STOCHASTIC LOCAL SEARCH FOR FALSIFICATION OF HYBRID SYSTEMS

Jyotirmoy Deshmukh
Xiaoqing Jin
James Kapinski
&
Oded Maler

ATVA 2015

What do we mean by formally verified?

© Google Image search

© Google Image search

?

Safety

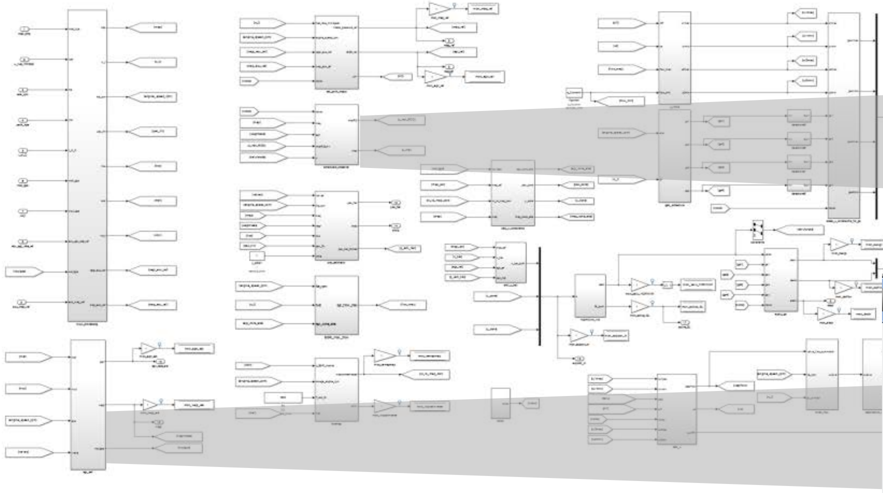
Low exhaust gas emissions

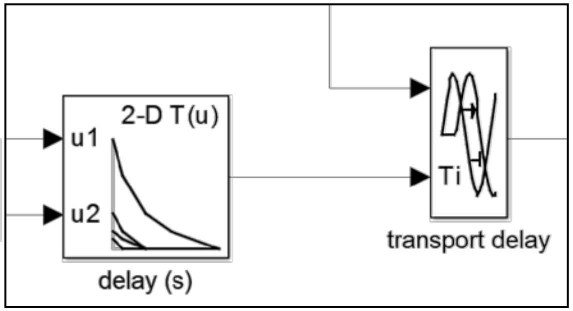
Good Fuel Efficiency

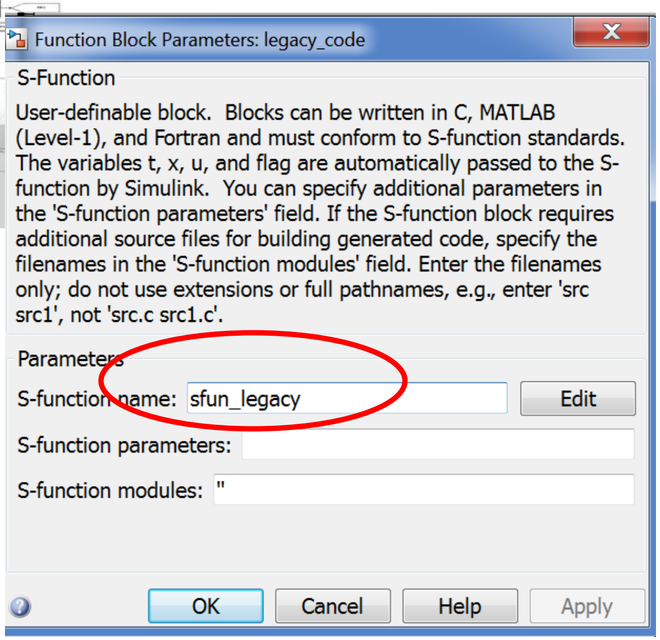
Drivability

Comfort

Industrial Models







S-Function

User-definable block. Blocks can be written in C, MATLAB (Level-1), and Fortran and must conform to S-function standards. The variables t, x, u, and flag are automatically passed to the S-function by Simulink. You can specify additional parameters in the 'S-function parameters' field. If the S-function block requires additional source files for building generated code, specify the filenames in the 'S-function modules' field. Enter the filenames only; do not use extensions or full pathnames, e.g., enter 'src src1', not 'src.c src1.c'.

