An Active Learning Approach to the Falsification of Black Box Cyber-Physical Systems

Simone Silvetti, Alberto Policriti, Luca Bortolussi

silvetti.simone@spes.uniud.it

silvetti@esteco.com

13th International Conference on integrated Formal Methods

Outline

Overview

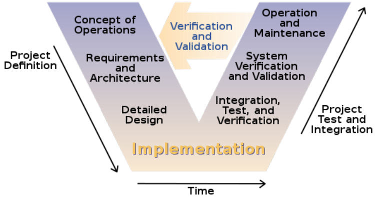
Model Based Development

Methodology based on a computational model of a real target system

- used at the early stage of the design phase
- used at the end to verify the compliance of the real system

Motivations

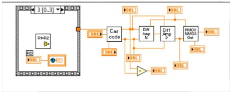
- reducing the time of prototyping
- reducing the cost of development



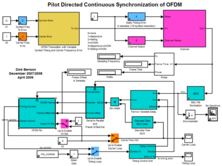
Models

Software: Block Diagram Systems

LabView



Simulink



Computational Models

- Hybrid Systems
- CPS
- Automata
- Statistical Models

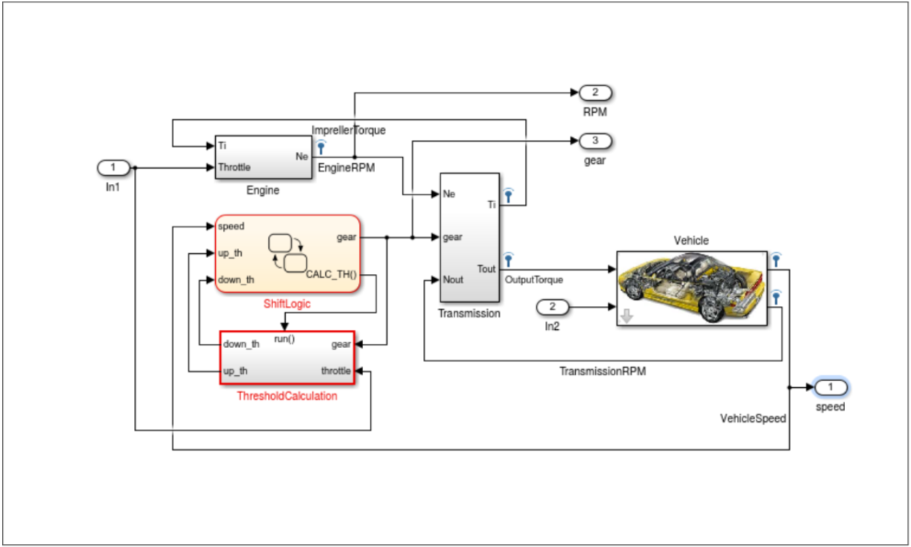
Problem

Too Much Complexity $\Rightarrow$ no standard Model checking techniques.

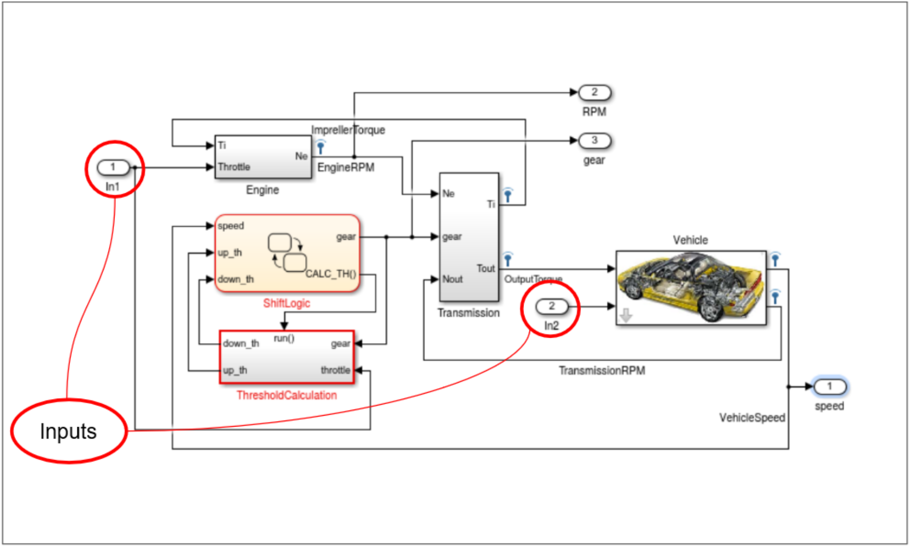
Solution

Black Box Assumption and **Search-based approach**.

