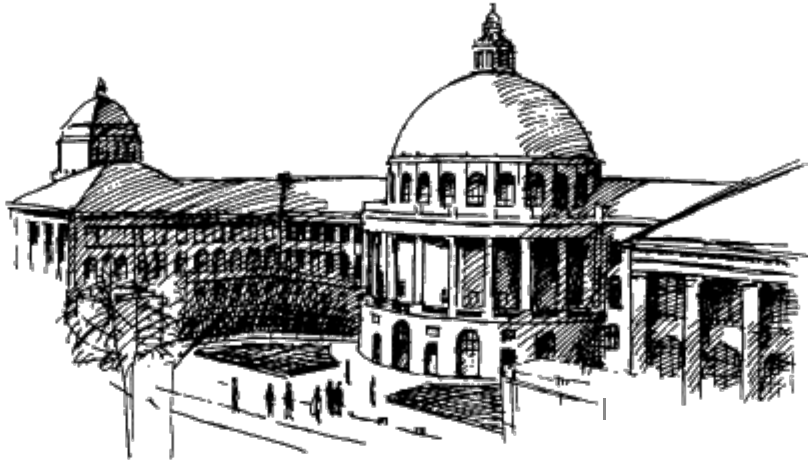
An Abstract Domain for Certifying Neural Networks



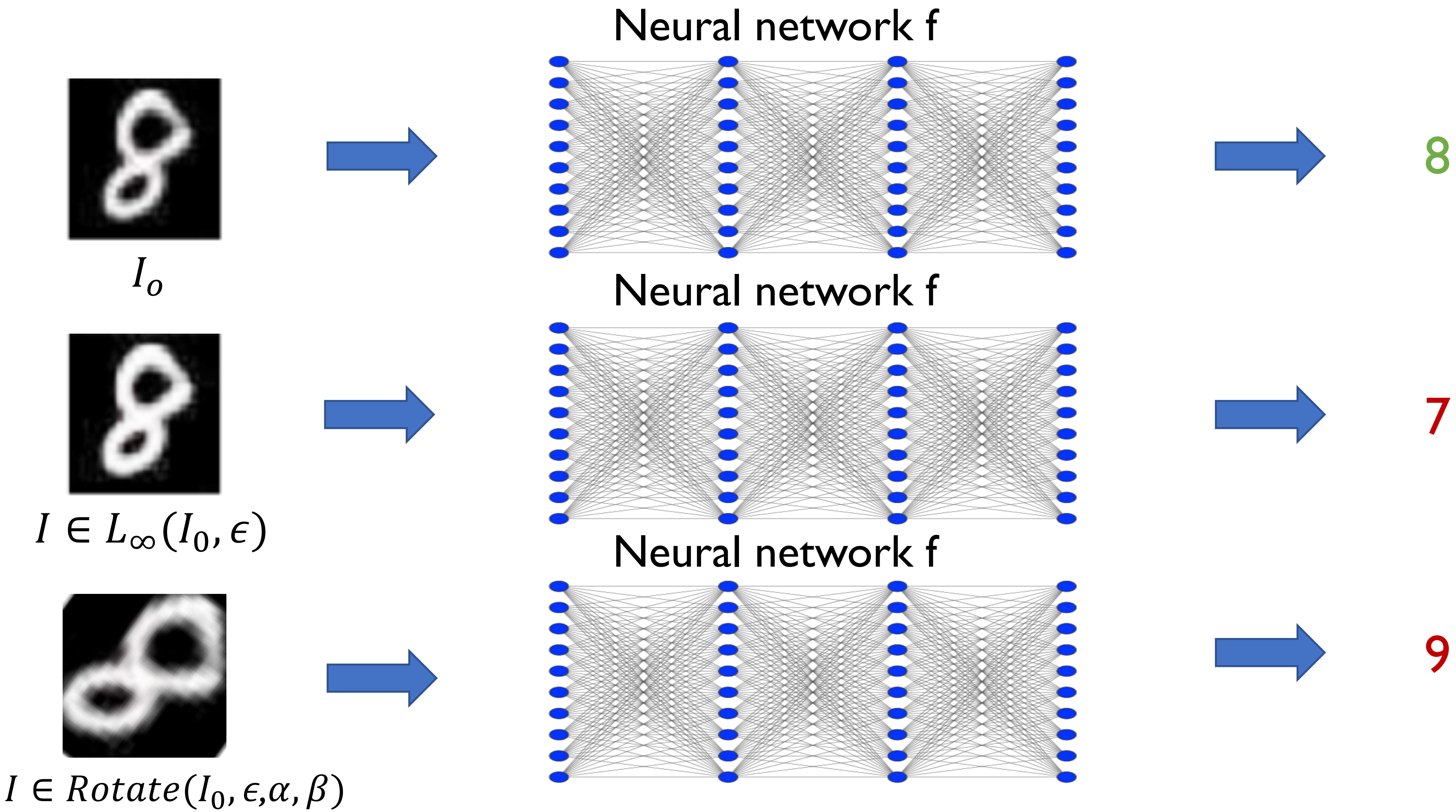

Gagandeep Singh   Timon Gehr   Markus Püschel   Martin Vechev

Department of Computer Science

ETH zürich



Adversarial input perturbations



Neural network robustness

Given:

Neural network $f: \mathbb{R}^m \longrightarrow \mathbb{R}^n$
Perturbation region $\mathcal{R}(I_0, \phi)$

Regions:

$L_\infty(I_0, \epsilon)$: All images I where pixel values in I and I_0 differ by *at most* ϵ

Rotate$(I_0, \epsilon, \alpha, \beta)$: All images I in $L_\infty(I_0, \epsilon)$ rotated by $\theta \in [\alpha, \beta]$

To Prove:

$\forall I \in \mathcal{R}(I_0, \phi). f(c) > f(j)$
where c is the correct output and j is *any* other output

Challenges

The size of $\mathcal{R}(I_0, \phi)$ grows exponentially in the number of pixels:

- cannot compute f(I) for all I separately

Prior Work

- Precise but does not scale:
  - SMT Solving [CAV'17]
  - Input refinement [USENIX'18]
  - Semidefinite relaxations [ICLR'18]
- Scales but imprecise
  - Linear relaxations [ICML'18]
  - Abstract interpretation [S&P'18, NIPS'18]

This work: contributions

A new abstract domain combining floating point Polyhedra with Intervals:

- custom transformers for common functions in neural networks such as affine transforms, ReLU, sigmoid, tanh, and maxpool activations
- scalable and precise analysis

First approach to certify robustness under rotation combined with linear interpolation:

- based on refinement of the abstract input
- $\epsilon = 0.001, \alpha = -45^o, \beta = 65^o$

DeepPoly:

- complete and parallelized end-to-end implementation based on ELINA
- https://github.com/eth-sri/eran

Network	ϵ	NIPS'18	DeepPoly
➢ 6 layers ➢ 3010 units	0.035	proves 21% 15.8 sec	proves 64% 4.8 sec
➢ 6 layers ➢ 34,688 units	0.3	proves 37% 17 sec	proves 43% 88 sec

Our Abstract Domain

Shape: associate a lower polyhedral $a_i^{\leq}$ and an upper polyhedral $a_i^{\geq}$ constraint with each x_i

$$a_i^{\leq}, a_i^{\geq} \in \{x \mapsto v + \sum_{j \in [i-1]} w_j \cdot x_j \mid v \in \mathbb{R} \cup \{-\infty, +\infty\}, w \in \mathbb{R}^{i-1}\} \text{ for } i \in [n]$$

Concretization of abstract element a:

$$\gamma_n(a) = \{x \in \mathbb{R}^n \mid \forall i \in [n].\, a_i^{\leq}(x) \leq x_i \wedge a_i^{\geq}(x) \geq x_i\}$$

Domain invariant: store auxiliary concrete lower and upper bounds l_i, u_i for each x_i

$$\gamma_n(a) \subseteq \times_{i \in [n]} [l_i, u_i]$$

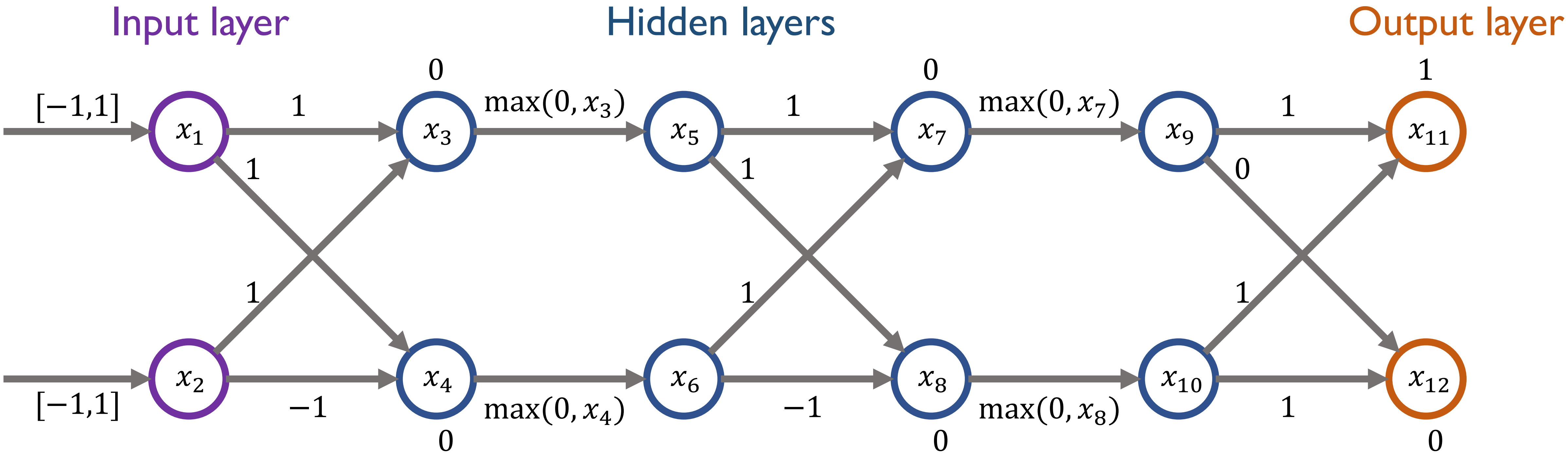
- less precise than Polyhedra, restriction needed to ensure scalability
- captures affine transformation precisely unlike Octagon, TVPI
- custom transformers for ReLU, sigmoid, tanh, and maxpool activations

n: #neurons, m: #constraints

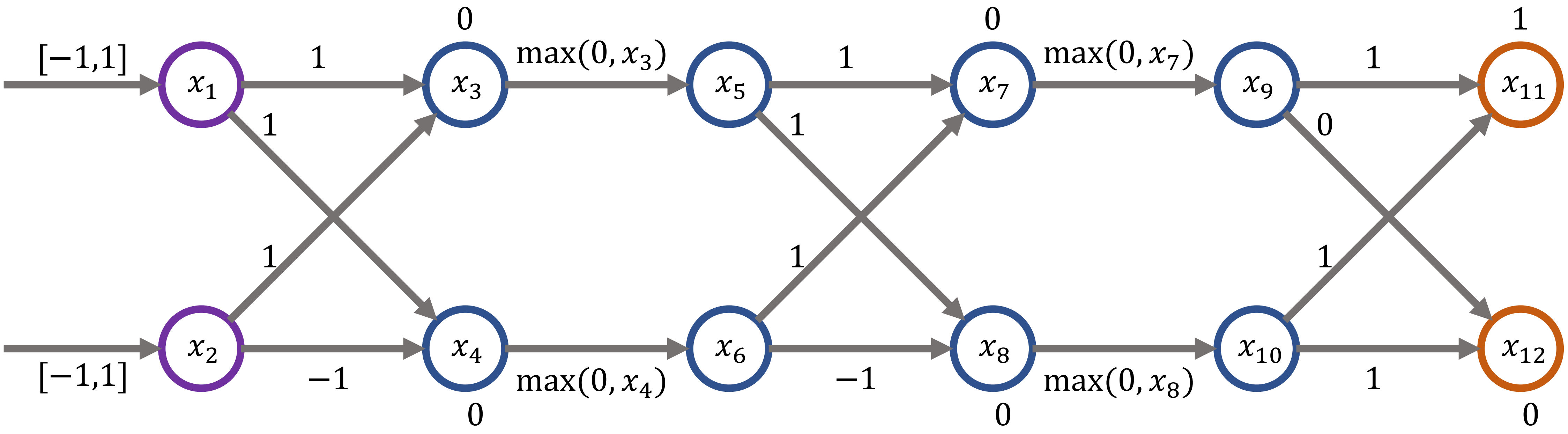
w_{max}: max #neurons in a layer, L: # layers

Transformer	Polyhedra	Our domain
Affine	$O(nm^2)$	$O(w_{max}^2 L)$
ReLU	$O(\exp(n, m))$	$O(1)$

