On Minimising the Maximum Expected Verification Time

Toni Mancini Federico Mari **Annalisa Massini** Igor Melatti
Ivano Salvo Enrico Tronci

SAPIENZA
UNIVERSITÀ DI ROMA

Computer Science Department – Sapienza University of Rome
Via Salaria 113, 00198 Roma - Italy
http://mclab.di.uniroma1.it/

IWES 2017
Rome – September 7–8, 2017

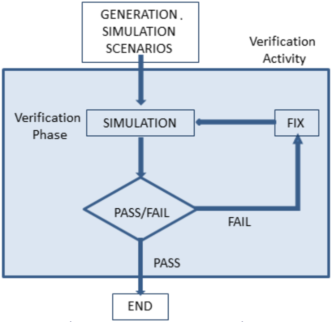
System Level Formal Verification

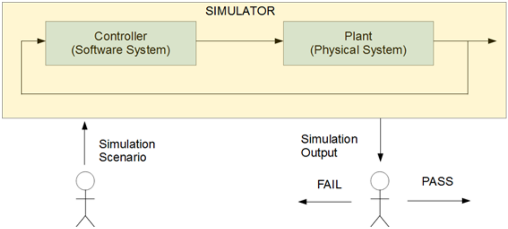
System Level Formal Verification (SLVF): verify that the *whole* (i.e., software + hardware) system meets the given specifications
Current workhorse: **Hardware In the Loop Simulation** (HILS)

SLFV may be effectively carried out by an exhaustive HILS:

- *All* relevant finite *simulation scenarios* are generated (*generation phase*)
- All simulation scenarios are simulated (*verification phase*)

Single verification phases are repeatedly performed, until the output is PASS



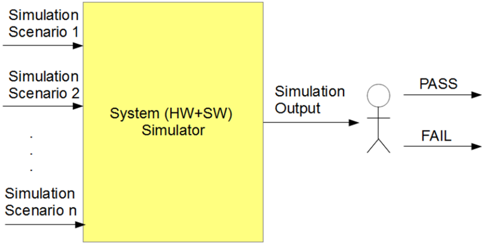


Motivations and Objectives

Main concern in a HILS campaign:

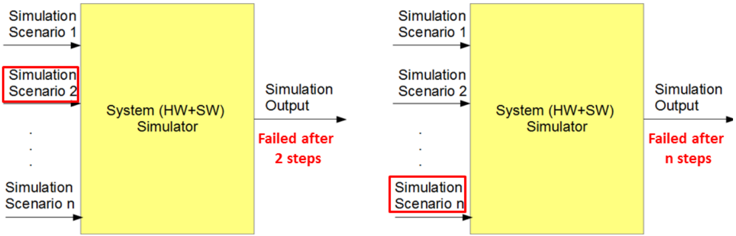
- **Time** needed by the whole verification activity may be **huge**

GOAL: minimise the time taken by the verification activity



Idea

- **Define the simulation scenarios** by using the *disturbances* (faults, delays, etc.) to be injected into the *System Under Verification* (SUV)
- **Reorder simulation scenarios** so that in each verification phase **the scenario witnessing the error occurs as soon as possible**



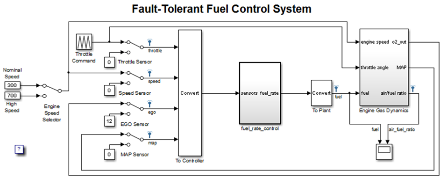
Simulation of all scenarios is very time consuming

Problem: **how to order scenarios** to avoid the simulation of them all

Example of Simulation scenario

Fuel Control System
model in the Simulink distribution

- Four sensors: throttle angle, speed, *Oxygen in Exhaust Gas* (EGO) and *Manifold Absolute Pressure* (MAP)
- **Disturbances** (*uncontrollable inputs* such as faults, delays, etc):
  - $d_1 \rightarrow$ fault on EGO (repaired in 1s)
  - $d_2 \rightarrow$ fault on MAP (repaired in 1s)
  - $d_3 \rightarrow$ no fault event



Set of disturbances $\mathcal{D}$ is $\{d_1, d_2, d_3\}$

Examples of simulation scenarios (*finite sequence of disturbances*):