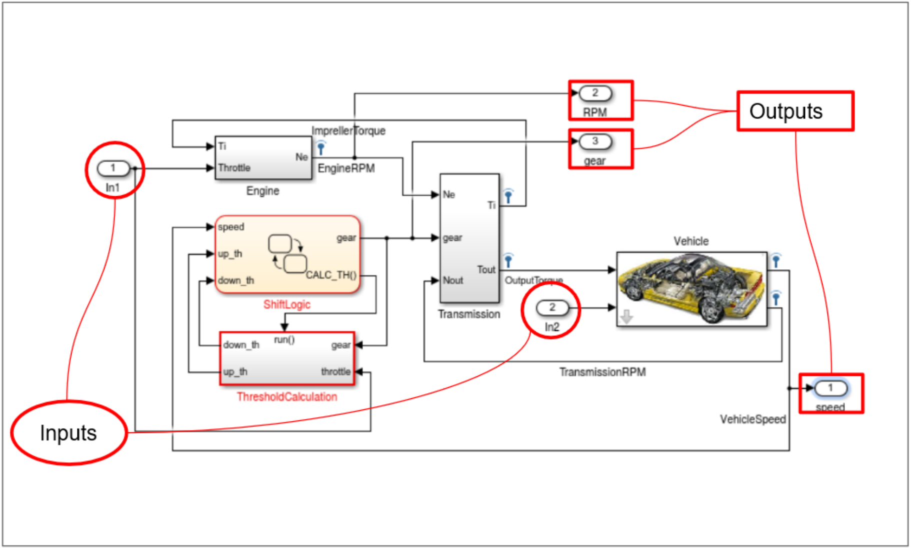
Simulink Model



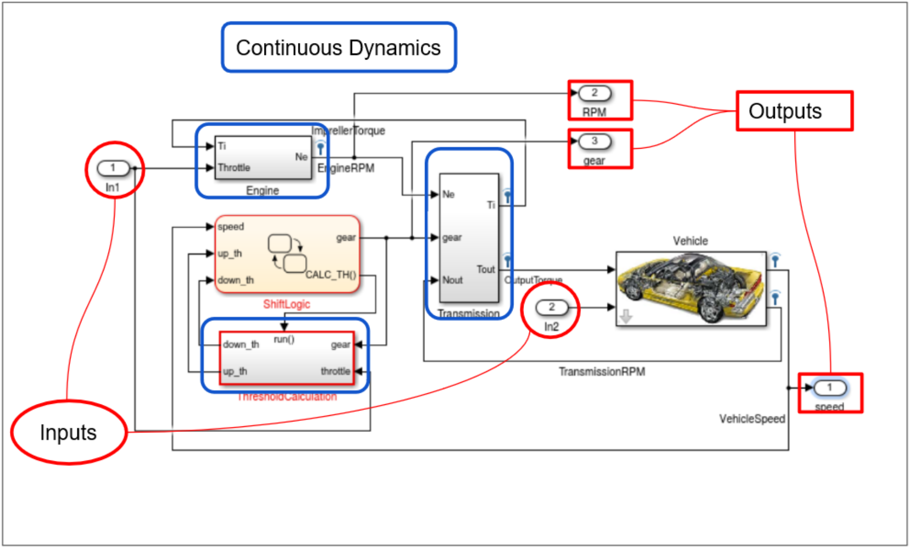
Simulink Model - Inputs



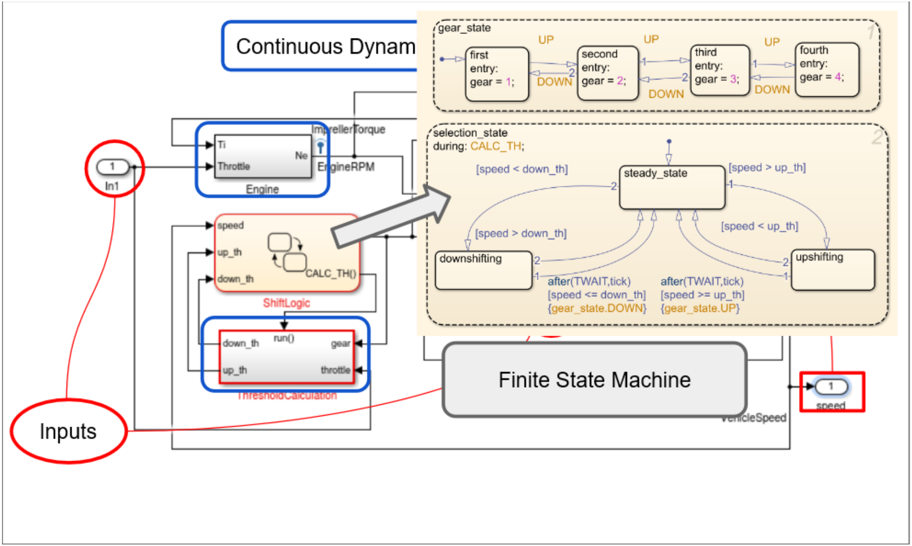
Simulink Model - Outputs



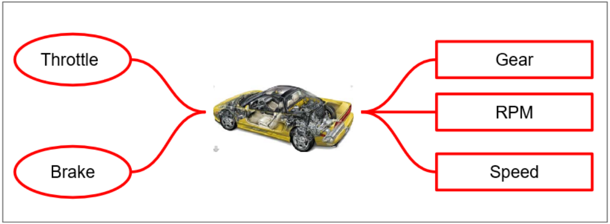
Simulink Model - Continuous Dynamics



Simulink Model - Finite State Machine



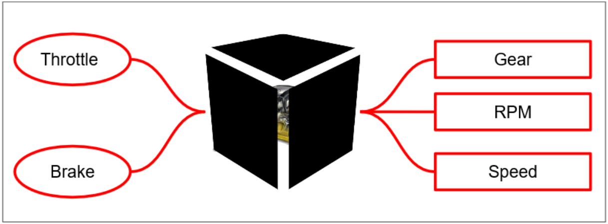
Black Box Assumption



Inputs & Outputs

The Inputs are Piece Wise Constant (PWC) Functions, the Outputs are PWC functions (Gear) or Continuous Functions.

Black Box Assumption



Black Box Assumption

- less information
- an more general approach (interesting by an industrial point of view)

The requirements: Signal Temporal Logic (STL)

Signal temporal logic is:

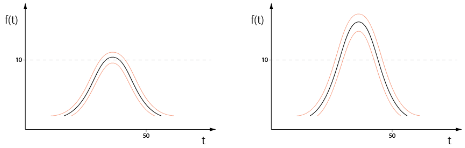
- a linear continuous time temporal logic.
