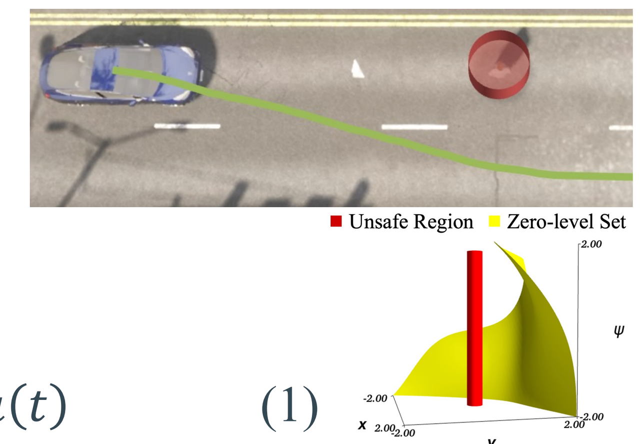
SEEV: Synthesis with Efficient Exact Verification for ReLU Neural Barrier Functions

Hongchao Zhang*[1], Zhizhen Qin*[2], Sicun Gao[2], Andrew Clark[1]
[1] Washington University in St. Louis, [2] University of California, San Diego
* Equal Contribution

Background

Safe Region: $\mathcal{C} = \{x: h(x) \geq 0\} \subseteq \mathcal{X}$
Positive invariance: $x(t) \in \mathcal{C} \subseteq \mathcal{X}$ for all $t \geq 0$, if $x(0) \in \mathcal{C}$.
Safety: positive invariance of a given safety region $\mathcal{C}$.



Dynamics: $\dot{x}(t) = f(x(t)) + g(x(t))u(t)$ (1)

Control Barrier Function (CBF) $b(x)$ is a smooth function that evaluates the 'safety' of the system. Let $\mathcal{D} = \{x: b(x) \geq 0\} \subseteq \mathcal{C}$.

Function $b(x)$ is a CBF if there exist $\mathcal{U}$ and class-κ function α, s.t.

$$\frac{\partial b}{\partial x}\left(f(x(t)) + g(x(t))u(t)\right) \geq -\alpha(b(x(t))) \quad (2)$$

However, u does not exist if $\frac{\partial b}{\partial x} g(x(t)) = 0$ and $\frac{\partial b}{\partial x} f(x(t)) < 0$.

Exact Verification of ReLU NCBFs [1]

Nagumo's Theorem The closed set $\mathcal{D}$ is positive invariant iff, whenever boundary states $x \in \partial\mathcal{D}$, $u \in \mathcal{U}$ satisfies

$$\frac{\partial b}{\partial x}\left(f(x) + g(x)u\right) \geq 0.$$

Intuition: Piece-wise Linearity of ReLU

Problem Studied

Given a system (1), synthesize a ReLU NCBF $b_\theta(x)$ s.t.

- Feasible: for all $x \in \mathcal{D}$, there exist u satisfying (2)
- Correct: $\mathcal{D} \subseteq \mathcal{C}$ (safe region)

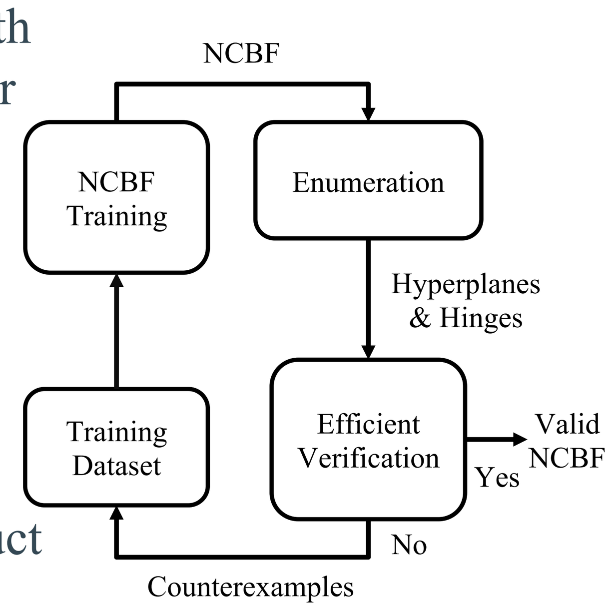
Challenge 1: Scalability for high-dimensional systems and deep NNs.