- $\delta_1 = \langle d_1, d_3, d_2, d_3 \rangle$ (of length 4)
- $\delta_2 = \langle d_2, d_3, d_2 \rangle$ (of length 3)

Example of ASE and Simulation campaign

Each verification phase is performed as a simulation campaign:

- **Simulation campaign** – permutation of elements of a finite set $\mathcal{A}$ of simulation scenarios such that no scenario is a prefix of another one (*Admissible System Environment* – ASE)

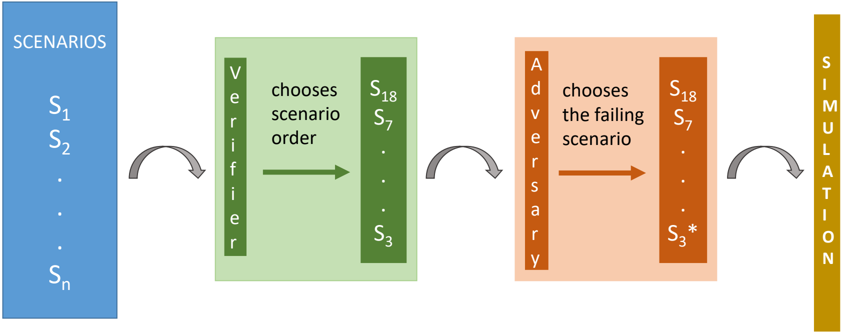
In the Fuel Control System model we can consider:

- The set of disturbances $\mathcal{D} = \{d_1, d_2, d_3\}$
- Assuming that at most *one fault* can occur in the *first position* of simulation scenarios of *length 3* $\rightarrow$ the simulation scenarios set is $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$ where:
  - $\delta_1 = \langle d_1, d_3, d_3 \rangle$ $\quad \delta_2 = \langle d_2, d_3, d_3 \rangle$ $\quad \delta_3 = \langle d_3, d_3, d_3 \rangle$
- The set of **simulation campaigns** $\mathrm{Sim}(\mathcal{A})$ consists of $3! = 6$ elements
- $\mathrm{Sim}(\mathcal{A}) = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}$, where:
  - $\sigma_1 = \langle \delta_1, \delta_2, \delta_3 \rangle$ $\quad \sigma_2 = \langle \delta_1, \delta_3, \delta_2 \rangle$ $\quad \sigma_3 = \langle \delta_2, \delta_1, \delta_3 \rangle$
  - $\sigma_4 = \langle \delta_2, \delta_3, \delta_1 \rangle$ $\quad \sigma_5 = \langle \delta_3, \delta_1, \delta_2 \rangle$ $\quad \sigma_6 = \langle \delta_3, \delta_2, \delta_1 \rangle$

Two-person Zero-sum Game

We model the verification phase as a **two-person zero-sum game**

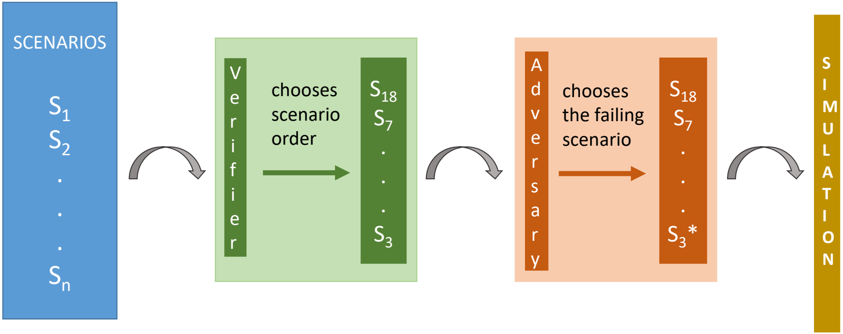
- **Player 1** (the *verifier*) chooses the (possibly probabilistic) ordering strategy in which scenarios will be simulated
- **Player 2** (the *adversary*) chooses which scenarios witness an error (failing scenario) in a predefined scenarios ordering (e.g., the lexicographic one)
- The goal for the **verifier** is to **minimise** the verification time



Two-person Zero-sum Game

Thus

- The **payoff** for our game is the **verification time**
- **Adversary objective** → place the failing scenario so that such scenario is the last (after the verifier has reordered all scenarios)
- **Verifier objective** → reorder the scenarios so that the failing one is the first