- the atomic predicates are of the form $\mu(\vec{X}) := [g(\vec{X}) \geq 0]$ where $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuous function.
- the syntax is

$$\phi := \bot \mid \top \mid \mu \mid \neg\phi \mid \phi \vee \phi \mid \phi \mathbf{U}_{[T_1,T_2]}\phi, \quad (1)$$

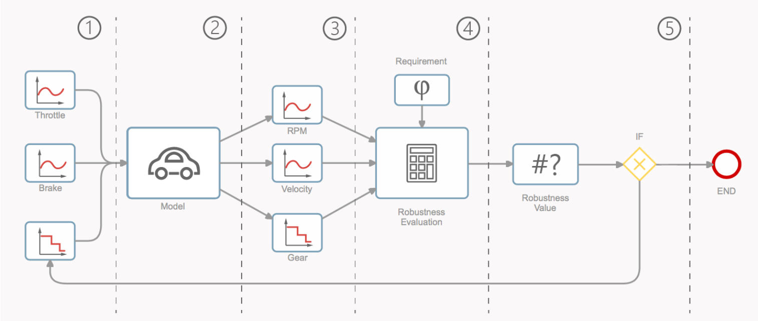
Example

$$\phi_1 := F_{[0,50]} |X_1 - X_2| > 10$$

1. **The Booleans semantics:** if a given path satisfies or not a given STL formula.
2. **The Quantitative semantics:** *How much* a given path satisfies or not a given STL formula.



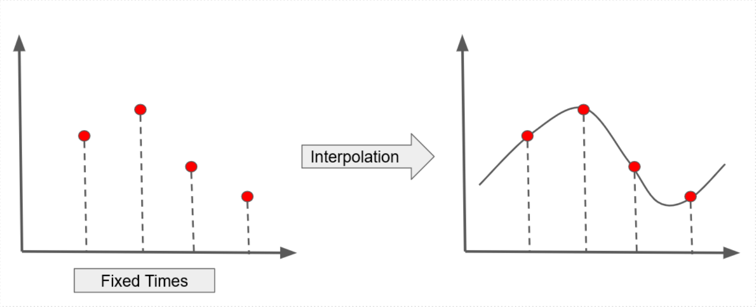
Search-Based Testing

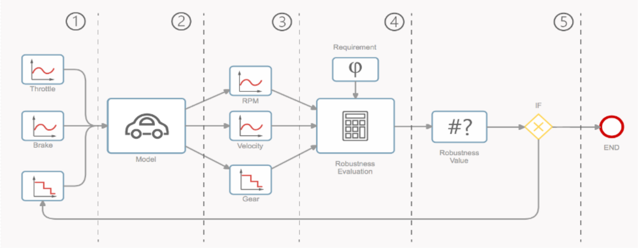


Falsification

- Goal: Find the input functions (1) which violate the requirements (4)
- Problems
  1. Falsify with a low number of simulations ⇒ Active Learning
  2. Functional Input Space(!!) ⇒ Adaptive Space Parameterization