Example: Analysis of a Toy Neural Network



$$\langle x_1 \geq -1, \quad \langle x_3 \geq x_1 + x_2,$$
$$x_1 \leq 1, \quad x_3 \leq x_1 + x_2,$$
$$l_1 = -1, \quad l_3 = -2,$$
$$u_1 = 1\rangle \quad u_3 = 2\rangle$$



$$\langle x_2 \geq -1, \quad \langle x_4 \geq x_1 - x_2,$$
$$x_2 \leq 1, \quad x_4 \leq x_1 - x_2,$$
$$l_2 = -1, \quad l_4 = -2,$$
$$u_2 = 1\rangle \quad u_4 = 2\rangle$$

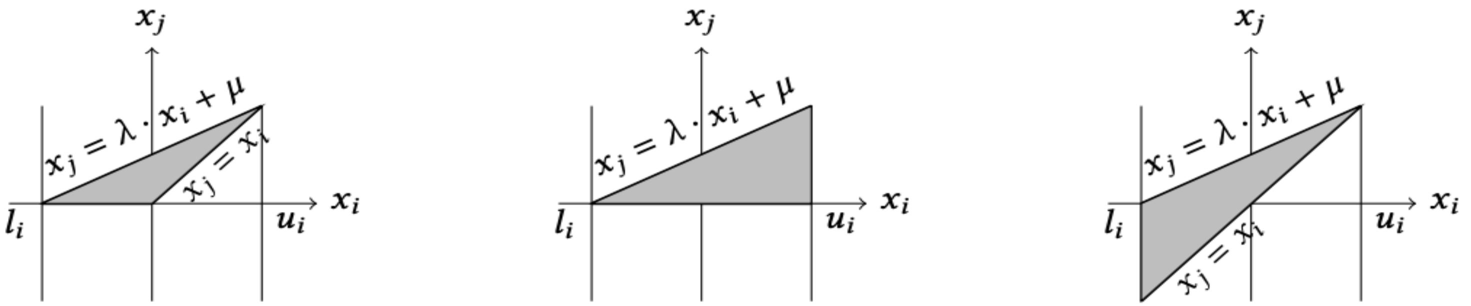
ReLU activation

Pointwise transformer for $x_j := max(0, x_i)$ that uses l_i, u_i

$if\ u_i \leq 0, a_j^{\leq} = a_j^{\geq} = 0, l_j = u_j = 0,$

$if\ l_i \geq 0, a_j^{\leq} = a_j^{\geq} = x_i, l_j = l_i, u_j = u_i,$

$if\ l_i < 0\ and\ u_i > 0$



(a) $x_i \leq x_j, 0 \leq x_j,$ $x_j \leq u_i(x_i - l_i)/(u_i - l_i).$ $l_j = 0, u_j = u_i$

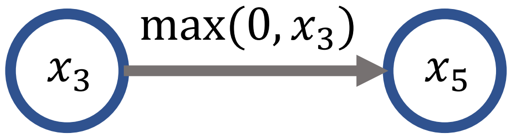
(b) $0 \leq x_j,$ $x_j \leq u_i(x_i - l_i)/(u_i - l_i),$ $l_j = 0,\ u_j = u_i$

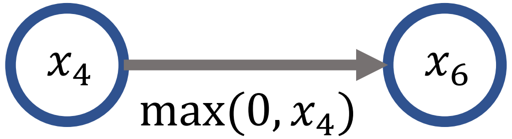
(c) $x_i \leq x_j,$ $x_j \leq u_i(x_i - l_i)/(u_i - l_i),$ $l_j = l_i,\ u_j = u_i$

choose (b) or (c) depending on the area

$\langle x_3 \geq x_1 + x_2,$ $x_3 \leq x_1 + x_2,$ $l_3 = -2,$ $u_3 = 2\rangle$

$\langle x_5 \geq 0,$ $x_5 \leq 0.5 \cdot x_3 + 1,$ $l_5 = 0,$ $u_5 = 2\rangle$





$\langle x_4 \geq x_1 - x_2,$ $x_4 \leq x_1 - x_2,$ $l_4 = -2,$ $u_4 = 2\rangle$

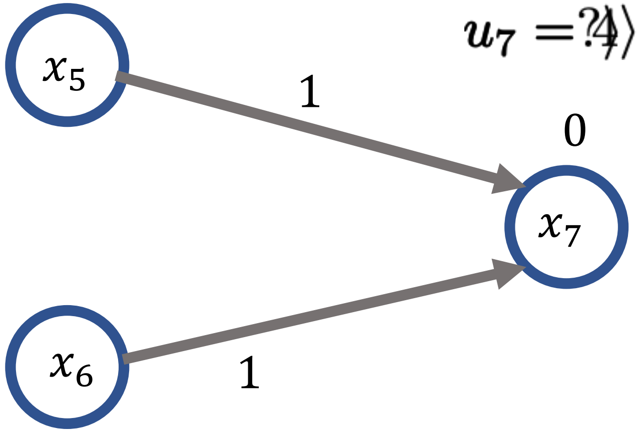
$\langle x_6 \geq 0,$ $x_6 \leq 0.5 \cdot x_4 + 1,$ $l_6 = 0,$ $u_6 = 2\rangle$

Constant runtime

Affine transformation after ReLU

$$\langle x_5 \geq 0,$$
$$x_5 \leq 0.5 \cdot x_3 + 1,$$
$$l_5 = 0,$$
$$u_5 = 2\rangle$$

$$\langle x_7 \geq x_5 + x_6,$$
$$x_7 \leq x_5 + x_6,$$
$$l_7 = ?0,$$
$$u_7 = ?4\rangle$$



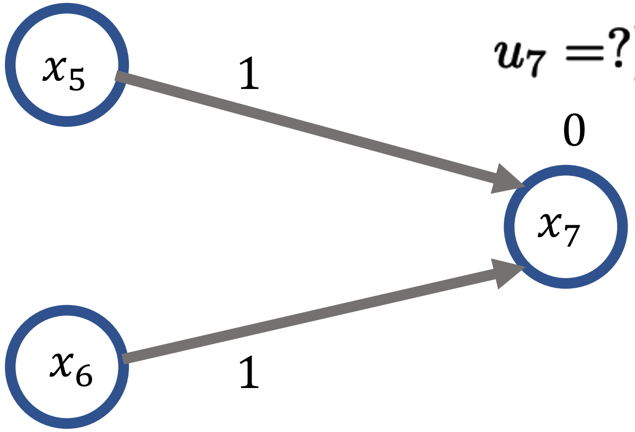
$$\langle x_6 \geq 0,$$
$$x_6 \leq 0.5 \cdot x_4 + 1,$$
$$l_6 = 0,$$
$$u_6 = 2\rangle$$

Imprecise upper bound u_7 by substituting u_5, u_6 for x_5 and x_6 in $a_7^{\geq}$

