\forall z_1 \in \boldsymbol{z}_1, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_1 = f_1(x)$$

$$\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1,\ \exists x_2 \in \boldsymbol{x}_2,\ z_2 = f_2(x)$$

Does not imply $\forall z_1 \in \boldsymbol{z}_1$ and $\forall z_2 \in \boldsymbol{z}_2, \exists x_1 \in \boldsymbol{x}_1$ and $\exists x_2 \in \boldsymbol{x}_2$ such that $z = f(x)$.

A solution (can be generalized to n-dim)

Suppose we can compute $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ with continuous selections x_2 and x_1 such that

$$\forall z_1 \in \boldsymbol{z}_1, \forall x_1 \in \boldsymbol{x}_1, \exists x_2 \in \boldsymbol{x}_2,\ z_1 = f_1(x) \text{under-approx of robust (to } x_1\text{) range of } f_1$$

$$\forall z_2 \in \boldsymbol{z}_2, \forall x_2 \in \boldsymbol{x}_2, \exists x_1 \in \boldsymbol{x}_1,\ z_2 = f_2(x) \text{under-approx of robust (to } x_2\text{) range of } f_2$$

By Brouwer fixpoint thm: $\boxed{\boldsymbol{z}_1 \times \boldsymbol{z}_2 \subseteq range(f, \boldsymbol{x}_1 \times \boldsymbol{x}_2)}$ (box / parallelepiped by preconditioning)

Robustly reachable sets

Robust range: states reachable whatever the disturbances on components $w \in \boldsymbol{x}_A$

$$\mathrm{range}_{\mathcal{A}\mathcal{E}}(f, \boldsymbol{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{E}}) = \{z \mid \forall w \in \boldsymbol{x}_{\mathcal{A}}, \exists u \in \boldsymbol{x}_{\mathcal{E}},\ z = f(w, u)\} \subseteq \mathrm{range}(f, \boldsymbol{x})$$

Robustly reachable sets

Robust range: states reachable whatever the disturbances on components $w \in \boldsymbol{x}_A$

$$\text{range}_{\mathcal{A}\mathcal{E}}(f, \boldsymbol{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{E}}) = \{z \,|\, \forall w \in \boldsymbol{x}_{\mathcal{A}}, \exists u \in \boldsymbol{x}_{\mathcal{E}},\ z = f(w,u)\} \subseteq \text{range}(f, \boldsymbol{x})$$

A particular case of robust reachability for dynamical systems with disturbances/inputs

$$(S_c)\begin{cases} \dot{x}(t) = f(x(t), u(t)) \\ x(0) \in \boldsymbol{x}_0, u(t) \in \mathbb{U} \subseteq \mathbb{R}^p \end{cases} \qquad (S_d)\begin{cases} x^{k+1} = f(x^k, u^k) \\ x^0 \in \boldsymbol{x}^0, u(k) \in \mathbb{U} \subseteq \mathbb{R}^p \end{cases} \qquad \text{flow } \varphi^f(t; x_0, u)$$

Sets reachable robustly to disturbances on components u_A:

$$R^f_{\mathcal{A}\mathcal{E}}(t; \boldsymbol{x}_0, \mathbb{U}) = \{x \in \mathcal{D} \,|\, \forall u_A \in \mathbb{U}_{\mathbb{A}}, \exists u_E \in \mathbb{U}_{\mathbb{E}}, \exists x_0 \in \boldsymbol{x}_0,\ x = \varphi^f(t; x_0, u_A, u_E)\}$$

- u_A can be seen as disturbance, u_E as control
- (classical) maximal reachability for $\mathbb{U}_{\mathbb{A}} = \emptyset$, minimal reachability for $\mathbb{U}_{\mathbb{E}} = \emptyset$

Robust Mean Value theorem

Similar to the generalized interval mean-value theorem, but with adversarial terms

- $f : \mathbb{R}^m \to \mathbb{R}$ be continuously differentiable, $\boldsymbol{x} = \boldsymbol{x}_{\mathcal{A}} \times \boldsymbol{x}_{\mathcal{E}}$ initial box
- $\{|\nabla_u f(w,u)|\ , w \in \boldsymbol{x}_{\mathcal{A}},\ u \in \boldsymbol{x}_{\mathcal{E}}\} \subseteq \boldsymbol{\nabla}_u$ and $\{\left|\nabla_w f(w, x^0_{\mathcal{E}})\right|\ , w \in \boldsymbol{x}_{\mathcal{A}}\} \subseteq \boldsymbol{\nabla}_w$

