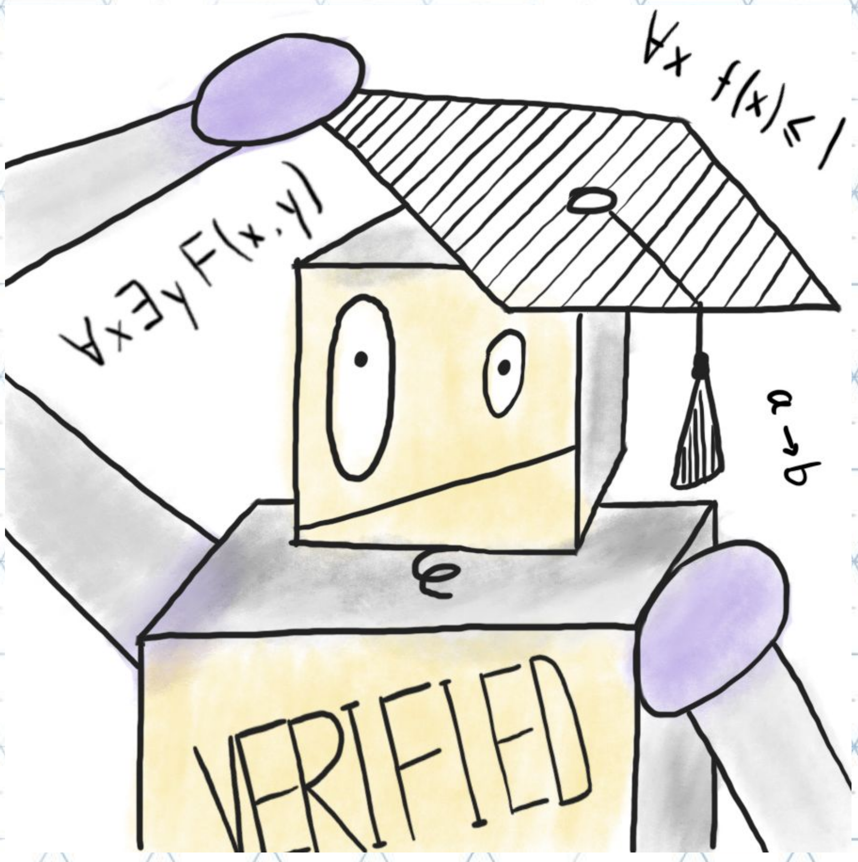
JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Formal Verification of Neural Network Controllers for Collision-Free Flight



Speakers: Daniel Genin

Coauthors: Ivan Papusha, Yanni Kouskoulas, Rosa Wu, Joshua Brulé, Galen Mullins, Tyler Young, Aurora Schmidt

Motivation

Challenges to verifying NN-based collision avoidance

- Very large state-action space
- Cumulative effect of multiple decisions
- NN opaqueness

Lots of work on ACAS Xu

- Guy Katz et al. – RELUPlex, Maribu
- Verified ~50 safety related properties

  e.g., if intruder is distant and much slower than ownship do not alert
- Local robustness properties
- ACAS Xu NNs and properties are now a benchmark used by VNN

Can *safety* of a NN-based collision avoidance system be verified in principle?

Simplified reinforcement learning model

Simplified 2D collision avoidance model/game

- Aircraft start a fixed distance apart
  - own on the left, intruder on the right
- Aircraft fly toward each other with equal constant horizontal speed
- On each time step each aircraft selects a vertical acceleration within fixed bounds
  - +/- 0.2 px/s^2 for own
  - +/- 0.15 px/s^2 for intruder
- Intruder maneuvers to intercept own aircraft
- Own aircraft must avoid intruder while minimizing altitude change