Backsubstitution

$\langle x_5 \geq 0,$
$x_5 \leq 0.5 \cdot x_3 + 1,$
$l_5 = 0,$
$u_5 = 2\rangle$

$\langle x_7 \geq x_5 + x_6,$
$x_7 \leq 0.5 \cdot x_3 + 0.5 \cdot x_4 + 2,$
$l_7 = ?,$
$u_7 = ?\rangle$



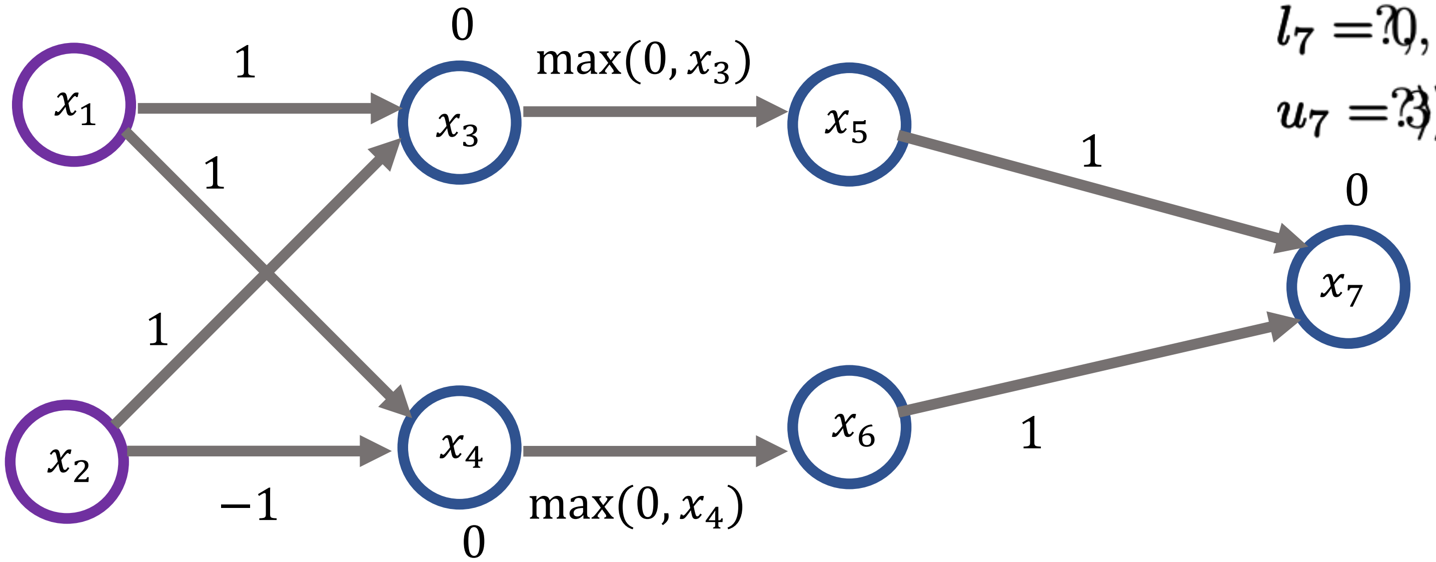
$\langle x_6 \geq 0,$
$x_6 \leq 0.5 \cdot x_4 + 1,$
$l_6 = 0,$
$u_6 = 2\rangle$

$\langle x_1 \ge -1,$
$x_1 \le 1,$
$l_1 = -1,$
$u_1 = 1\rangle$

$\langle x_3 \ge x_1 + x_2,$
$x_3 \le x_1 + x_2,$
$l_3 = -2,$
$u_3 = 2\rangle$

$\langle x_5 \ge 0,$
$x_5 \le 0.5 \cdot x_3 + 1,$
$l_5 = 0,$
$u_5 = 2\rangle$

$\langle x_7 \ge 0,$
$x_7 \le 0.5 \cdot x_3 + 0.5 \cdot x_4 + 2,$
$l_7 = 0,$
$u_7 = 3\rangle$



$\langle x_2 \ge -1,$
$x_2 \le 1,$
$l_2 = -1,$
$u_2 = 1\rangle$

$\langle x_4 \ge x_1 - x_2,$
$x_4 \le x_1 - x_2,$
$l_4 = -2,$
$u_4 = 2\rangle$

$\langle x_6 \ge 0,$
$x_6 \le 0.5 \cdot x_4 + 1,$
$l_6 = 0,$
$u_6 = 2\rangle$

Affine transformation with backsubstitution is pointwise, complexity: $O(w_{max}^2 L)$

$$\langle x_1 \geq -1,\; x_1 \leq 1,\; l_1 = -1,\; u_1 = 1\rangle$$

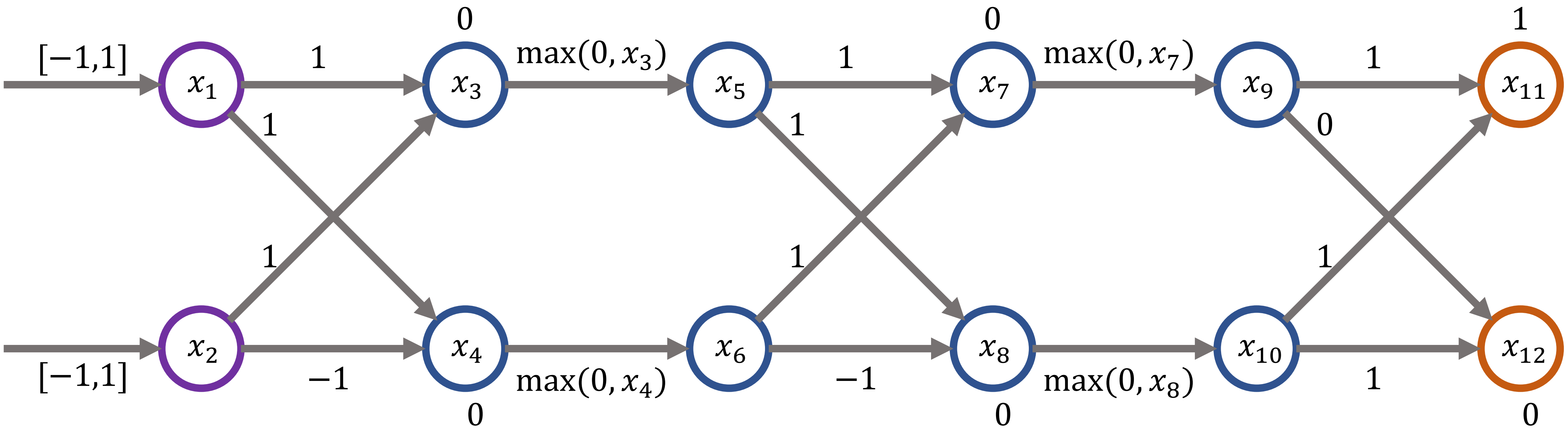
$$\langle x_3 \geq x_1 + x_2,\; x_3 \leq x_1 + x_2,\; l_3 = -2,\; u_3 = 2\rangle$$