Error Injection Strategy

The **error injection** is the (probabilistic) strategy of the **adversary** player:

- An *error injection strategy* x, for an ASE $\mathcal{A}$, is a function x: $\mathcal{A} \to [0, 1]$ such that $\sum_{\alpha \in \mathcal{A}} x(\alpha) = 1$

Example of Error Injection Strategy

Consider the ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$

- $x_1(\delta_1) = \frac{1}{3}$ $x_1(\delta_2) = \frac{1}{3}$ $x_1(\delta_3) = \frac{1}{3}$
- $x_2(\delta_1) = 0$ $x_2(\delta_2) = 1$ $x_2(\delta_3) = 0$

Note that:

- Strategy x_2 consists in deterministically choosing δ_2 as the failing scenario
- x_2 is a pure strategy, whilst x_1 is not

Simulation Strategy

The **simulation strategy** is the (probabilistic) strategy of the **verifier** player:

- A *simulation strategy* y, for an ASE $\mathcal{A}$, is a function $y\colon \mathrm{Sim}(\mathcal{A}) \to [0,1]$ such that $\sum_{\sigma \in \mathrm{Sim}(\mathcal{A})} y(\sigma) = 1$

Example of Simulation Strategy

Consider the ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$ and $\mathrm{Sim}(\mathcal{A}) = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6\}$

- $y_1(\sigma_i) = \frac{1}{6}, i = 1, \ldots, 6$
- $y_2(\sigma_2) = \frac{1}{2}, y_2(\sigma_4) = \frac{1}{2}, y_2(\sigma_i) = 0, i = 1, 3, 5, 6$

Expected Verification Time

The **Expected Verification Time** (EVT) for the verification activity is the expected number of simulation scenarios to be simulated before hitting the one that witnesses the error

$$\mathrm{EVT}(x, y) = \sum_{\delta \in \mathcal{A}} \sum_{\sigma \in \mathrm{Sim}(\mathcal{A})} x(\delta)\chi(\sigma, \delta)y(\sigma)$$

where:

- $\chi(\sigma, \delta)$ is the position of simulation scenario δ in the simulation campaign σ
- x is the adversary error injection strategy
- y is the simulation strategy

The **Worst Case Expected Verification Time** (WCEVT) is the maximum EVT after any adversary choice

$$\mathrm{WCEVT}(y) = \max_{x \in X} \mathrm{EVT}(x, y)$$

Example of Expected Verification Time

Consider:

- The ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$
- The error injection strategy x_1: $x_1(\delta_1) = \frac{1}{3}, x_1(\delta_2) = \frac{1}{3}, x_1(\delta_3) = \frac{1}{3}$
- The simulation strategy y_2: $y_2(\sigma_2) = \frac{1}{2}, y_2(\sigma_4) = \frac{1}{2}, y_2(\sigma_i) = 0, i = 1, 3, 5, 6,$ where $\sigma_2 = \langle \delta_1, \delta_3, \delta_2 \rangle$ and $\sigma_4 = \langle \delta_2, \delta_3, \delta_1 \rangle$

The **Expected Verification Time** is:

$$\mathrm{EVT}(x_1, y_2) = \sum_{i=1}^{3} x_1(\delta_i)\chi(\sigma_2, \delta_i) y_2(\sigma_2) + \sum_{i=1}^{3} x_1(\delta_i)\chi(\sigma_4, \delta_i) y_2(\sigma_4) =$$

$$= \sum_{i=1}^{3} \frac{1}{3}\chi(\sigma_2, \delta_i)\frac{1}{2} + \sum_{i=1}^{3} \frac{1}{3}\chi(\sigma_4, \delta_i)\frac{1}{2} = \frac{1}{6}\sum_{i=1}^{3} i + \frac{1}{6}\sum_{i=1}^{3} i = 2$$

Theorem on MiniMax Expected Verification Time

Our main result (inspired by the Minimax Theorem of Von Neumann) provides:

- a **lower bound for the verifier payoff**, that is the minimum value for the Worst Case Expected Verification Time $\mathrm{MiniMaxEVT}$
- the conditions for a **simulation strategy** to be **optimal** (attaining the **optimal payoff**)