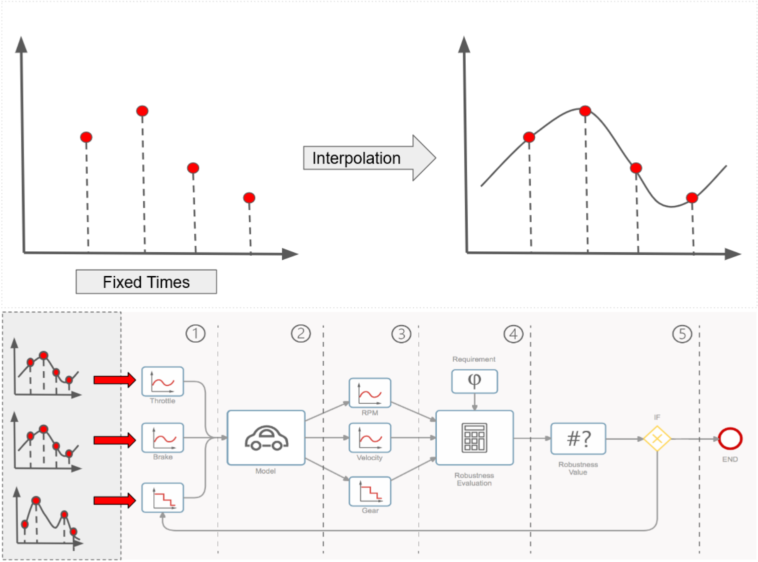
Fixed Parameterization





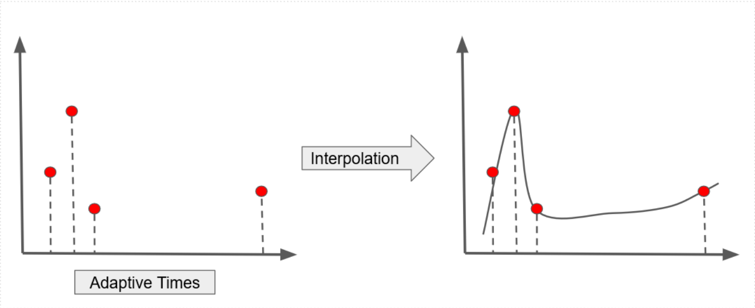
n adaptive control points ⇒ n variable to optimize

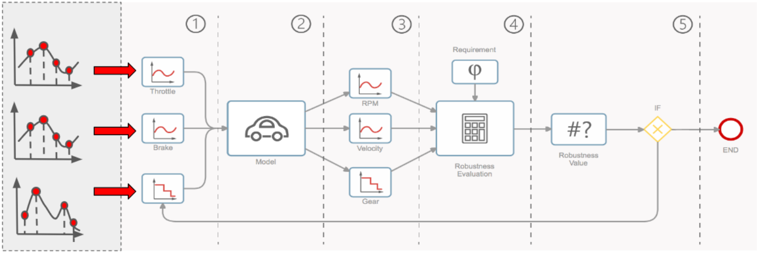
Fixed Parameterization



n fixed control points ⇒ n variable to optimize

Adaptive Parameterization





n adaptive control points ⇒ 2n variable to optimize

Domain Estimation Problem

Domain Estimation Problem

Consider a function $\rho : \Theta \to \mathbb{R}$ and an interval $I \subseteq \mathbb{R}$. We define the *domain estimation problem* as the task of identifying the set:

$$\mathcal{B} = \{\theta \in \Theta | f(\theta) \in I\} \subseteq \Theta \quad (2)$$

In practice, if $\mathcal{B} \neq \emptyset$, we will limit us to identify a subset $B \subseteq \mathcal{B}$ of size n.

Falsification $\sim$ Domain estimation problems

$$\mathcal{B} = \{\theta \in \Theta | \rho(\theta) \in (-\infty, 0)\} \subseteq \Theta$$

⇩

Gaussian Processes

Gaussian Processes

Definition

A random variable $f(\theta), \theta \in \Theta$ is a GP

$$f \sim \mathcal{GP}(m, k) \iff (f(\theta_1), f(\theta_2), \ldots, f(\theta_n)) \sim \mathcal{N}(\mathbf{m}, K)$$

