Counter-example Guided Abstract Refinement for NN Verification

Stefano Demarchi and Dario Guidotti

Università degli Studi di Genova, DIBRIS (Department of Informatics, Bioengineering, Robotics and Systems Engineering)

1. Context

Neural networks are widely adopted in diverse fields and applications. While they provide fast and useful results, they have been proved to suffer from safety and reliability issues.

- Neural Networks are functions organized in layers
- Connections between nodes represent linear transformations
- Nodes apply non-linear *activation* functions

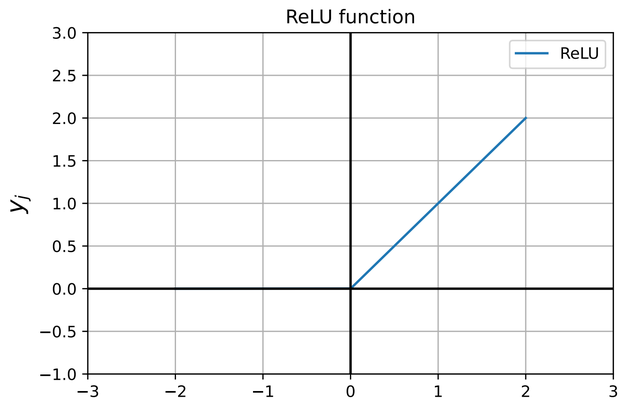

Figure 1: The ReLU activation function, corresponding to $y_j = max(0, z_j)$

2. Abstract analysis

Abstracting the input domains as sets we reason on how the network transforms them, using exact and over-approximated algorithms.

The problem of NN verification is to compute the output reachable set and prove it does not intersect some unsafe zone.

If the over-approximation invalidates the property, we can try to refine the abstraction in order to mitigate the approximation error.

	Algorithm		
	Complete	Approximate	Mixed
Exact output	X		
Combinatorial growth	X		
Quadratic growth			X
Linear growth		X	

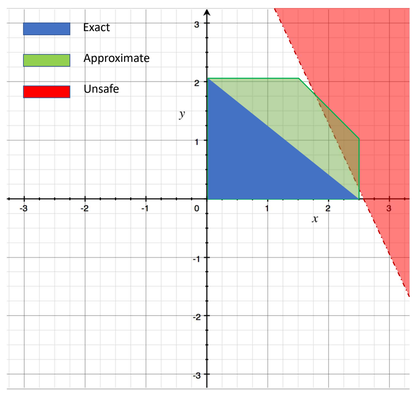

Figure 2: An example of the approximate reachable set (green) violating the safety property (red) while the exact reachable set (blue) is safe.

3. CEGAR

Exact algorithm. If the output reachable set lies in the unsafe zone the property is violated. The counter input set containing all possible inputs in the input set that lead the neural network to unsafe states is

$$\mathbf{C} = \bigcup_{i=1}^{k} (c, V, \overline{P}_i), \overline{P}_i \neq 0$$

Approximate algorithm. The approximation introduces new variables and the inversion of the output set is not possible. We define the *abstract counter output set* (ACOS) the intersection between the output and the unsafe zone. We sample $\hat{y}$ from the ACOS and use a minimization problem for finding a $\hat{x}$ in the input

$$\hat{x} = \min_{x} ||\hat{y} - \nu(x)||_2$$

Relevance. If the search problem fails due to non-convexity, we measure the *relevance* of neurons enhancing it with the areas of approximate ReLUs: the approximation is larger when the sets are widespread.

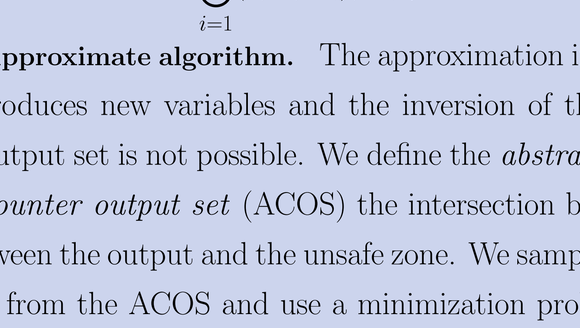

Figure 3: An example of the measure of neuron relevance: tighter bounds produce triangles with smaller areas (left).

4. Experiments and Conclusions

ACAS-Xu benchmarks are a well-known case study in avionics.

PROPERTY	NETWORK	MIXED		CEGAR-PS		CEGAR-mR	
		TIME	VERIFIED	TIME	VERIFIED	TIME	VERIFIED
# 3	1_1	13	T	10	3/10	9	9/10
	1_3	10	T	14	6/10	10	0/10
	2_3	7	T	10	9/10	7	6/10
	4_3	15	T	17	10/10	14	10/10
	5_1	6	T	11	10/10	9	10/10
# 4	1_1	11	T	10	0/10	9	0/10
	1_3	8	T	16	0/10	11	0/10
	3_2	12	T	12	10/10	12	10/10
	4_2	12	T	11	10/10	12	10/10

- The mixed algorithm only considers the approximation areas
- CEGAR-PS computes the product of the relevances and the areas
- CEGAR-mR only considers the relevance

The CEGAR algorithms do not clearly enhance the results, but give us insight on how the neurons behave and add a level of explainability to the verification algorithm.

References

[1] Stefano Demarchi, Dario Guidotti, Andrea Pitto, and Armando Tacchella. Formal Verification of Neural Networks: a Case Study about Adaptive Cruise Control. In *International Conference on Modelling and Simulation*, 2022.

[2] Dario Guidotti, Francesco Leofante, Luca Pulina, and Armando Tacchella. Verification of neural networks: enhancing scalability through pruning. *arXiv preprint arXiv:2003.07636*, 2020.

[3] Dario Guidotti, Luca Pulina, and Armando Tacchella. Never 2.0: Learning, verification and repair of deep neural networks. *arXiv preprint arXiv:2011.09933*, 2020.

[4] Dario Guidotti, Luca Pulina, and Armando Tacchella. pyNeVer: A framework for learning and verification of neural networks. In *International Symposium on Automated Technology for Verification and Analysis*, pages 357–363. Springer, 2021.

[5] Luca Pulina and Armando Tacchella. Never: a tool for artificial neural networks verification. *Annals of Mathematics and Artificial Intelligence*, 62(3-4):403–425, 2011.

[6] Dung Tran. *Verification of Learning-enabled Cyber-Physical Systems*. PhD thesis, Vanderbilt University, 2020.

[7] Hoang-Dung Tran, Diago Manzanas Lopez, Patrick Musau, Xiaodong Yang, Luan Viet Nguyen, Weiming Xiang, and Taylor T Johnson. Star-based reachability analysis of deep neural networks. In *International Symposium on Formal Methods*, pages 670–686. Springer, 2019.