Then:

$$\mathrm{range}_{\mathcal{A}\mathcal{E}}(f, \boldsymbol{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{E}}) \subseteq [\underline{f^0} - \langle \overline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle + \langle \underline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle, \overline{f^0} + \langle \overline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle - \langle \underline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle]$$

$$[\overline{f^0} - \langle \underline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle + \langle \overline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle, \underline{f^0} + \langle \underline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle - \langle \overline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle] \subseteq \mathrm{range}_{\mathcal{A}\mathcal{E}}(f, \boldsymbol{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{E}})$$

Intuition:

- Control $u \in \boldsymbol{x}_{\mathcal{E}}$ acts positively on the (exact) range width : widens the over (resp. under) approximation by $\langle \overline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle[-1, 1]$ (resp. $\langle \underline{\nabla}_u, r(\boldsymbol{x}_{\mathcal{E}})\rangle[-1, 1]$)
- Disturbance $w \in \boldsymbol{x}_{\mathcal{A}}$ acts as an adversary: shrinks down the over (resp. under) approximation by $\langle \underline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle[-1, 1]$ (resp. by $\langle \overline{\nabla}_w, r(\boldsymbol{x}_{\mathcal{A}})\rangle[-1, 1]$)

Example in 2-D

$$f(x) = (5x_1^2 + x_2^2 - 2x_1x_2 - 4, x_1^2 + 5x_2^2 - 2x_1x_2 - 4)^\mathsf{T} \text{ for } \boldsymbol{x} = [0.9, 1.1]^2$$

Using $f(1,1) = 0$, $\nabla f(\boldsymbol{x}) \subseteq \begin{pmatrix} [6.8, 9.2] & [-0.4, 0.4] \\ [-0.4, 0.4] & [6.8, 9.2] \end{pmatrix}$ thus $\mid \nabla f(\boldsymbol{x}) \mid \subseteq \begin{pmatrix} [6.8, 9.2] & [0, 0.4] \\ [0, 0.4] & [6.8, 9.2] \end{pmatrix}$

Example in 2-D

$$f(x) = (5x_1^2 + x_2^2 - 2x_1x_2 - 4, x_1^2 + 5x_2^2 - 2x_1x_2 - 4)^\mathsf{T} \text{ for } \boldsymbol{x} = [0.9, 1.1]^2$$

Using $f(1,1) = 0$, $\nabla f(\boldsymbol{x}) \subseteq \begin{pmatrix} [6.8, 9.2] & [-0.4, 0.4] \\ [-0.4, 0.4] & [6.8, 9.2] \end{pmatrix}$ thus $|\nabla f(\boldsymbol{x})| \subseteq \begin{pmatrix} [6.8, 9.2] & [0, 0.4] \\ [0, 0.4] & [6.8, 9.2] \end{pmatrix}$

1-D mean-value approximations

$\text{range}(f, \boldsymbol{x}) \subseteq [-0.96, 0.96]^2$ e.g. $\text{range}(f_1, \boldsymbol{x}) \subseteq$ $0 + (9.2 \times 0.1 + 0.4 \times 0.1)[-1, 1]$

$\boxed{[-0.68, 0.68] \subseteq \text{range}(f_1, \boldsymbol{x})}$

as $(0 + 0.68 \times 0.1 + 0 \times 0.1)[-1, 1] \subseteq$ $\text{range}(f_1, \boldsymbol{x})$

$\boxed{[-0.68, 0.68] \subseteq \text{range}(f_2, \boldsymbol{x})}$ similarly

Example in 2-D

$$f(x) = (5x_1^2 + x_2^2 - 2x_1x_2 - 4, x_1^2 + 5x_2^2 - 2x_1x_2 - 4)^\intercal \text{ for } \boldsymbol{x} = [0.9, 1.1]^2$$

Using $f(1,1) = 0$, $\nabla f(\boldsymbol{x}) \subseteq \begin{pmatrix} [6.8, 9.2] & [-0.4, 0.4] \\ [-0.4, 0.4] & [6.8, 9.2] \end{pmatrix}$ thus $| \nabla f(\boldsymbol{x}) | \subseteq \begin{pmatrix} [6.8, 9.2] & [0, 0.4] \\ [0, 0.4] & [6.8, 9.2] \end{pmatrix}$