Challenge 2: Synthesize NCBFs satisfying conditions

- Loss may not converge to zero
- Hard to obtain a formal verification

Contributions

1. Propose a framework for Synthesis with Efficient Exact Verification (SEEV) for ReLU NCBFs.
2. Develop a training procedure that reduces the number of segments that must be verified.
3. Construct hierarchical verification that efficiently enumerate hyperplanes, exploit sufficient conditions and conduct exact verification.

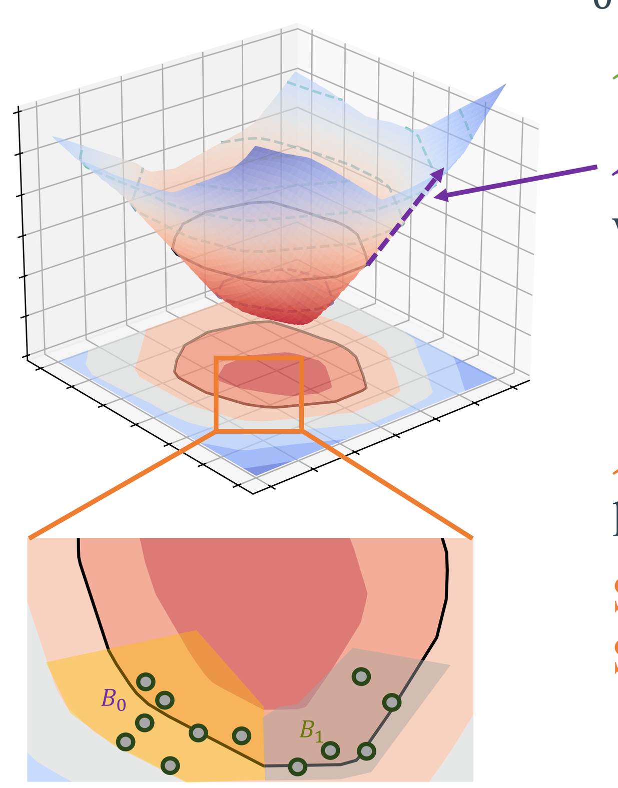


Synthesis with Efficient Exact Verification (SEEV)

Goal: Synthesize NCBFs to be feasible and correct.

Training dataset $\mathcal{T}$: initialized by uniform sampling over $\mathcal{X}$.

$$\min_\theta \lambda_{\mathcal{B}} \mathcal{L}_{\mathcal{B}}(\mathcal{T}) + \lambda_f \mathcal{L}_f(\mathcal{T}) + \lambda_c \mathcal{L}_c(\mathcal{T})$$



$\mathcal{L}_c$ correctness regularizer enforces the correctness of the NCBF.

$\mathcal{L}_f$ feasibility regularizer $\mathcal{L}_f = \|u - \pi_{nom}(x)\|_2^2 + r$ where r is the slack variable for the safety filter [2]

$$\min_u \|u - \pi_{\text{nom}}(x)\|_2^2$$
$$\text{s.t. } \mathcal{W}(\mathbf{S}_i)^T (f(\hat{x}) + g(\hat{x})u) + r \geq 0$$

$\mathcal{L}_{\mathcal{B}}$ Boundary activation regularizer limits the number of hyperplanes & hinges by penalizing the dissimilarity

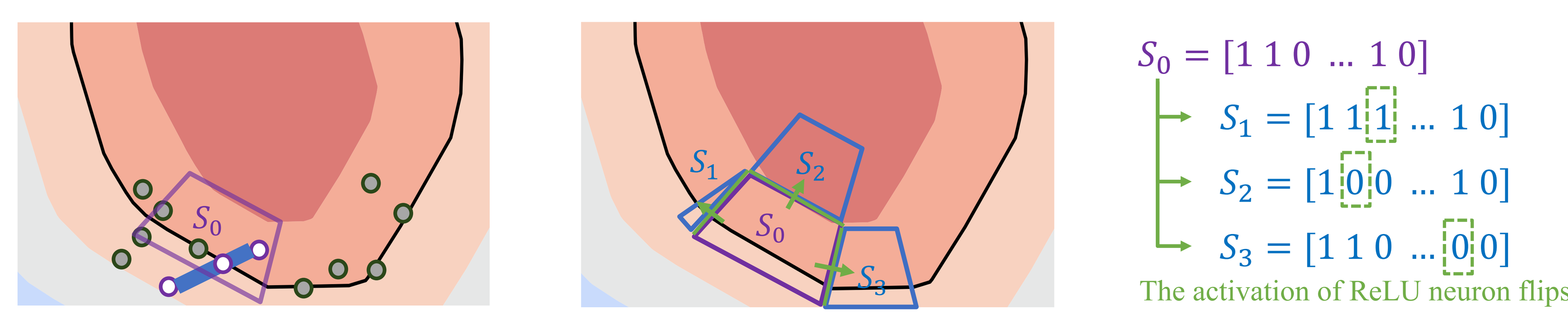
Step 1 Identify boundary samples; Step 2 Clustering;
Step 3 Penalize dissimilarity