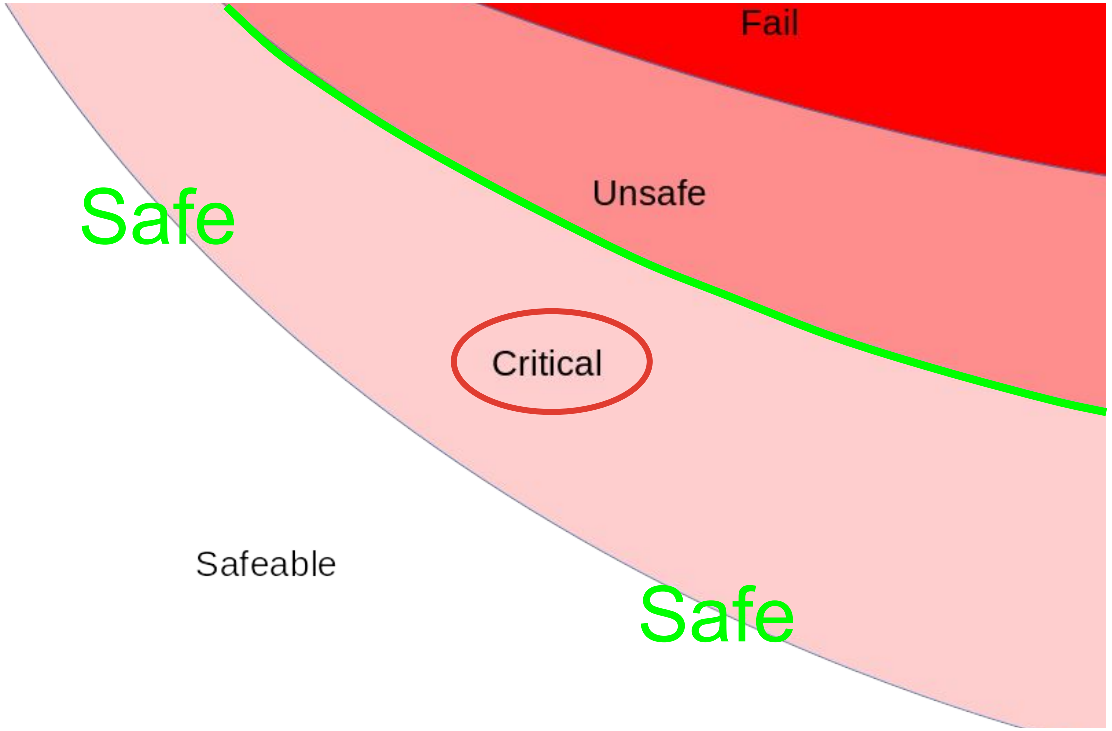
RL agent trained with use of only *affine activation units*: proximal policy optimization with a [16,16] hidden layer configuration and hard-tanh activation functions

$$\underbrace{[x] \xrightarrow{4} \boxed{\text{Linear}} \xrightarrow{16} \boxed{\text{Hardtanh}} \xrightarrow{16} \boxed{\text{Linear}} \xrightarrow{16} \boxed{\text{Hardtanh}} \xrightarrow{16} \boxed{\text{Linear}} \xrightarrow{1} [u]}_{u=\text{NN}(x;\theta)}$$

When *must* the system act to preserve safety?

State space stratification

- ***Safeable***: The intruder can be safely avoided (even in the worst case) on the next time step, no matter the currently selected action
- ***Critical***: The intruder can be safely avoided on the next time step (even in the worst case) *if* appropriate action is selected on current time step
- ***Unsafe***: The intruder cannot definitively be avoided on the next time step no matter the currently selected action
- ***Fail***: Collision is unavoidable or is already happening



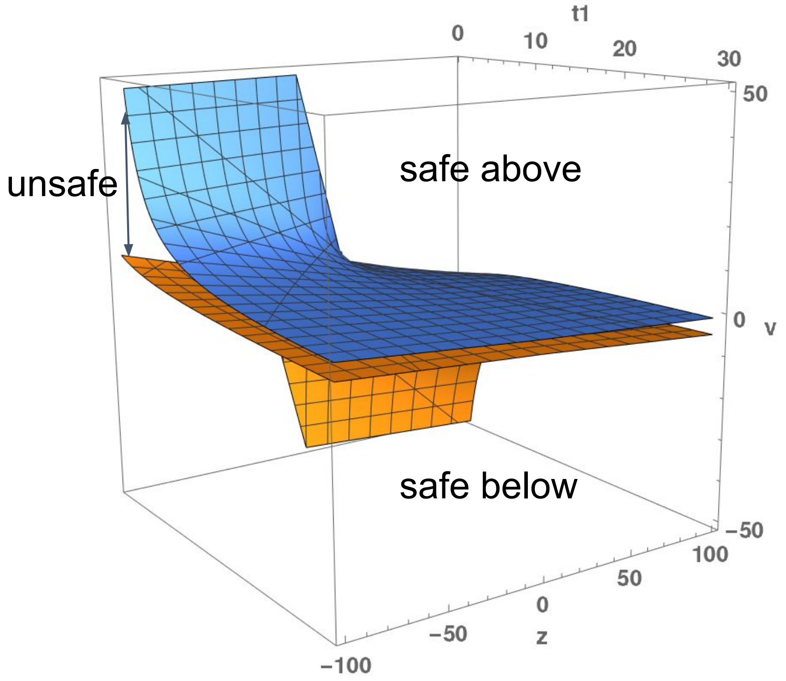
Yanni Kouskoulas, Aurora Schmidt, Jean-Baptiste Jeannin, Daniel Genin, Jessica Lopez, "Provably Safe Controller Synthesis Using Safety Proofs as Building Blocks," 7th International Conference in Software Engineering Research and Innovation (CONISOFT'19).