1-D mean-value approximations

$\mathrm{range}(f, \boldsymbol{x}) \subseteq [-0.96, 0.96]^2$

$\boxed{[-0.68, 0.68] \subseteq \mathrm{range}(f_1, \boldsymbol{x})}$

$\boxed{[-0.68, 0.68] \subseteq \mathrm{range}(f_2, \boldsymbol{x})}$

2-D under-approximation by robust range

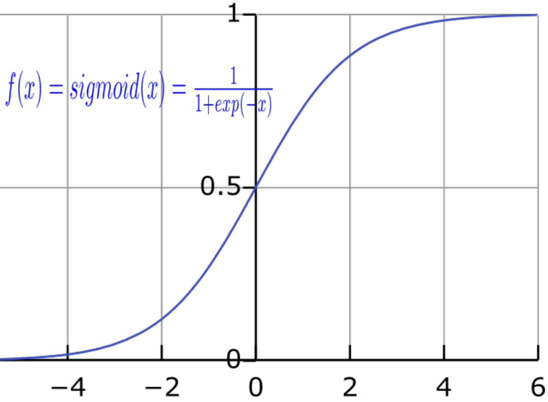
- We obtain $\boxed{[-0.64, 0.64]^2 \subseteq \mathrm{range}(f, \boldsymbol{x})}$ by

$$\forall z_1 \in \boldsymbol{z}_1, \forall x_2 \in \boldsymbol{x}_2, \exists x_1 \in \boldsymbol{x}_1, z_1 = f_1(\boldsymbol{x})$$
$$\forall z_2 \in \boldsymbol{z}_2, \forall x_1 \in \boldsymbol{x}_1, \exists x_2 \in \boldsymbol{x}_2, z_2 = f_2(\boldsymbol{x})$$

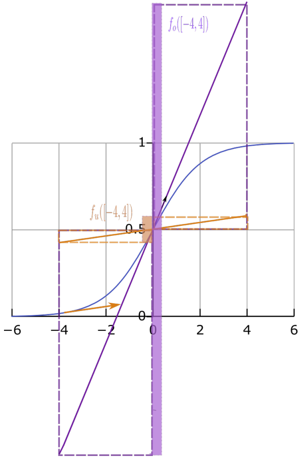
- e.g. for z_1 (similar for z_2):
  $f_1(1,1) +$
  $[-6.8 \times 0.1 + 0.4 \times 0.1, 6.8 \times 0.1 - 0.4 \times 0.1] =$
  $[-0.64, 0.64] \subseteq \mathrm{range}_{AE}(f_1, \boldsymbol{x}, 2)$

Approximating the range of the sigmoid function

range(f,[-4,4])?

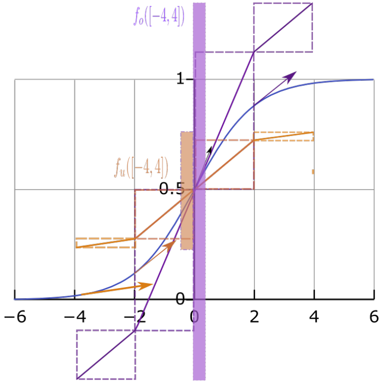


Approximating the range of the sigmoid function



- Not so accurate/satisfying...
- First natural idea: input domain partition? Costly and convex union of the under-approximating boxes is in general not an under-approximation of range$(f, \boldsymbol{x})$

Refinement by local quadrature



Mean-value extension is an interval abstraction of

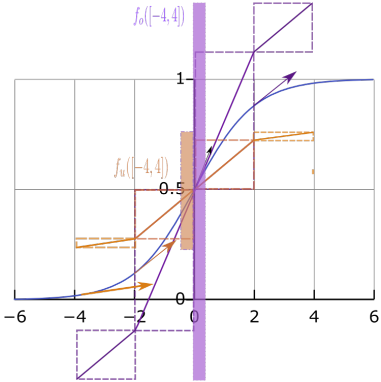
$f(x) = f(x_0) + \int_{x_0}^{x} f'(x)dx$

Use a partition $\boldsymbol{x} = \boldsymbol{x}_1 \cup \boldsymbol{x}_2$ to refine:

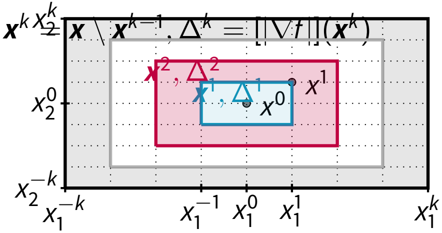
$$f^0 + \langle \underline{\nabla}^1, dx^1 \rangle [-1,1] + \langle \underline{\nabla}^2, dx^2 \rangle [-1,1] \subseteq \mathrm{range}(f, \boldsymbol{x}^1 \cup \boldsymbol{x}^2)$$

$$\mathrm{range}(f, \boldsymbol{x}^1 \cup \boldsymbol{x}^2) \subseteq f^0 + \langle \overline{\nabla}^1, dx^1 \rangle [-1,1] + \langle \overline{\nabla}^2, dx^2 \rangle [-1,1]$$

Refinement by local quadrature



Generalizes to more partitions and n dimensions.



Higher-order AE extensions

Theorem

Let g be an elementary[1] approximation function for f, s.t.

$$\forall w \in \mathbf{x}_{\mathcal{A}},\ \forall u \in \mathbf{x}_{\mathcal{E}},\ \exists \xi \in \mathbf{e},\ f(w,u) = g(w,u,\xi)$$

Then any under-approx $\mathcal{I}_g$ (resp. over-approx $\mathcal{O}_g$) of the range of g robust to $x_{\mathcal{A}}$ and ξ is an under-approx (resp. over-approx) of the range of f robust to $x_{\mathcal{A}}$, i.e.

$$\mathcal{I}_g \subseteq range_{\mathcal{A}\mathcal{E}}(f, \mathbf{x}_{\mathcal{A}}, \boldsymbol{x}_{\mathcal{E}}) \subseteq \mathcal{O}_g$$

Typically, $g(w,u,\xi)$ Taylor expansion of f (with $x = (w,u)$ and ξ from Lagrange remainder):

$$g(x,\xi) = f(x^0) + \sum_{i=1}^{n} \frac{(x - x^0)^i}{i!} D^i f(x^0) + D^{n+1} f(\xi) \frac{(x-x^0)^{n+1}}{(n+1)!}$$

Higher-order AE extensions

Theorem

- *Let g be an elementary function* $g(w,u,\xi) = \alpha(w,u) + \beta(w,u,\xi)$
- $\mathcal{I}_\alpha$ *under-approx of the range of* α *robust to* w, $\mathcal{O}_\beta$ *over-approx of the range of* β

Then the range of g robust with to $w \in \mathbf{x}_\mathcal{A}$ *and* $\xi \in \mathbf{x}$ *is under-approximated by*

$$\mathcal{I}_g = [\underline{\mathcal{I}_\alpha} + \overline{\mathcal{O}}_\beta, \overline{\mathcal{I}}_\alpha + \underline{\mathcal{O}_\beta}] \subseteq range_{\mathcal{A}\mathcal{E}}(f, \mathbf{x}_\mathcal{A}, \mathbf{x}_\mathcal{E})u$$

Typically, $g(w,u,\xi)$ Taylor expansion of f:

$$g(x,\xi) = \underbrace{f(x^0) + \sum_{i=1}^{n} \frac{(x-x^0)^i}{i!} D^i f(x^0)}_{\alpha(x)} + \underbrace{D^{n+1} f(\xi) \frac{(x-x^0)^{n+1}}{(n+1)!}}_{\beta(x,\xi)}$$

- Easily applicable for $n = 1$ (linear expression can be exactly evaluated)

Application to the reachability of discrete-time systems

Application to reachability of discrete-time systems

Algorithm 1: requires propagating n-dim under-approx. at each step

Iteratively compute function image, with as input, the previously computed approximations (under and over-approximations I^k and O^k of the reachable set $\boldsymbol{z}^k$):

$$\begin{cases} I^0 = \boldsymbol{z}^0, \; O^0 = \boldsymbol{z}^0 \\ I^{k+1} = \mathcal{I}(f, I^k, \pi), \; O^{k+1} = \mathcal{O}(f, O^k, \pi) \end{cases}$$

