



# Differentiable Logic for Correct-by-Construction Neural Networks

Thomas Flinkow

*Department of Computer Science*
*Maynooth University*

Dagstuhl Seminar 26031: *Software Contracts Meet System Contracts*
13th January 2026

# Outline

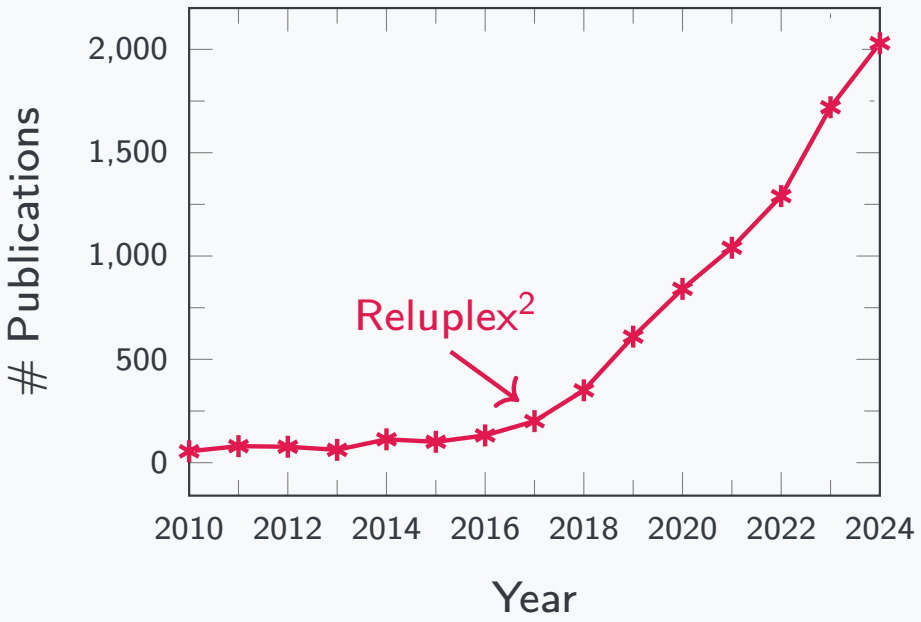
# Why Verify Neural Networks?


**Figure 1:** Stickers placed on a stop sign cause a neural network to misclassify it as *Speed Limit 45 mph*.[1]

[1] K. Eykholt et al. (2018). 'Robust Physical-World Attacks on Deep Learning Visual Classification'. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. DOI: 10.1109/CVPR.2018.00175.

# How to Verify Neural Networks?

Paper titles containing "formal verification" and "neural network" over time




**Community Efforts**

**VNN-COMP** International Verification of Neural Networks Competition (https://sites.google.com/view/vnn2025)

**VNN-LIB** International Standard for the Verification of Neural Networks (https://www.vnnlib.org)

[2] G. Katz et al. (2017). 'Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks'. In: *Computer Aided Verification*. DOI: 10.1007/978-3-319-63387-9_5.