$$\langle x_5 \geq 0,\; x_5 \leq 0.5 \cdot x_3 + 1,\; l_5 = 0,\; u_5 = 2\rangle$$

$$\langle x_7 \geq x_5 + x_6,\; x_7 \leq x_5 + x_6,\; l_7 = 0,\; u_7 = 3\rangle$$

$$\langle x_9 \geq x_7,\; x_9 \leq x_7,\; l_9 = 0,\; u_9 = 3\rangle$$

$$\langle x_{11} \geq x_9 + x_{10} + 1,\; x_{11} \leq x_9 + x_{10} + 1,\; l_{11} = 1,\; u_{11} = 5.5\rangle$$



$$\langle x_2 \geq -1,\; x_2 \leq 1,\; l_2 = -1,\; u_2 = 1\rangle$$

$$\langle x_4 \geq x_1 - x_2,\; x_4 \leq x_1 - x_2,\; l_4 = -2,\; u_4 = 2\rangle$$

$$\langle x_6 \geq 0,\; x_6 \leq 0.5 \cdot x_4 + 1,\; l_6 = 0,\; u_6 = 2\rangle$$

$$\langle x_8 \geq x_5 - x_6,\; x_8 \leq x_5 - x_6,\; l_8 = -2,\; u_8 = 2\rangle$$

$$\langle x_{10} \geq 0,\; x_{10} \leq 0.5 \cdot x_8 + 1,\; l_{10} = 0,\; u_{10} = 2\rangle$$

$$\langle x_{12} \geq x_{10},\; x_{11} \leq x_{10},\; l_{12} = 0,\; u_{12} = 2\rangle$$

Checking for robustness

Prove $x_{11} - x_{12} > 0$ for all inputs in $[-1{,}1]\times[-1{,}1]$

$$\begin{aligned} &\langle x_{11} \geq x_9 + x_{10} + 1, \\ &x_{11} \leq x_9 + x_{10} + 1, \\ &l_{11} = 1, \\ &u_{11} = 5.5\rangle \end{aligned} \qquad \begin{aligned} &\langle x_{12} \geq x_{10}, \\ &x_{11} \leq x_{10}, \\ &l_{12} = 0, \\ &u_{12} = 2\rangle \end{aligned}$$

Computing lower bound for $x_{11} - x_{12}$ using l_{11}, u_{12} gives -1 which is an imprecise result

With backsubstitution, one gets 1 as the lower bound for $x_{11} - x_{12}$, proving robustness

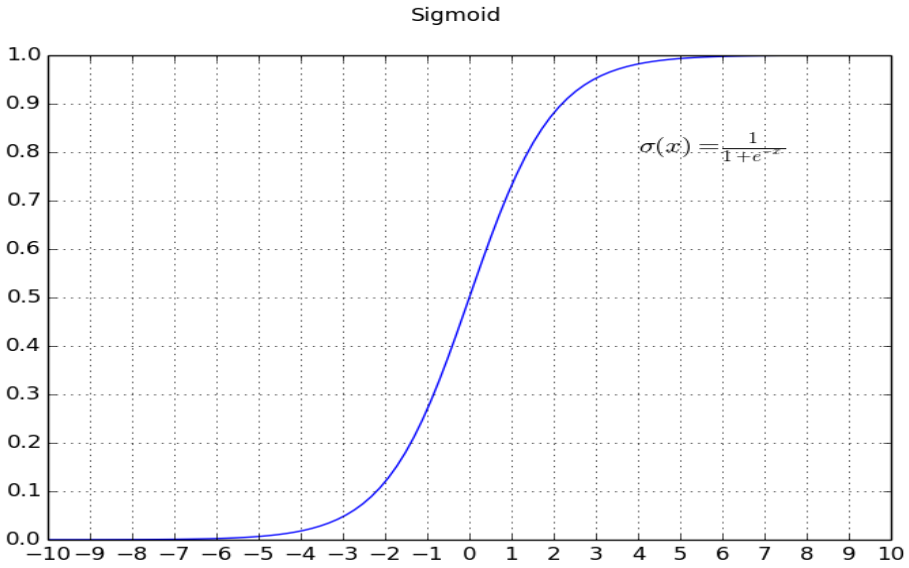
More complex perturbations: rotations

Challenge: *Rotate*$(I_0, \epsilon, \alpha, \beta)$ is non-linear and cannot be captured in our domain unlike $L_\infty(I_0, \epsilon)$

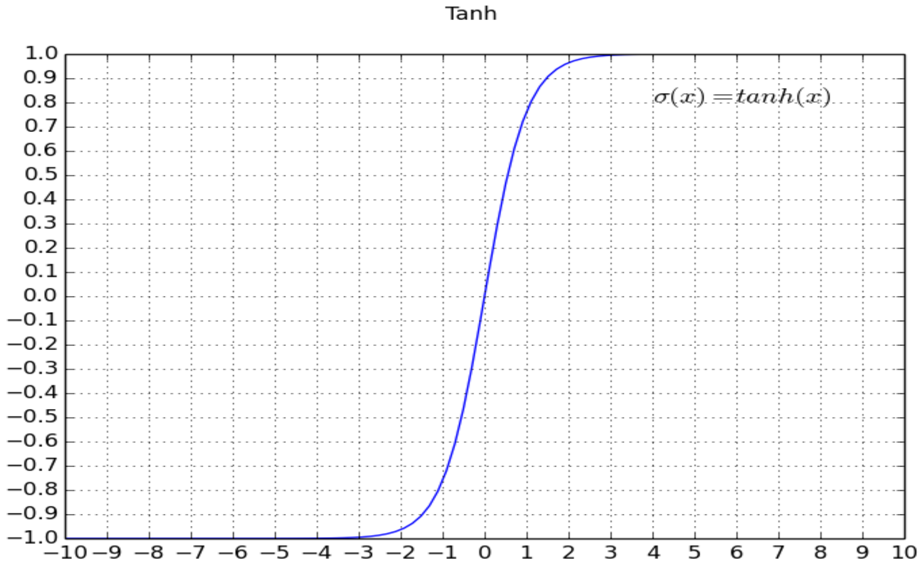
Solution: Over-approximate *Rotate*$(I_0, \epsilon, \alpha, \beta)$ with boxes and use input refinement for precision