$$\mathcal{L}_{\mathcal{B}} = \frac{1}{N_{\mathbf{B}}} \sum_{\mathbf{B}_i \in \mathcal{B}} \frac{1}{|\mathcal{T}_{\mathbf{B}_i}|^2} \sum_{\hat{x}_i, \hat{x}_j \in \mathcal{T}_{\mathbf{B}_i}} \left\|\phi_{\sigma_k}(x_i) - \phi_{\sigma_k}(x_j)\right\|_2^2$$

Efficient Exact Verification for ReLU NBFs

How to compute the $\partial b/\partial x$ efficiently?

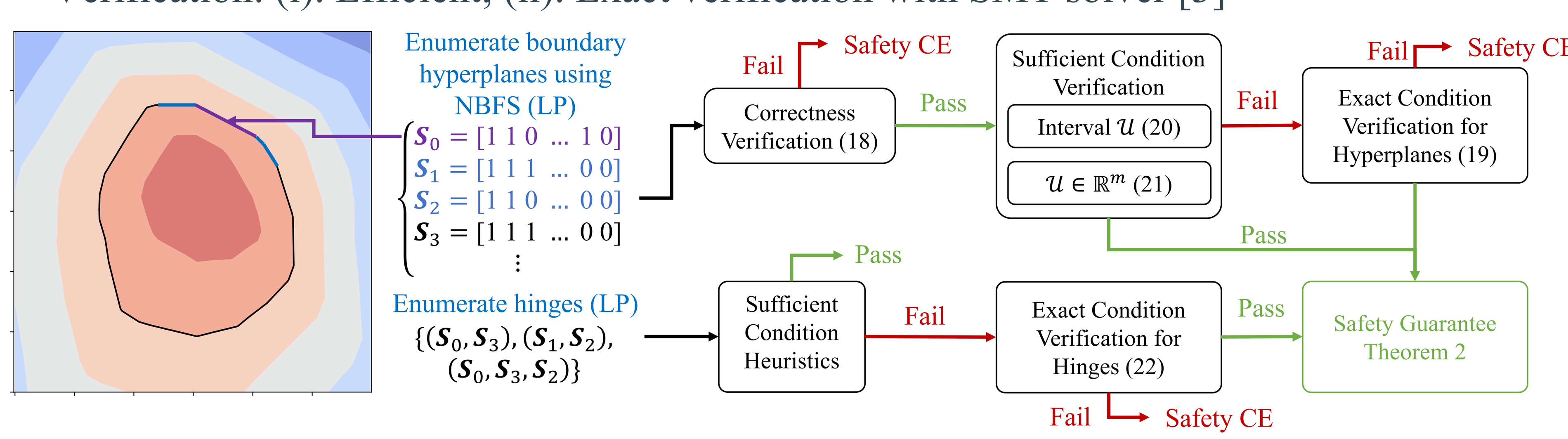
- Derivative of a NCBF can be characterized by activation sets $\mathbf{S}$. (Enumerate $\mathbf{S}$)
- Derivative of a NCBF in one activation set $\mathbf{S}$ is linear. (Linear Program)



Binary Search Find S_0 containing $b(x) = 0$ $\xrightarrow{S_0}$ Search neighbors in a BFS manner $\xrightarrow{\{S_1, S_2, \dots, S_m\}}$ Find the sets of S and $\mathcal{X}(S)$