# Training to Satisfy Constraints

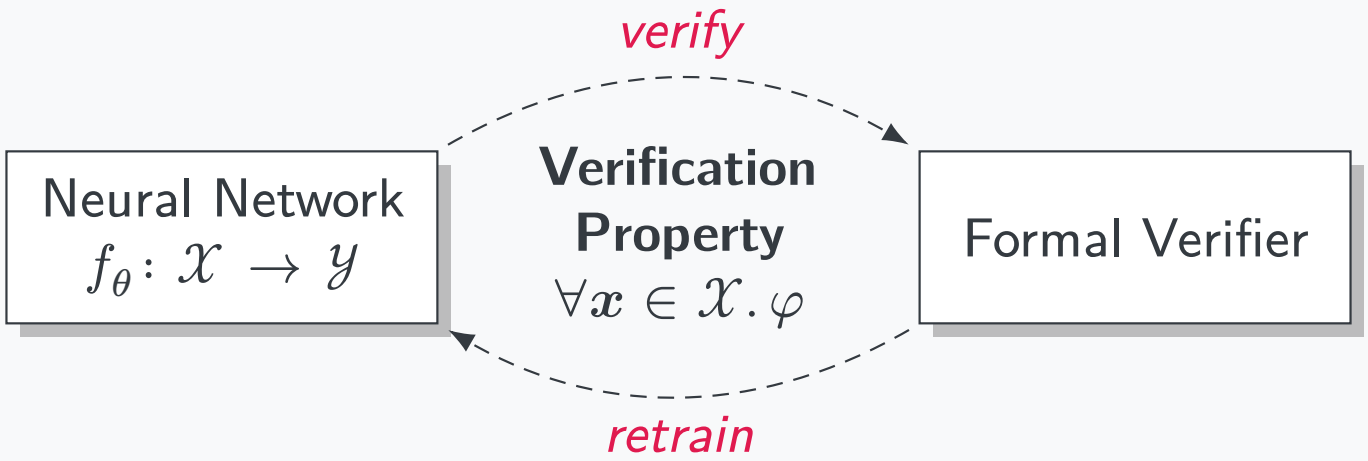



**Figure 2:** The continuous verification cycle.

**Training Process:**

(1) translate $\varphi$ into real-valued loss $[\![\varphi]\!]$, and (2) find a counter-example in $\mathcal{X}$ that violates $\varphi$.

$$(3)\ \text{maximise} \underset{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}{\mathbb{E}} \left[ \min_{\boldsymbol{x}'\in\mathcal{X}} [\![\varphi]\!](\boldsymbol{x}', \boldsymbol{y}; f_\theta) \right].$$

$[\![-]\!]$ is a mapping (called *differentiable logic*) from a logical specification into real-valued loss.

# Example

- **Constraint:** $\forall \boldsymbol{x} \in \mathcal{X}. (5 \leq f(\boldsymbol{x}) \land f(\boldsymbol{x}) \leq 10)$
- **Translate:** (e.g. using DL2[3]: $[\![a \leq b]\!]_{\text{DL2}} = \max(0, a - b)$ and $[\![a \land b]\!]_{\text{DL2}} = a + b$)

$$
\begin{aligned}
&[\![5 \leq f(\boldsymbol{x}) \land f(\boldsymbol{x}) \leq 10]\!]_{\text{DL2}} \\
&[\![5 \leq f(\boldsymbol{x})]\!] + [\![f(\boldsymbol{x}) \leq 10]\!]_{\text{DL2}} \\
&\max(0, 5 - f(\boldsymbol{x})) + \max(0, f(\boldsymbol{x}) - 10)
\end{aligned}
$$

- **Find a counterexample:**

$$x^* = \max_{\boldsymbol{x}' \in \mathcal{X}} [\![5 \leq f(\boldsymbol{x}) \land f(\boldsymbol{x}) \leq 10]\!]_{\text{DL2}}$$

- **Use counterexample in training.**

[3] M. Fischer et al. (2019). 'DL2: Training and Querying Neural Networks with Logic'. In: *Proceedings of the 36th International Conference on Machine Learning*.

# Previous Work: Comparing Differentiable Logics[4]

| Logic | Domain | $[\![T]\!]$ | $[\![F]\!]$ | $[\![\neg x]\!]$ | $[\![x \wedge y]\!]$ | $[\![x \vee y]\!]$ |
|---|---|---|---|---|---|---|
| DL2 | $[0, \infty)$ | $0$ | $\infty$ | undefined | $x + y$ | $xy$ |
| Gödel | $[0, 1]$ | $1$ | $0$ | $1 - x$ | $\min(x, y)$ | $\max(x, y)$ |

**Research Question**

How do existing differentiable logics (DLs) compare in terms of:

- gradients (learning behaviour)?
- logical consistency?
- establishing formal guarantees?

**Findings**

Training with *any* DL yields significantly improved *empirical* constraint satisfaction, but fails to provide (strong) formal guarantees.

[4] T. Flinkow, B. A. Pearlmutter et al. (2025). 'Comparing Differentiable Logics for Learning with Logical Constraints'. In: *Science of Computer Programming*. DOI: 10.1016/j.scico.2025.103280.

# Previous Work: Alsomitra Drone Controller[5]


**(a)** An *Alsomitra macrocarpa* seed. **(b)** The desired linear trajectory.

**Constraint:** *If the drone is above and close to the line, pitching down quickly and moving fast, the network should make it pitch up.*

| Logic | RMSE | CAcc (%) | CSec (%) |
|---|---|---|---|
| Baseline | $3.61 \times 10^{-4}$ | 0.00 | 0.00 |
| DL2 | $1.23 \times 10^{-3}$ | 100.00 | 95.31 |
| Fuzzy logic | $1.16 \times 10^{-3}$ | 100.00 | 92.19 |

[5] C. Kessler et al. (2026). 'Neural Network Verification for Gliding Drone Control: A Case Study'. In: *AI Verification*. DOI: 10.1007/978-3-031-99991-8_9.