Result: Prove robustness for networks under *Rotate*$(I_0, 0.001, -45, 65)$

More in the paper



Sigmoid transformer



Tanh transformer

Maxpool transformer $y := max(x_1, x_2, \ldots, x_r)$

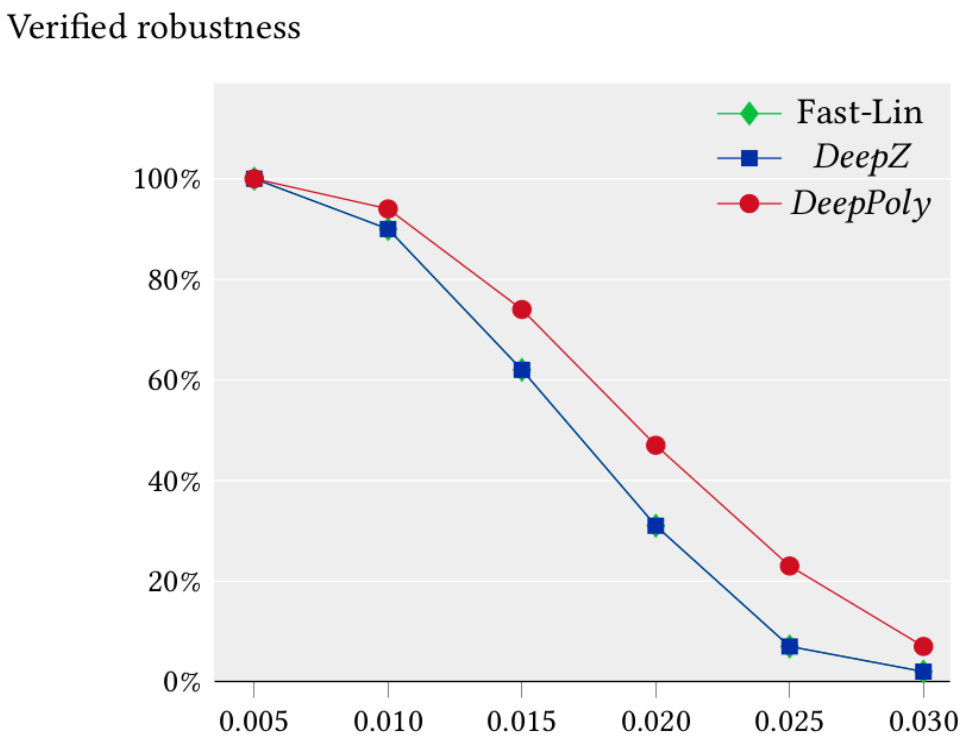
$$a_i^{\leq}, a_i^{\geq} \in \{x \mapsto [v^-, v^+] \oplus_f \sum_{j \in [i-1]} [w_j^-, w_j^+] \otimes_f x_j\}$$

Floating point soundness

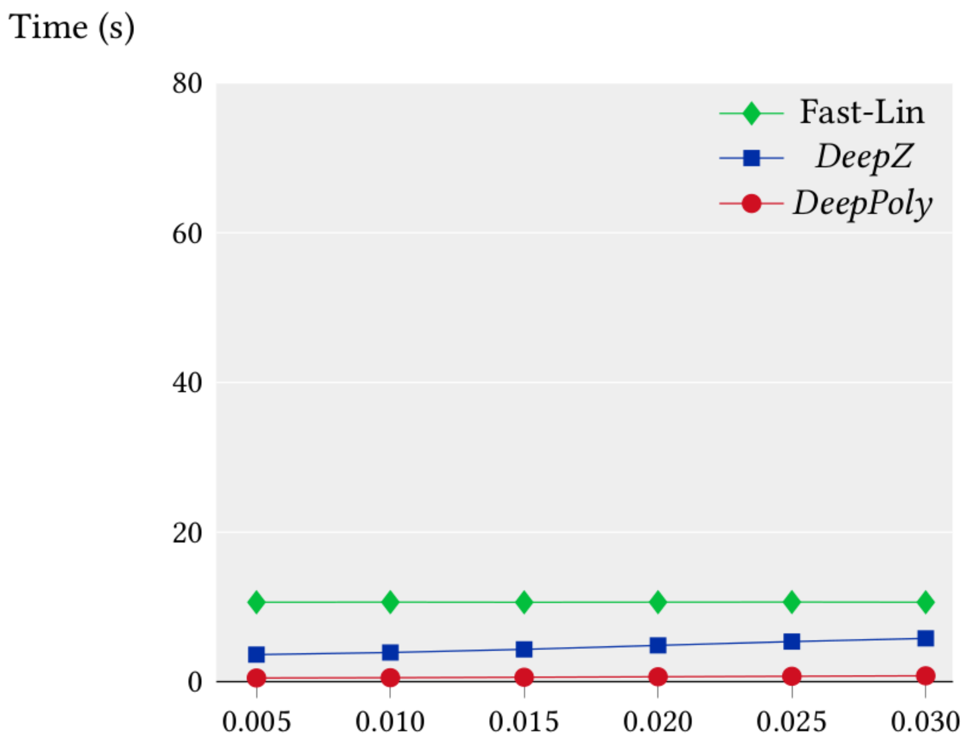
Experimental evaluation

- Neural network architectures:
    - fully connected feedforward (FFNN)
    - convolutional (CNN)
- Training:
    - trained to be robust with DiffAI [ICML'18] and PGD [CVPR'18]
    - without adversarial training
- Datasets:
    - MNIST
    - CIFAR10
- DeepPoly vs. state-of-the-art DeepZ [NIPS'18] and Fast-Lin [ICML'18]