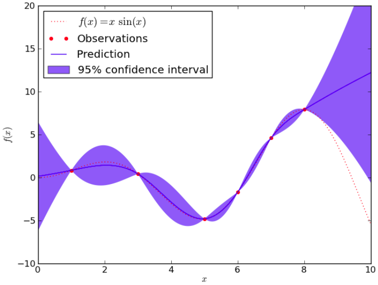
where $\mathbf{m} = (m(\theta_1; h_1), m(\theta_2; h_1), \ldots, m(\theta_n; h_1))$ and $K_{ij} = k(f(\theta_i), f(\theta_j); h_2)$

Prediction

$$\{f(\theta_1), \ldots, f(\theta_n), f(\theta')\} \sim \mathcal{N}(\mathbf{m}', K')$$

$$\mathbb{E}(f(\theta')) = (k(\theta', \theta_1), \ldots, k(\theta', \theta_N))K_N^{-1} r$$

$$var(f(\theta')) = k(\theta', \theta') - K(\theta, r)K_N^{-1}K(\theta, r)^T$$



Domain Estimation Problem

Domain Estimation Problem

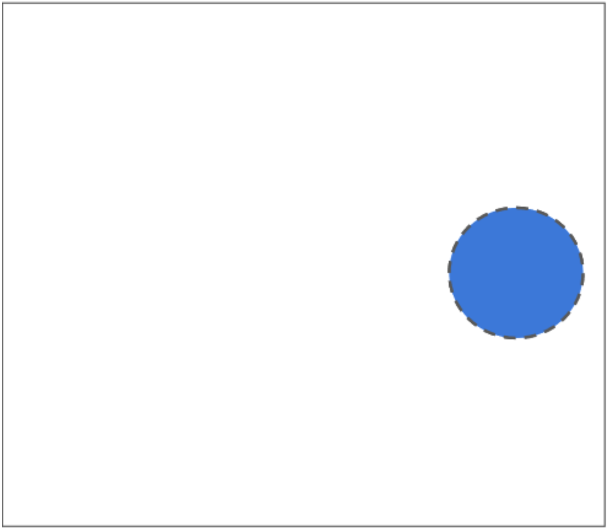
- Train Set: $K(\rho) = \{(\theta_i, \rho(\theta_i))\}_{i \leq n}$ (the partial knowledge)
- Gaussian Process: $\rho_K(\theta) \sim GP(m_K(\theta), \sigma_K(\theta))$ (the partial model)

$$P(\rho_K(\theta) < 0) = CDF\left(\frac{0 - m_K(\theta)}{\sigma_K(\theta)}\right)$$

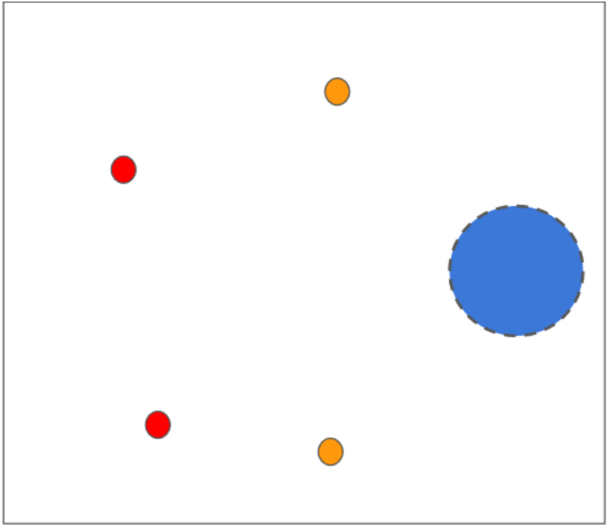
Simple Idea

Iteratively explore the area which is more probable to falsify the system by sampling from $P(\rho_K(\theta) < 0)$.