# Recent Work (in Progress): Strong(er) Formal Guarantees

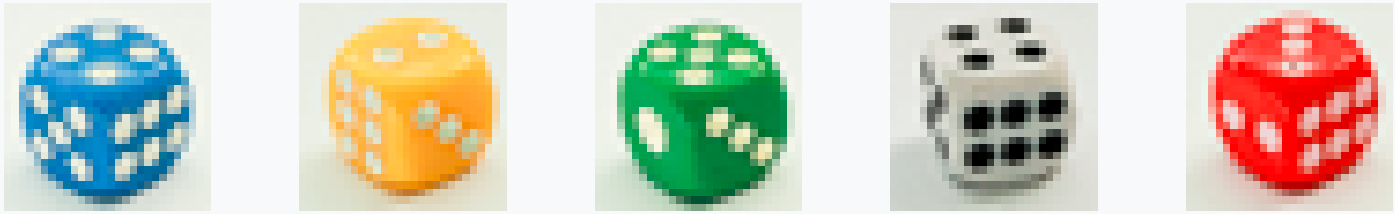
## Issues

- DL2 $[\![-]\!]_{\text{DL2}} : \Phi \to [0, \infty)$ loss is $0$ if the constraint is satisfied.
- Consider the simple constraint $f(\boldsymbol{x}) \leq 5$.
- $[\![a \leq b]\!]_{\text{DL2}} = \max(0, a - b)$ means gradients vanish once $a \leq b$.
- **Not *finding* a counterexample does not mean there is none!**

## Ideas for a new differentiable logic

- Express not just *violation*, but also *satisfaction* of constraints $[\![-]\!]_{\text{Ours}} : \Phi \to (-\infty, \infty)$.
- Smooth connectives $[\![\wedge^s]\!]_{\text{Ours}}$ and $[\![\vee^s]\!]_{\text{Ours}}$ that approximate $\min$ and $\max$ as $s \to \infty$
- Non-vanishing derivatives everywhere. (i.e. make it *easier* to find counterexamples)

# Recent Work (in Progress): Strong(er) Formal Guarantees



**Constraint:** *Predictions must be physically possible: the network should not predict two faces that are on opposite sides of the die.*

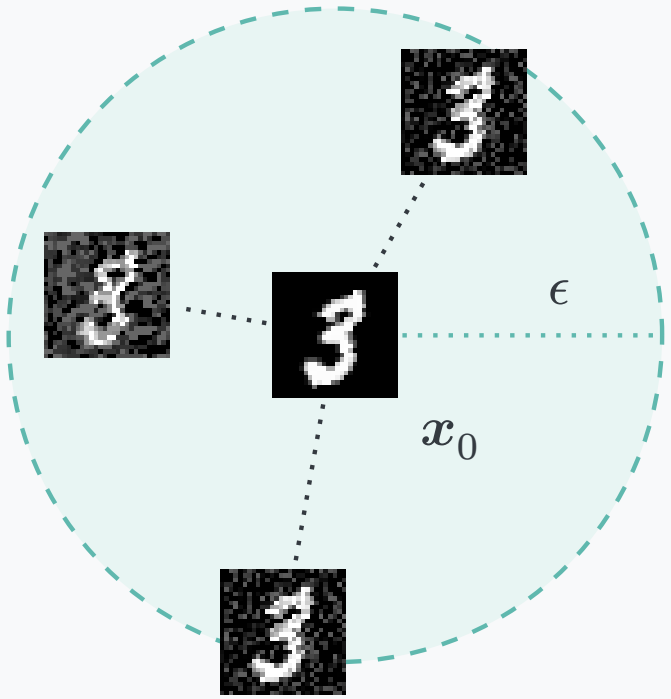
| Logic | PAcc (%) | CAcc (%) | CSec (%) | VSat (%)[a] | |
|---|---|---|---|---|---|
| | | | | $\epsilon = 4/255$ | $\epsilon = 16/255$ |
| Baseline | 85.8 | 100.0 | 2.9 | 25.9 | 0.0 |
| DL2 | 86.0 | 97.1 | 41.2 | $83.8^{(1)}$ | $0.0^{(1)}$ |
| STL | 79.9 | 100.0 | 100.0 | 98.5 | $1.8^{(11)}$ |
| **Ours** | **77.5** | **100.0** | **100.0** | **100.0** | $\mathbf{92.7}^{(5)}$ |

[a] Superscript $^{(k)}$ indicates the inputs that timed out and remain unknown.