Theorem

Let $\mathcal{A} = \{\delta_1, \ldots, \delta_n\}$ be an ASE. Then the following statements hold:

- *The value for the minimum WCEVT is* $\mathrm{MiniMaxEVT} = \frac{n+1}{2}$
- *A simulation strategy $y \in Y$ is optimal* **iff** *it satisfies the following constraints:*

$$\sum_{t=1}^{n} t \sum_{\chi(\sigma,\delta_i)=t} y(\sigma) = \frac{n+1}{2} \text{ for } i \in [1, n]$$

- *A simulation strategy attaining the optimal payoff* $\mathrm{MiniMaxEVT}$ *is the uniform simulation strategy* $\hat{y}(\sigma) = \frac{1}{n!}$

Optimal Simulation Strategies

The simulation strategy attaining the minimum WCEVT is **not unique**

There is an **infinite number of optimal simulation strategies**, that is any solution to the (feasibility) LP problem:

$$\begin{cases} \sum_{t=1}^{n} t \sum_{\chi(\sigma,\delta_i)=t} y(\sigma) = \frac{n+1}{2} \text{ for } i \in [1, n] \\ \sum_{\sigma \in \mathrm{Sim}(\mathcal{A})} y(\sigma) = 1 \\ 0 \leq y(\sigma) \leq 1 \text{ for } \sigma \in \mathrm{Sim}(\mathcal{A}). \end{cases}$$

The **set of solutions** is a **closed bounded convex polytope**

Example

- Consider the ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$ and $\mathrm{Sim}(\mathcal{A})$
- An optimal strategy has payoff $\mathrm{MiniMaxEVT} = \frac{n+1}{2} = \frac{3+1}{2} = 2$
- Consider the two simulation strategies:
  - y_1: $y_1(\sigma_i) = \frac{1}{6}, i = 1, \ldots, 6$
  - y_2: $y_2(\sigma_2) = \frac{1}{2}, y_2(\sigma_4) = \frac{1}{2}, y_2(\sigma_i) = 0, i = 1, 3, 5, 6$

Both strategies are optimal

Example (continued)

For simulation strategy y_1:

- $y_1(\sigma) = \frac{1}{n!} = \frac{1}{6}$ for all $\sigma \in \mathrm{Sim}(\mathcal{A})$
- $\mathrm{WCEVT}(y_1) = 2$

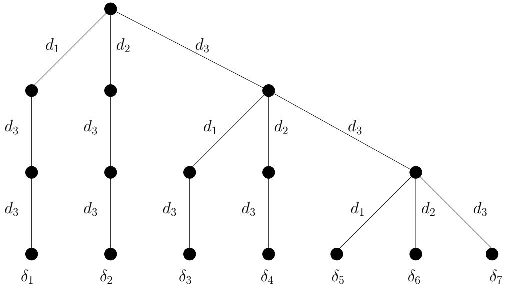
For simulation strategy y_2:

- y_2 consists in choosing at random $\sigma_2 = \langle \delta_1, \delta_3, \delta_2 \rangle$ or $\sigma_4 = \langle \delta_2, \delta_3, \delta_1 \rangle$
- Then $\mathrm{EVT}(x, y_2) = \frac{1}{2} \sum_{i=1}^{3} x(\delta_i)\chi(\sigma_2, \delta_i) + \frac{1}{2} \sum_{i=1}^{3} x(\delta_i)\chi(\sigma_4, \delta_i)$
  - $\mathrm{EVT}(x_1^*, y_2) = \frac{1}{2}[x_1^*(\delta_1)\chi(\sigma_2, \delta_1) + x_1^*(\delta_1)\chi(\sigma_4, \delta_1)] = \frac{1}{2}[1+3] = 2$
  - $\mathrm{EVT}(x_2^*, y_2) = \frac{1}{2}[x_2^*(\delta_2)\chi(\sigma_2, \delta_2) + x_2^*(\delta_2)\chi(\sigma_4, \delta_2)] = \frac{1}{2}[2+2] = 2$
  - $\mathrm{EVT}(x_3^*, y_2) = \frac{1}{2}[x_3^*(\delta_3)\chi(\sigma_2, \delta_3) + x_3^*(\delta_3)\chi(\sigma_4, \delta_3)] = \frac{1}{2}[3+1] = 2$