Results

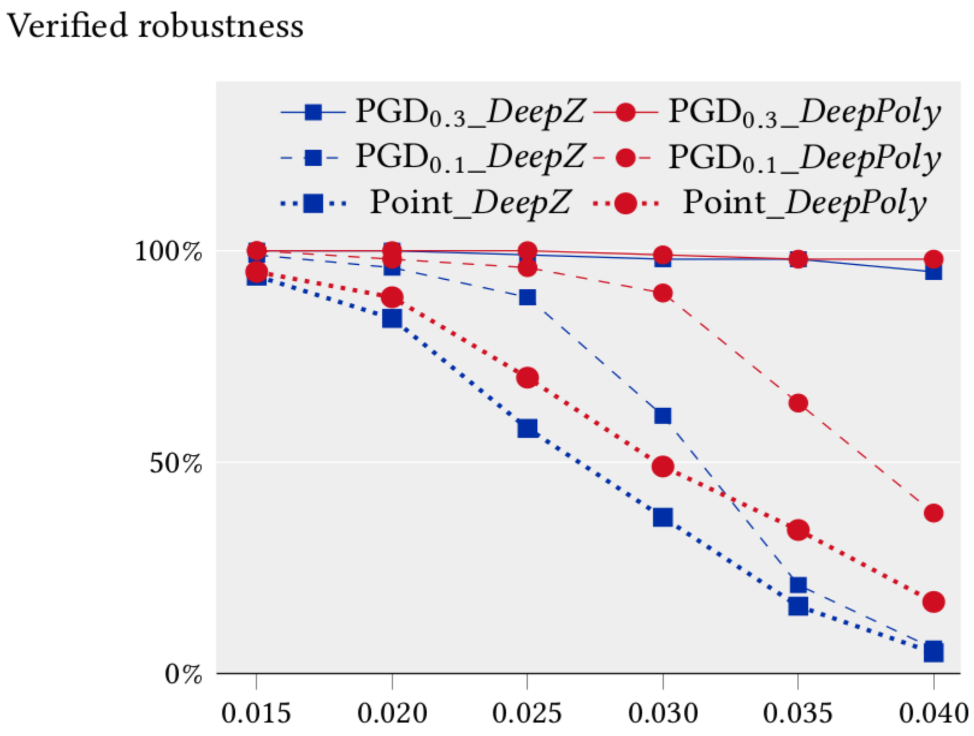


(a) MNIST 6 × 100

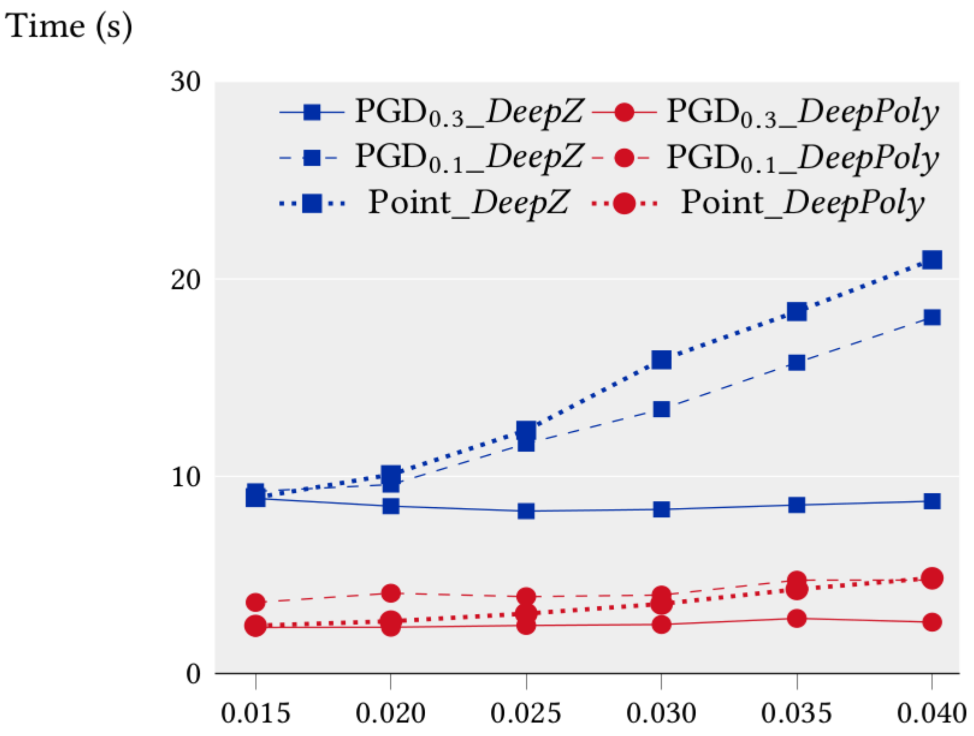


(b) MNIST 6 × 100

MNIST FFNN (3,010 hidden units)

