Simulation-Based Adversarial Test Generation for Autonomous Vehicles with Machine Learning Components

**Background / Purpose**

- **Autonomous vehicles** generally use machine learning components (neural networks) and are very difficult to thoroughly test.
- **8.8 billion miles** of testing is required to demonstrate human-driver level safety.*


- **Falsification** is an optimization-guided automated testing technique for Cyber-Physical Systems.

**Goal:** Falsification of formal requirements for an autonomous vehicle in a closed-loop with the perception system by generating adversarial examples in the scenario configuration space.

**Problem Definition**

- **Given** • A base test scenario (Outline of the tested operation),
  - Discrete (e.g., car colors and models, pedestrian clothing colors) and continuous variables (e.g., positions and velocities of the vehicles and pedestrians) of the scenario,
  - Formal system requirements (Signal Temporal Logic / Metric Temporal Logic),

- **Find** • A discrete scenario setup,
  - A set of values for the continuous scenario variables

- **Such that** System under test starts failing to satisfy its requirements

**Approach**

- Create virtual framework for testing autonomous vehicles; includes controller, environment, and DNN-based perception system.
- Utilize **Covering Arrays** to create test cases that cover specific number of combinations of discrete and discretized parameters.
- Utilize **Falsification** to search for values of continuous variables that cause behaviors on the boundary between good and bad.

**Why Combinatorial Testing?**

Neural-Network based perception systems may be sensitive to certain combinations of shapes, colors and positions.

**Using Covering Arrays to Identify Critical Parameter Values**

- **Discrete Parameters**
  - Colors of Vehicles (5x5),
  - Models of Vehicles (6x6),
  - Pedestrian shirt color (5),
  - Pedestrian Pants color (5),
  - Fog: Exists/does not exist (2)

- **Continuous Parameters**
  - Position of Ego Vehicle
  - Position of Agent Vehicle 1
  - Pedestrian walking speed

**Falsification**

**Test Generation Approaches**

- **Global Uniform Random:** Over Discrete & Continuous Space
- **Covering Arrays + Uniform Random Falsification (CA+UR):**
  1. Use the best covering array results
  2. Do random search over the continuous Space

- **Covering Arrays + Simulated Annealing Falsification (CA+SA):**
  1. Use the best covering array results
  2. Apply Simulated Annealing optimization to search over the continuous space for minimum robustness value

CA + SA can utilize the discrete parameters of the cases with no collision and find collision cases by applying optimization on continuous space

**Experiment Results**

- **Safety Requirement:** The Ego vehicle shall not collide into another vehicle or a pedestrian.

Average minimum robustness values found by different approaches. (Smaller is better)

We search for “zero” robustness: boundary between bad/good behavior

- **Collision**
  - Results for naive (non-production) vehicle control system

- **S-TaLiRo** is used as the Falsification Framework