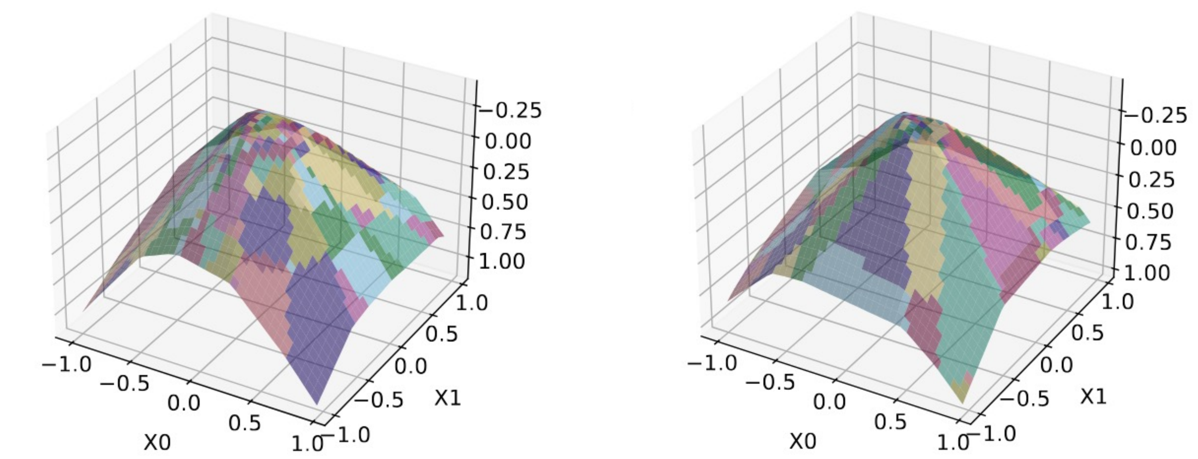
Features:

- Enumeration: (i). Only rely on linear program; (ii). CPU only; (iii) Multi-process enabled
- Verification: (i). Efficient; (ii). Exact verification with SMT solver [3]



Experiments

We consider the Darboux, Obstacle Avoidance (OA), hi-ord$_8$ and Spacecraft Rendezvous (SR) problems and compared our approach with SMT-based verifiers.



(a) $r = 0$ (b) $r = 50$

Table1: Comparison of N the number of boundary hyperplanes and $\mathcal{C}$ coverage of the safe region $\mathcal{D}$ of NCBF trained with (r) and without (o) boundary hyperplane regularizer

Case	n	L	M	N_o	C_o	$N_{r=1}$	$\rho_{r=1}$	$N_{r=10}$	$\rho_{r=10}$	$N_{r=50}$	$\rho_{r=50}$
OA	3	2	8	26	89.46%	25	0.996	23.3	0.994	**13.3**	1.006
	3	2	16	116	83.74%	119	1.012	111	1.005	**98**	1.055
	3	4	8	40	91.94%	38	0.988	36	0.993	**13**	0.937
	3	4	16	156	87.81%	170	0.971	147	1.003	**64**	1.038
SR	6	2	8	2868	98.58%	2753	1	1559	1	**418**	1
	6	4	8	6371	98.64%	6218	1	3055	1	**627**	1
	6	2	16	N/A	N/A	204175	N/A	68783	N/A	**13930**	N/A

Table2: Comparison of verification run-time of NCBF in seconds.

Case	n	L	M	N	t_h	t_g	SEEV	Baseline [23]	dReal	Z3
Darboux	2	2	256	15	2.5s	0	2.5s	315s	>3h	>3h
	2	2	512	15	3.3s	0	3.3s	631s	>3h	>3h
OA	3	2	16	86	0.41s	0	0.41s	16.0s	>3h	>3h
	3	4	8	15	0.39	0	0.39	16.1s	>3h	>3h
	3	4	16	136	0.65s	0	0.65s	36.7s	>3h	>3h
	3	1	128	5778	20.6s	0	20.6s	207s	>3h	>3h
hi-ord$_8$	8	2	8	73	0.54s	0	0.54s	>3h	>3h	>3h
	8	2	16	3048	11.8s	0	11.8s	>3h	>3h	>3h
	8	4	16	3984	22.4s	0	22.4s	>3h	>3h	>3h
SR	6	2	8	2200	7.1s	2.7s	9.8s	179s	UTD	UTD
	6	4	8	4918	45.8s	14.3s	60.1s	298.7s	UTD	UTD

SEEV outperforms the LiRPA-based method proposed in the baseline [23]

SOTA SMT-based Methods are not directly applicable

[1] Zhang, Hongchao, et al. "Exact verification of relu neural control barrier functions." Advances in neural information processing systems 36 (2023): 5685-5705.
[2] Dawson, Charles, Sicun Gao, and Chuchu Fan. "Safe control with learned certificates: A survey of neural lyapunov, barrier, and contraction methods for robotics and control." IEEE Transactions on Robotics 39.3 (2023): 1749-1767.
[3] Gao, Sicun, Soonho Kong, and Edmund M. Clarke. "dReal: An SMT solver for nonlinear theories over the reals." International conference on automated deduction. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

Paper Code