- This implies $\mathrm{WCEVT}(y_2) = \max_{x^* \in X^*} \mathrm{EVT}(x^*, y_2) = 2 = \mathrm{WCEVT}(y_1)$

Monte Carlo-like Simulation

A prefix tree can be used to represent a simulation strategy



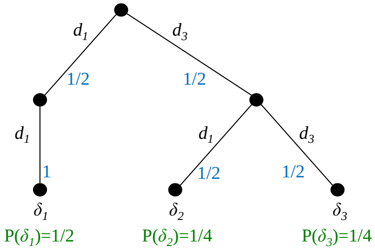
On-line scenario generation often rests on **Monte Carlo**-like approaches:

- At each simulation step a disturbance injection is chosen at random (for each simulation run)
- A Monte Carlo simulation strategy corresponds to a random walk on the **disturbance tree** associated to ASE $\mathcal{A}$

A Monte Carlo simulation strategy may not be optimal

Example of Non-Optimal Monte Carlo Simulation Strategy

- Consider the ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$, where $\delta_1 = \langle d_1, d_1 \rangle$, $\delta_2 = \langle d_3, d_1 \rangle$, $\delta_3 = \langle d_3, d_3 \rangle$
- An optimal strategy y should yield a payoff $\mathrm{MiniMaxEVT} = \frac{n+1}{2} = 2$

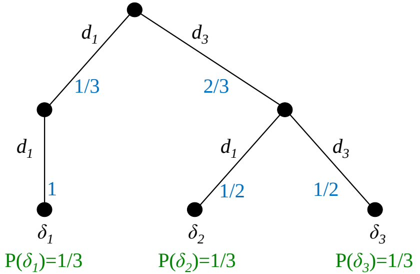


- For the considered y (computing the conditional probability) we have:
$$y(\sigma_1) = \frac{1}{4},\ y(\sigma_2) = \frac{1}{4},\ y(\sigma_3) = \frac{1}{6},\ y(\sigma_4) = \frac{1}{12},\ y(\sigma_5) = \frac{1}{6}, \text{ and } y(\sigma_6) = \frac{1}{12}$$
- Considering only pure error injection strategies x_1^*, x_2^*, x_3^*, we obtain:
$$\mathrm{EVT}(x_1^*, y) = \frac{5}{3},\ \mathrm{EVT}(x_2^*, y) = \frac{13}{6}, \text{ and } \mathrm{EVT}(x_3^*, y) = \frac{13}{6}$$
- Thus $\mathrm{WCEVT}(y) = \max\left\{\frac{5}{3}, \frac{13}{6}, \frac{13}{6}\right\} = \frac{13}{6} > 2$

A Monte Carlo simulation strategy may not be optimal

Example of Optimal Monte Carlo Simulation Strategy

- **Optimality sufficient condition**: If for all $\delta \in \mathcal{A}$, $P(\delta) = \frac{1}{|\mathcal{A}|}$, then the Monte Carlo simulation strategy y for $(\mathcal{A}, p)$ is optimal
- Consider the ASE $\mathcal{A} = \{\delta_1, \delta_2, \delta_3\}$, where $\delta_1 = \langle d_1, d_1 \rangle$, $\delta_2 = \langle d_3, d_1 \rangle$, $\delta_3 = \langle d_3, d_3 \rangle$



$P(\delta_i) = \frac{1}{3}$ for all $\delta_i \in \mathcal{A}$, thus the Monte Carlo simulation strategy $y_{\mathcal{A}}$ is **optimal**

Conclusions and Future Work

We addressed the problem of identifying an ordering on the scenarios (sequences of disturbances) to be simulated so as to minimise the WCEVT

Our results can be summarised as follows:

- The minimum WCEVT is $\frac{n+1}{2}$, where n is the number of scenarios to simulate
- There is an infinite set of optimal simulation strategies (strategies for which the minimum WCEVT is attained), forming a bounded convex polytope
- Ordering simulation scenarios uniformly at random yields an optimal simulation strategy
- We show how to select probability distribution on disturbances to have an optimal simulation strategy for on–line Monte Carlo-based simulation settings

Future Work

- Search effective methods to generate on-line optimal simulation campaigns

Thanks