Piecewise constant approximation

The functions a_{above} and a_{below} are nonlinear and must be approximated to enable SMT solvers to decide safety efficiently.

For simplicity we used piecewise constant approximation on a square grid.

Monotonicity of a_{above} and a_{below} made the computation simple and efficient.



Level sets of a_{above} and a_{below} separating the critical and unsafe regions of the state space.

Verification

The NN controller will avoid all collisions provided the

safe predicate holds in every *critical state*

- Convert NN to Z3 expression using Lantern (https://github.com/JHUAPL/lantern-smt)
- Use piecewise constant over approximation to compute a_{above} and a_{below} acceleration bounds
- Generate safety predicate constraints for critical states
- Use Z3 to identify passing and failing states

Numerical experiment

1. **Train**: A simple (4x16x16x1) HardTanh actor-critic RL network

1. **Sub-divide:** 23.2M cubes over the 3D space of (relative) altitude, (relative) vertical velocity, and time to entry into horizontal conflict

1. **Over approximate the *critical set*:** only the 185k cubes that are critical need to be verified

2. **Verify**:
   - bad news: only 21.4k of 185k critical cubes are formally verified safe, and over 90% of the rest had a counterexample;
   - good news: a random sampling showed that only 9.9% of scenarios ever even visited a critical cube

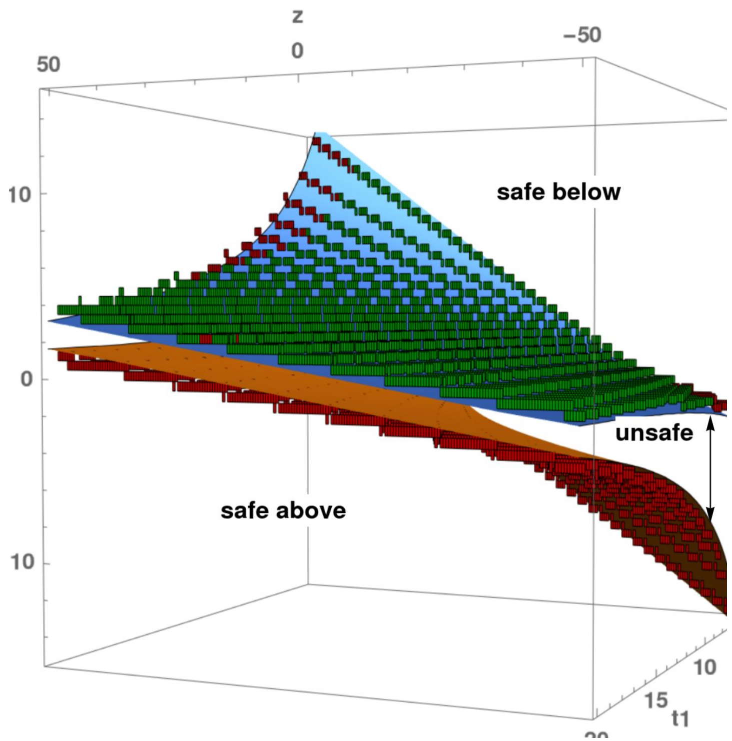


Table 1: Verification results for all hypercubes

Total Hypercubes	23,200,000
Hypercubes identified as critical	184,936 (100%)
Failing critical cubes	163,518 (88.4%)
Verified safeable critical cubes	21,418 (11.5%)

Table 2: Testing of the Network and Safety Guarantees

Total test scenarios	181,147
Number with collisions	12,678
Collision scenarios predicted by unsafe cube traversal	12,678
Collision scenarios not traversing unsafe cube	0
Total number visited critical cubes	1,722
Total visited safe cubes	635
Total visited failing cubes	1,087

JOHNS HOPKINS

APPLIED PHYSICS LABORATORY