- n-dimensional range under-approximation can be source of loss of precision

Algorithm 2: propagates only over-approx. of range and Jacobian of iterated loop body f

for k from 0 to $K-1$ **do**

$I^{k+1} := \mathcal{I}(f^{k+1}, \boldsymbol{z}^0, \pi), O^{k+1} := \mathcal{O}(f^{k+1}, \boldsymbol{z}^O, \pi)$

end for

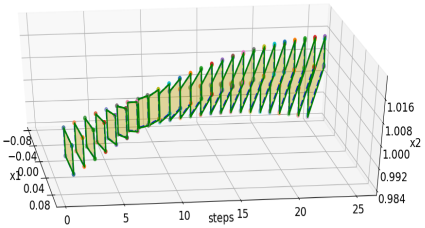
- generally more costly and more precise than Algo 1 (differentiation of the iterated function)

Test model

Model

$$x_1^{k+1} = x_1^k + (0.5(x_1^k)^2 - 0.5(x_2^k)^2)\Delta$$
$$x_2^{k+1} = x_2^k + 2x_1^k x_2^k \Delta$$

with $x_1^0 \in [0.05, 0.1], x_2^0 \in [0.99, 1.00]$ and $\Delta = 0.01$.



Under- (yellow) and over-approximated (green) reachable sets over time up to 25 steps with Algorithm 1, skewed boxes (0.02s computation time)

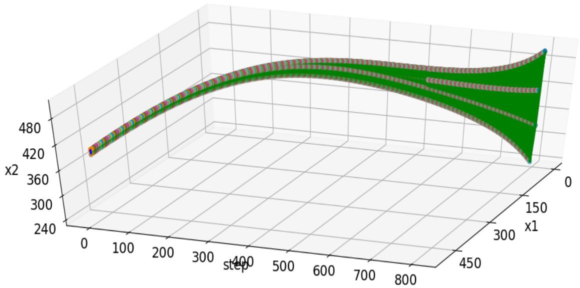
Honeybees Site Choice Model [Dreossi et al. (2016)]

$$x_1^{k+1} = x_1^k - (\beta_1 x_1^k x_2^k + \beta_2 x_1^k x_3^k)\Delta$$
$$x_2^{k+1} = x_2^k + (\beta_1 x_1^k x_2^k - \gamma x_2^k + \delta \beta_1 x_2^k x_4^k$$
$$x_3^{k+1} = x_3^k + (\beta_2 x_1^k x_3^k - \gamma x_3^k + \delta \beta_2 x_3^k x_5^k$$
$$x_4^{k+1} = x_4^k + (\gamma x_2^k - \delta \beta_1 x_2^k x_4^k - \alpha \beta_2 x_3^k$$
$$x_5^{k+1} = x_5^k + (\gamma x_3^k - \delta \beta_2 x_3^k x_5^k - \alpha \beta_1 x_2^k$$

$x_1^0 = 500, x_2^0 \in [390, 400], x_3^0 \in [90,$
$x_4^0 = x_5^0 = 0$ and $\beta_1 = \beta_2 = 0.001, \gamma$
$\delta = 0.5, \alpha = 0.7$, and $\Delta = 0.01$.

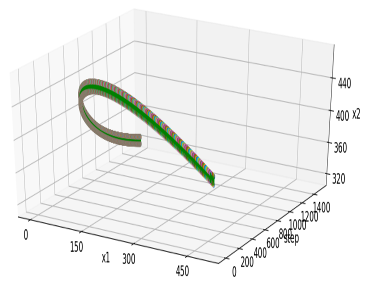
Algorithm 1

Only 1.7s analysis time, but imprecise (800 steps here, later diverges))

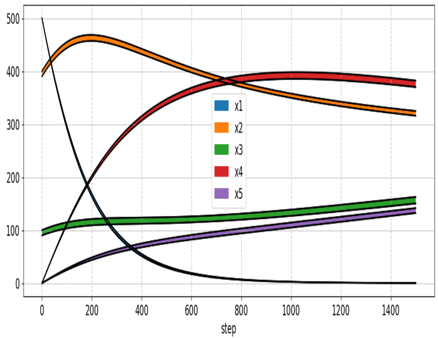


Honeybees Site Choice Model [Dreossi et al. (2016)]

Algorithm 2 (57s analysis time, 1500 steps)



Joint range $(x_1, x2)(k)$



Projected approximations (filled region is under-approx, plain black line is over-approx)

(slightly faster and tighter than Dreossi 2016 for over-approx while also providing under-approx)