Parameters

S-function name: sfun_legacy

S-function parameters:

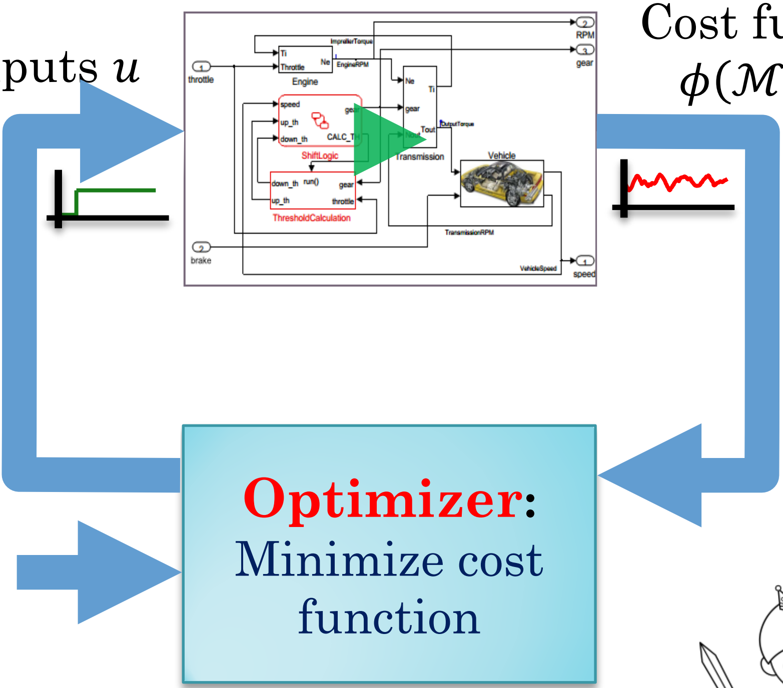
S-function modules: ''

© Google Image search

Verification and Validation Challenges

- **Complex models**
  - Discrete and continuous in time and values
  - Nonlinear dynamics (including variable time delays)
  - High dimensional Look-up-tables
  - Legacy code or other black-box components
- **Proprietary model formats**
  - Simulink, convenient but not formal
  - Translation to formal models, time consuming and error prone
- **Lack of machine-checkable requirements**

Simulation-based Falsification



Model $\mathcal{M}$

Inputs u

Cost function $\phi(\mathcal{M}, p, u)$

Optimizer: Minimize cost function

Inputs ranges φ

© Google Image search

QUANTIFYING PROPERTY SATISFACTION

- Robust satisfaction[1] [2] of temporal logic property ϕ by given simulation trace $y(\cdot)$:
  - Function mapping ϕ and y to $\mathbb{R}$
  - Positive number = y satisfies ϕ
  - Negative number = y does not satisfy ϕ
  - Moving towards zero = moving towards violation

[1] **S-TaLiRo** G. Fainekos, and G. J. Pappas. *Robustness of temporal logic specifications for continuous-time signals*. Theoretical Computer Science 2009.

[2] **Breach** A. Donzé, and O. Maler. *Robust satisfaction of temporal logic over real-valued signals*. FORMATS 2010

Simulation-based Falsification

👍

- Treat existing design artifacts as a black box
- Provide visual feedback through simulation traces

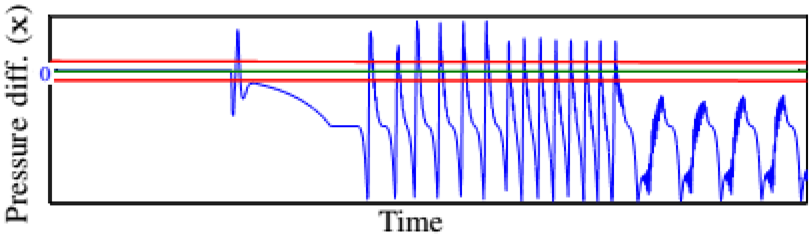
👎

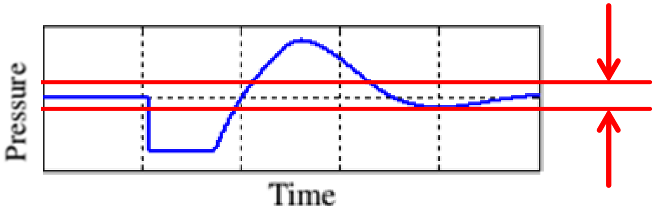
- Not verification, no guarantees of completeness (except asymptotic/probabilistic)