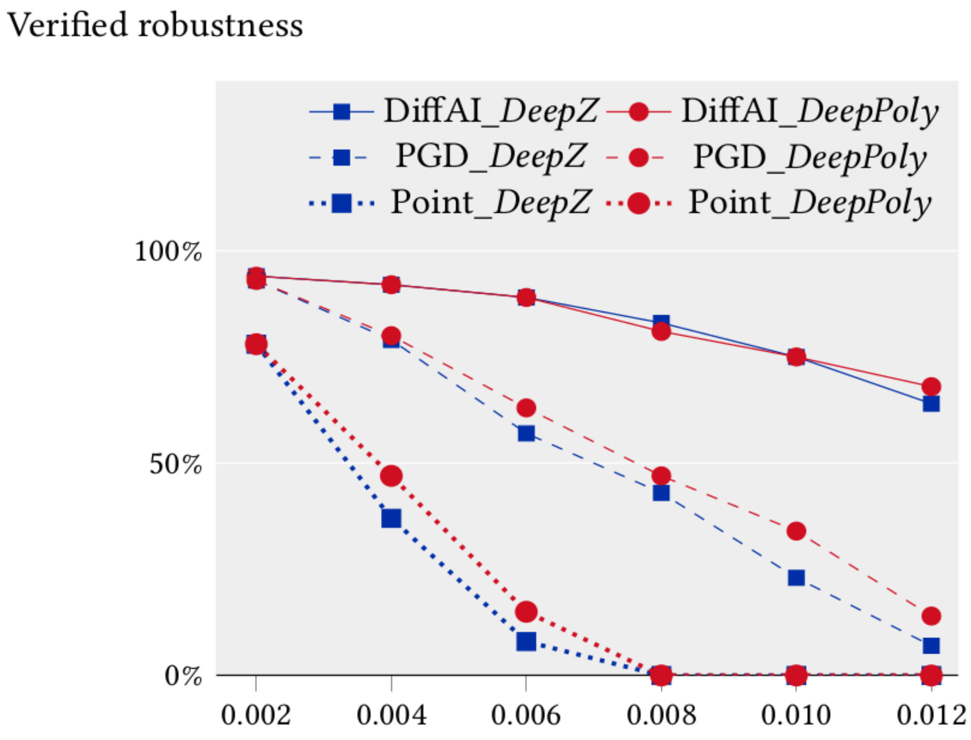


(a) MNIST 6×500 ReLU

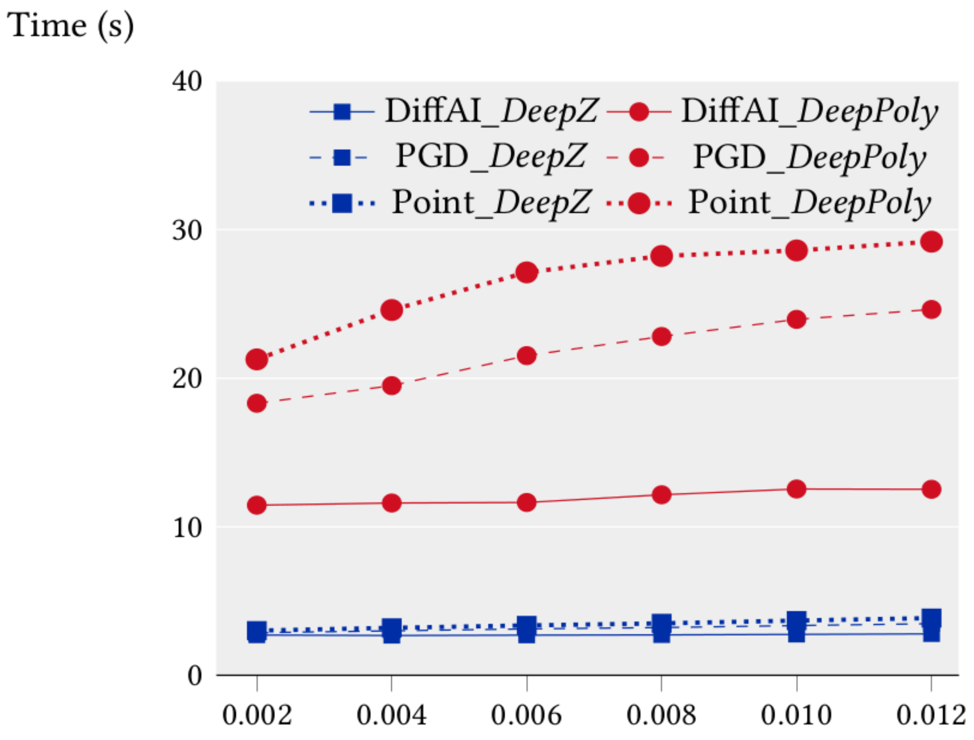


(b) MNIST 6×500 ReLU

CIFAR10 CNNs (4,852 hidden units)



(a) CIFAR10 ConvSmall



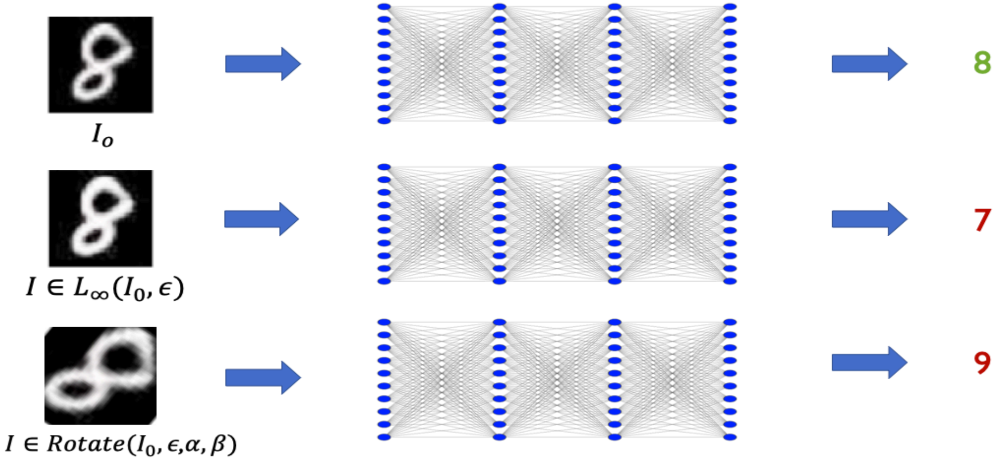
(b) CIFAR10 ConvSmall

Large Defended CNNs
trained via DiffAI [ICML'18]

Dataset	Model	#hidden units	ϵ	%verified robustness		Average runtime (s)	
				DeepZ	DeepPoly	DeepZ	DeepPoly
MNIST	ConvBig	34,688	0.1	97	97	5	50
	ConvBig	34,688	0.2	79	78	7	61
	ConvBig	34,688	0.3	37	43	17	88
	ConvSuper	88,500	0.1	97	97	133	400
CIFAR10	ConvBig	62,464	0.006	50	52	39	322
	ConvBig	62,464	0.008	33	40	46	331

Conclusion

Adversarial input perturbations

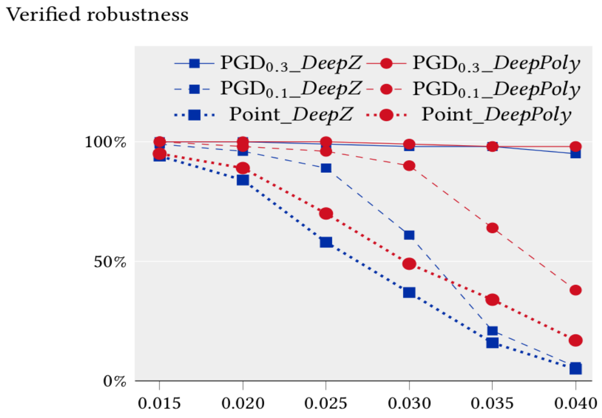


A new abstract domain combining floating point Polyhedra with Intervals:

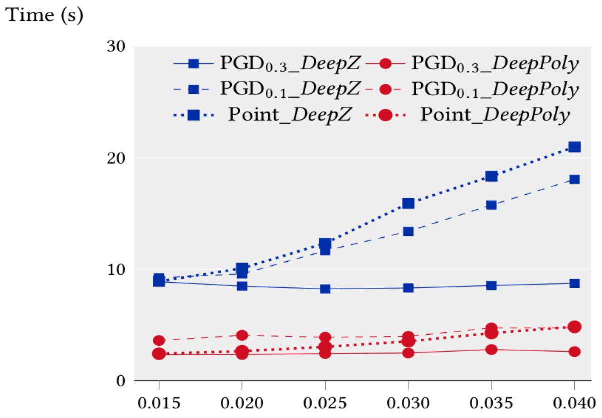
n: #neurons, m: #constraints

w_{max}: max #neurons in a layer, L: # layers

Transformer	Polyhedra	Our domain
Affine	$O(nm^2)$	$O(w_{max}^2 L)$
ReLU	$O(\exp(n, m))$	$O(1)$



(a) MNIST 6×500 ReLU



(b) MNIST 6×500 ReLU

DeepPoly:

- complete and parallelized end-to-end implementation based on ELINA
- https://github.com/eth-sri/eran