Coverage-Guided Fuzz Testing for Cyber-Physical Systems

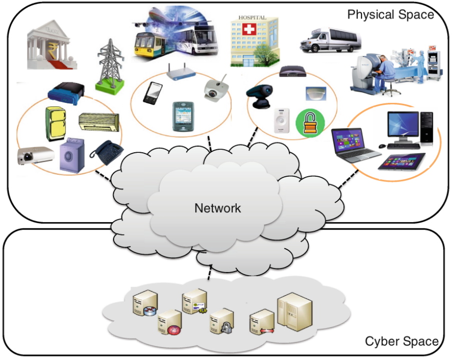
Sanaz Sheikhi Edward Kim Parasara S. Duggirala Stanley Bak

ICCPS 2022

Stony Brook University

THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Motivation



CPS properties:

- Hardware and software space
- Complex protocols

Challenges:

- Does the CPS works correctly?
- How to generate test cases?

Fuzz Testing

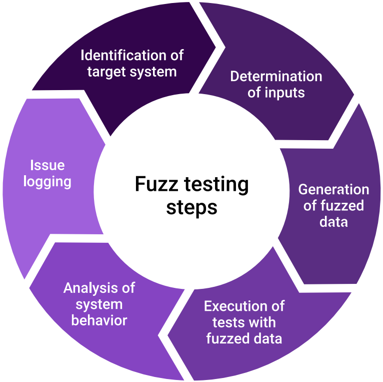


Image from `www.synopsys.com`

An automated software testing method injecting invalid, malformed, or unexpected inputs into a system to reveal bugs and vulnerabilities.

Fuzz Testing CPSs

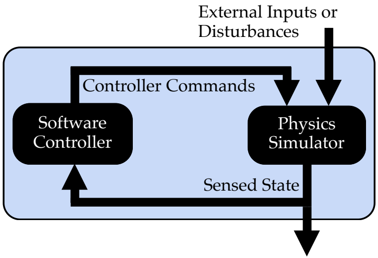
Challenges of CPS fuzz testing:

- Continuous states,
- Inputs that change over time

CPSFuzz

- **Novel coverage notion** to evaluate fuzz testing methodology effectiveness for CPS.
- **Customized power schedule**: leverages coverage score to select promising inputs to find failures in new system states.
- **Customized mutation strategy**: reasons with the causal nature of a CPS.

CPS Execution Model



- Black-box simulator model:

$$f : X \times U \times W \to Y$$

- Black-box software controller:

$$g : Y \to U$$

- **Goal**: find external input sequences, $w_0, w_1, \ldots w_T$, that cause errors

CPS Coverage Metric

Designe properties:

- Adding more events never decreases the metric
- Identical events do not increase the metric
- Similar events have a lower impact than dis-similar events

\- Input: sensed states at events
\- Output: scalar coverage score

$$\mathcal{S} : \mathsf{Set}[Y] \to \mathbb{R}$$

CPS Coverage Metric

- **Objective Space Projection Function:** maps sensed state to a o-dimensional Euclidean space:

$$\mathcal{P} : Y \rightarrow \mathbb{R}^o$$

- **Objective Space Exploration Limits:** box bounds within the objective space:

$$\mathcal{B} \in \mathbb{R}^{2o}$$

- **Kernel function:** measures the similarity of states in the objective space using o-dimensional normal distribution

$$\mathcal{N}(\mu, \sigma^2)$$

  - μ: a point in the objective space of each event
  - σ: a fixed hyper-parameter

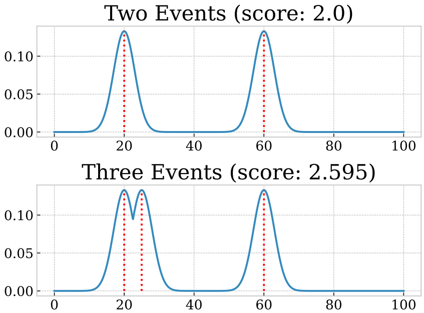
CPS Coverage Metric

Metric computation:

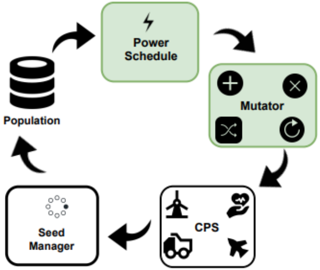
- Map each event to the objective space,
- Apply kernel functions to measure states similarity,
- Integrate the maximum of the kernels

$$\mathcal{S}(\mathsf{Set}[Y]) = \int_B \max_{y \in \mathsf{Set}[Y]} \mathcal{N}(\mathcal{P}(y), \sigma^2)(b) \; \mathrm{d}b$$

CPS Coverage Metric (Example)



CPSFuzz Architecture



CPSFuzz overview

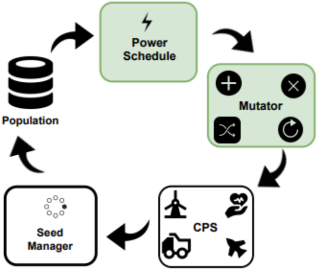
- **CPS**: execution or simulation.

- **Seed**: initial inputs for mutation.

- **Population**: set of all inputs, and test results.

-**Seed Manager**: maintains the population.

CPSFuzz Architecture



CPSFuzz overview

- **Power Schedule**: selects a seed based on seeds' energy.
- **Energy**: probability that a seed will be picked.
- **Mutator**: performs various operations on a valid seed.

CPSFuzz Power Schedule

-**Problem of generic power schedule :**

- Waste testing cycles on duplicate seeds.
- Deprives promising seeds.

- **CPSFuzz solution**:

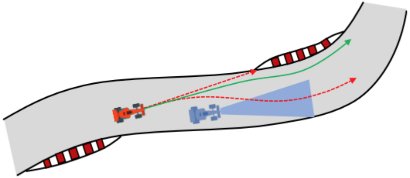
- Finds a subset of the objective state space with *minimum* CPS coverage score.
- Picks a seed that improves the coverage of the subset.

CPSFuzz Mutator

- **Problem of generic mutation:**
  - Blind input modification
  - Fine-grained operations

- **CPSFuzz's Mutation:**
  - Maps subset of state space to an interval in input sequence.
  - Employs coarse-grained mutations at control command level.

Evaluation



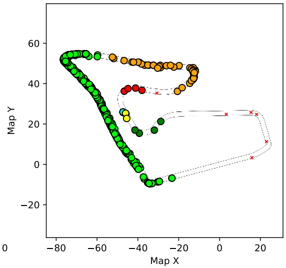
Case study:

- F1TENTH autonomous racing competition
- Stress test overtake maneuvers
- Perturb the adversarial agent behavior
- Interesting events: collisions

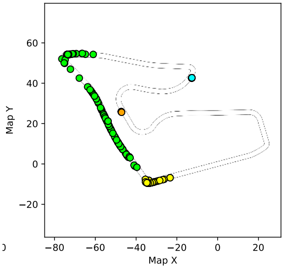
Comparison:

- Hypothesis
- Atheris
- Random approach

Evaluation



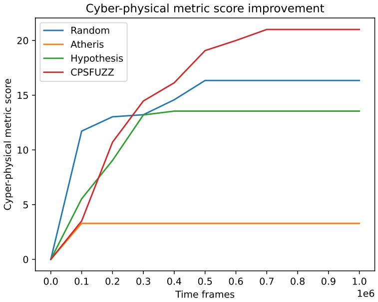
(a) CPSFuzz



(b) Random approach

DBScan: measuring uniqueness of failures by spatial clustering.

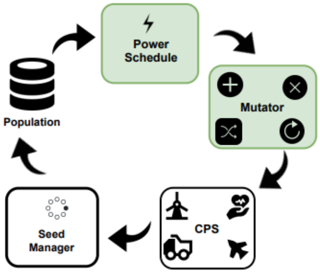
Evaluation



Fuzzer	# Test cases	Score
CPSFuzz	361	21.06
Atheris	635	3.28
Hypothesis	562	13.54
Random	499	16.34

Median scores during five runs of test case generation, one million frames at each run

Conclusion



CPSFuzz overview

- CPSFuzz: a framework for fuzz testing CPSs
- Notion of objective state space coverage
- `https://github.com/sanazsheikhi/CPSFuzz/tree/master`