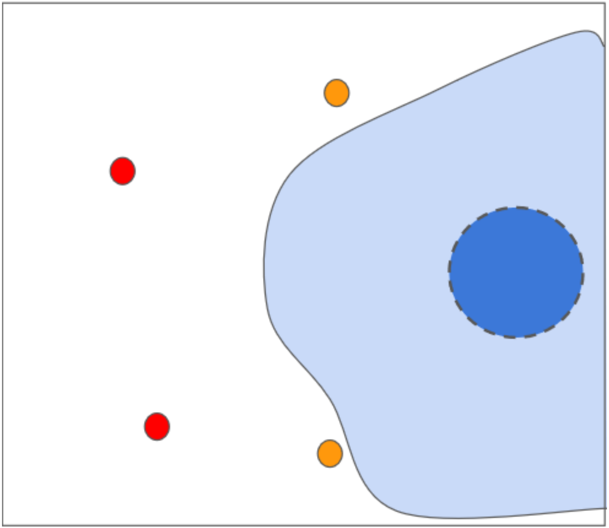
Algorithm - I



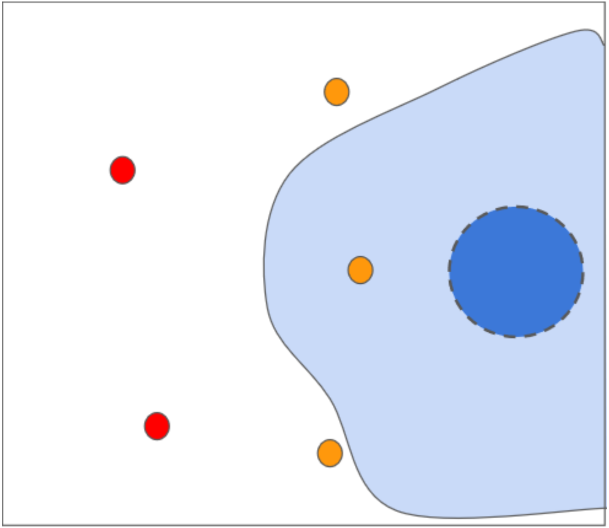
Algorithm - II



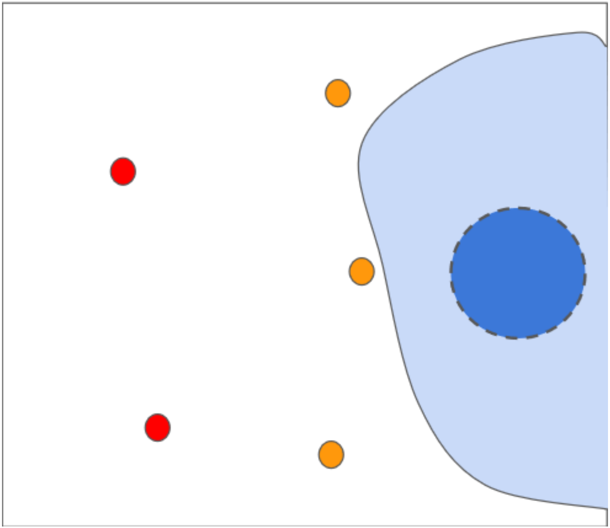
Aglorithm - III



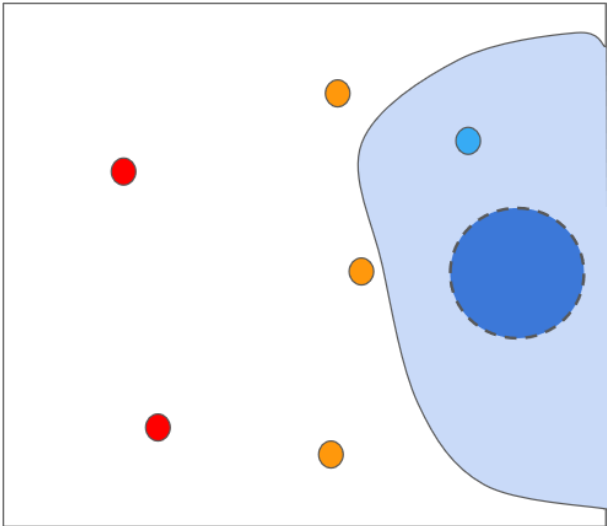
Algorithm - IV



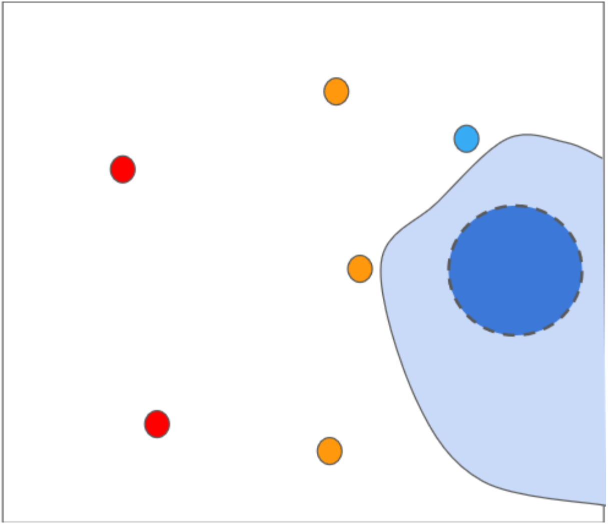
Algorithm - V



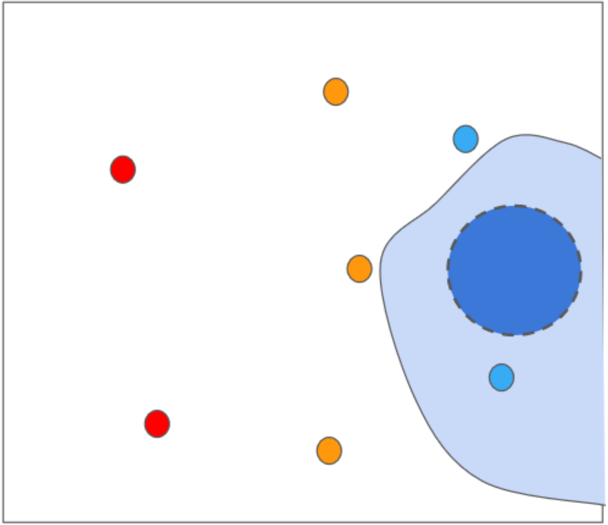
Algorithm - VI



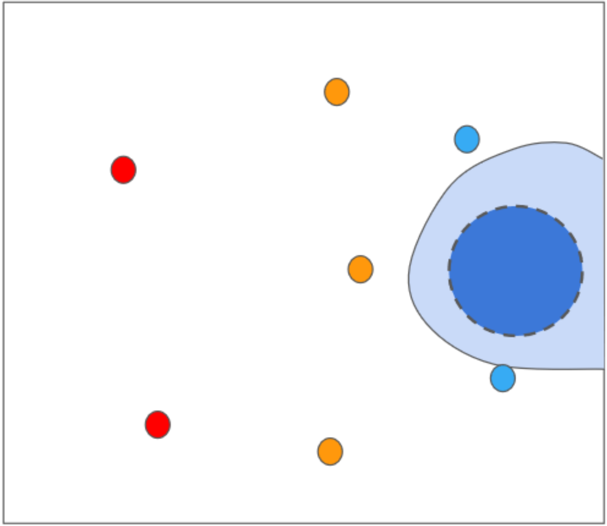
Algorithm - VII



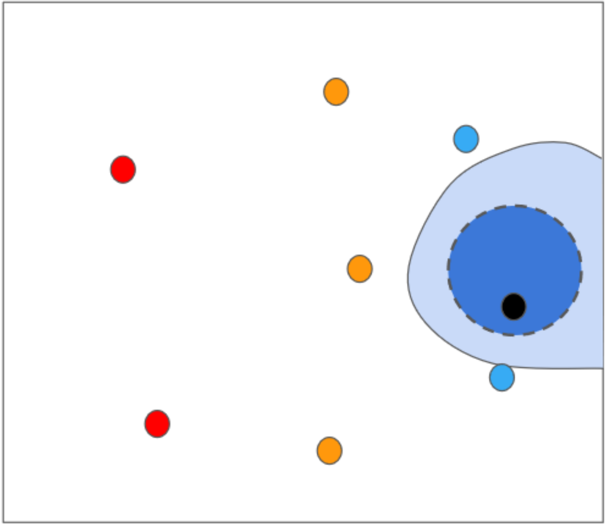
Algorithm - VIII



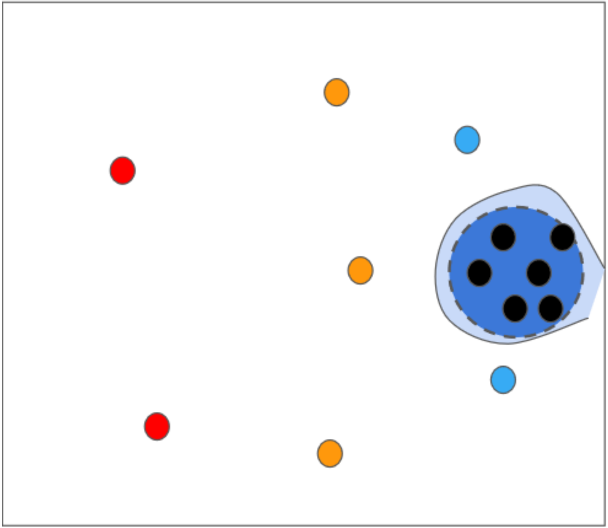
Algorithm - IX



Algorithm - X



Algorithm - XI



Probabilistic Approximation Semantics

Definition ($\mathcal{L}_0$ and $\mathcal{L}$)

$\mathcal{L}_0$: [$\subset$ STL]: atomic propositions + $\phi_1 \mathbf{U}_T \phi_2$, $\mathbf{F}_T \phi$, $\mathbf{G}_T \phi$,
that cannot be equivalently written as Boolean combinations of simpler formulas;

$$\mathbf{F}_T(\phi_1 \vee \phi_2) \equiv \mathbf{F}_T \phi_1 \vee \mathbf{F}_T \phi_2 \notin \mathcal{L}_0$$

$\mathcal{L}$: the boolean connective closure of $\mathcal{L}_0$.