# Specifications?

**Local robustness:**

$$\forall \boldsymbol{x}.\ \underbrace{\|\boldsymbol{x} - \boldsymbol{x}_0\| \leq \epsilon}_{\text{for all inputs similar to known input } \boldsymbol{x}_0} \implies \underbrace{f(\boldsymbol{x}) \approx f(\boldsymbol{x}_0)}_{\text{the network behaviour should be similar}}$$

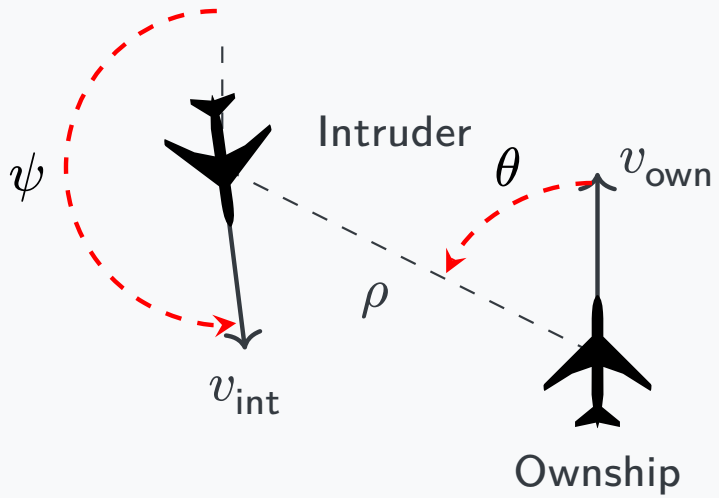



**Problems**

- Natural Language Processing: discrete space, no meaningful sentences in $\epsilon$-ball
- Cyber-physical Systems: low-dimensional input space with semantics (e.g. velocity, distance)

# Component ↔ System Specifications

**Constraint:** 'For an intruder that is *near* and *approaching from the left*, the network should advise "strong right"'.




"*near* and *approaching from the left*":

$$250\,\text{ft} \leq \rho \leq 400\,\text{ft}, \quad 0.2\,\text{rad} \leq \theta \leq 0.4\,\text{rad}, \quad \ldots$$

**Problem:** Verified safe in isolation, unsafe in closed-loop setting![6]

[6] S. Bak et al. (2022). 'Neural Network Compression of ACAS Xu Early Prototype Is Unsafe'. In: *NASA Formal Methods*. DOI: 10.1007/978-3-031-06773-0_15.

# Summary & Conclusion

## 1. General Motivation

Correct-by-construction ML models by specification-driven training with differentiable logic.

## 2. Current Work

A differentiable logic with stronger formal guarantees.

## 3. Future Work

More expressive specifications.


**Thomas Flinkow**

Department of Computer Science
Maynooth University

Email: thomas.flinkow@mu.ie
https://www.cs.nuim.ie/~tflinkow/


Maynooth University
National University of Ireland Maynooth

Taighde Éireann
Research Ireland

**Thank you! Any questions?**


1. T. Flinkow, B. A. Pearlmutter et al. (2025). 'Comparing Differentiable Logics for Learning with Logical Constraints'. In: *Science of Computer Programming* 244, p. 103280. ISSN: 0167-6423. DOI: 10.1016/j.scico.2025.103280
2. T. Flinkow, M. Casadio et al. (2025). *A General Framework for Property-Driven Machine Learning*. DOI: 10.48550/arXiv.2505.00466. arXiv: 2505.00466 [cs]



This publication has emanated from research conducted with the financial support of Taighde Éireann – Research Ireland grant number 20/FFP-P/8853 and the ADAPT Research Ireland Centre for AI Driven Digital Content Technology at Maynooth University under grant 13/RC/2106_P2.