Definition (Probabilistic Approximation Semantics of $\mathcal{L}$)

The probabilistic approximation function $\gamma : \mathcal{L} \times Path^{\mathcal{M}} \times [0, \infty) \to [0, 1]$ is defined by:

- $\gamma(\phi, \theta, t) = P(f_{K(\phi)}(\theta) > 0)$
- $\gamma(\neg\psi, \theta, t) = 1 - \gamma(\psi, \theta, t)$
- $\gamma(\psi_1 \wedge \psi_2, \theta, t) = \gamma(\psi_1, \theta, t) * \gamma(\psi_2, \theta, t)$
- $\gamma(\psi_1 \vee \psi_2, \theta, t) = \gamma(\psi_1, \theta, t) + \gamma(\psi_2, \theta, t) - \gamma(\psi_1 \wedge \psi_2, \theta, t)$

Test Case & Results

Automotive Requirements

- $\phi_1(\bar{v}, \bar{\omega}) = \mathbf{G}_{[0,30]}(v \leq \bar{v} \wedge \omega \leq \bar{\omega})$ (in the next 30 seconds the engine and vehicle speed never reach $\bar{\omega}$ rpm and $\bar{v}$ km/h, respectively)
- $\phi_2(\bar{v}, \bar{\omega}) = \mathbf{G}_{[0,30]}(\omega \leq \bar{\omega}) \rightarrow \mathbf{G}_{[0,10]}(v \leq \bar{v})$ (if the engine speed is always less than $\bar{\omega}$ rpm, then the vehicle speed can not exceed $\bar{v}$ km/h in less than 10 sec)
- $\phi_3(\bar{v}, \bar{\omega}) = \mathbf{F}_{[0,10]}(v \geq \bar{v}) \rightarrow \mathbf{G}_{[0,30]}(\omega \leq \bar{\omega})$ (the vehicle speed is above $\bar{v}$ km/h than from that point on the engine speed is always less than $\bar{\omega}$ rpm)

	Adaptive DEA		Adaptive GP-UCB		S-TaLiRo		
Req	nval	times	nval	times	nval	times	Alg
ϕ_1	**4.42 ± 0.53**	2.16 ± 0.61	**4.16 ± 2.40**	0.55 ± 0.30	5.16 ± 4.32	0.57 ± 0.48	UR
ϕ_1	**6.90 ± 2.22**	5.78 ± 3.88	8.7 ± 1.78	1.52 ± 0.40	39.64 ± 44.49	4.46 ± 4.99	SA
ϕ_2	**3.24 ± 1.98**	1.57 ± 1.91	7.94 ± 3.90	1.55 ± 1.23	12.78 ± 11.27	1.46 ± 1.28	CE
ϕ_2	**10.14 ± 2.95**	12.39 ± 6.96	23.9 ± 7.39	9.86 ± 4.54	59 ± 42	6.83 ± 4.93	SA
ϕ_2	**8.52 ± 2.90**	9.13 ± 5.90	13.6 ± 3.48	4.12 ± 1.67	43.1 ± 39.23	4.89 ± 4.43	SA
ϕ_3	**5.02 ± 0.97**	2.91 ± 1.20	5.44 ± 3.14	0.91 ± 0.67	10.04 ± 7.30	1.15 ± 0.84	CE
ϕ_3	**7.70 ± 2.36**	7.07 ± 3.87	10.52 ± 1.76	2.43 ± 0.92	11 ± 9.10	1.25 ± 1.03	UR

Conditional Safety Property

Falsification of Conditional Safety Property

$$\mathbf{G}_T(\phi_{cond} \to \phi_{safe})$$

Goal: exploring cases in which the formula is falsified but the antecedent condition holds
Domain Estimation Approach:

- sampling to achieve ϕ_{cond}
- sampling to falsify ϕ_{safe}

Adding one sampling routine in the Domain Estimation Algorithm.

A formula which cannot be falsified!

$\mathbf{G}_{[0,30]}(\omega \leq 3000 \to v \leq 100)$

- GP-UCB: 43% of input satisfying $\omega \leq 3000$
- DEA: 87% of input satisfying $\omega \leq 3000$

Challenges & Further studies

Results

Our Approach

- permits to reduce the minimum number of evaluations needed to falsify a model (respect to the state-of-art S-TaLiro Toolbox [1])
- can be easily customize to solve Conditional Safety Property

Further Studies

- Analyzing the sparse approximation techniques which reduces the computational cost of the Gaussian Processes
- Improving the sampling approach of Domain Estimation Algorithm (MCMC, etc..)

[1] Annpureddy, Yashwanth, et al. "S-taliro: A tool for temporal logic falsification for hybrid systems".International Conference on Tools and Algorithms for the Construction and Analysis of Systems. Springer Berlin Heidelberg, 2011.

Thank You