Many Success Stories

- Can successfully find these behaviors from prototype air path control system model

© Google Image search

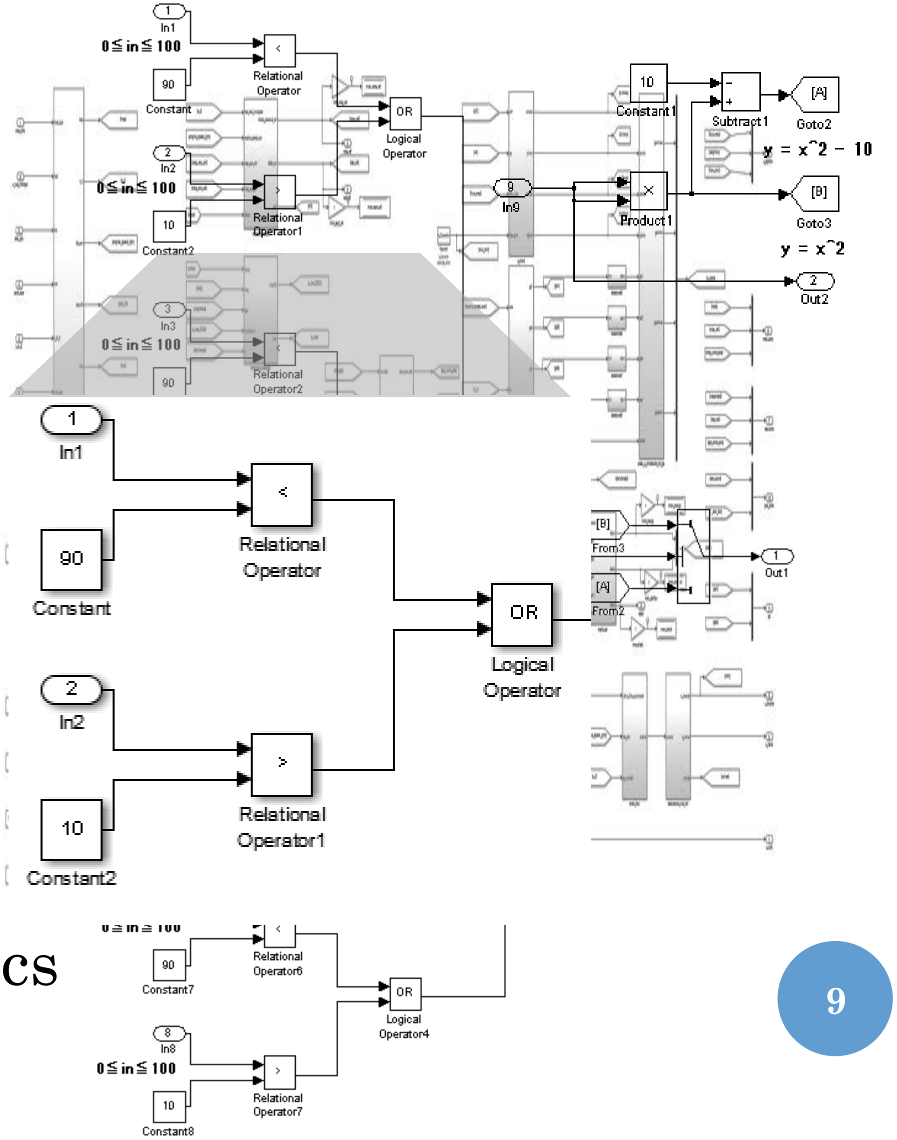




$|\mu| < 5\%$

Reality Never Ends as in a Fairy Tale

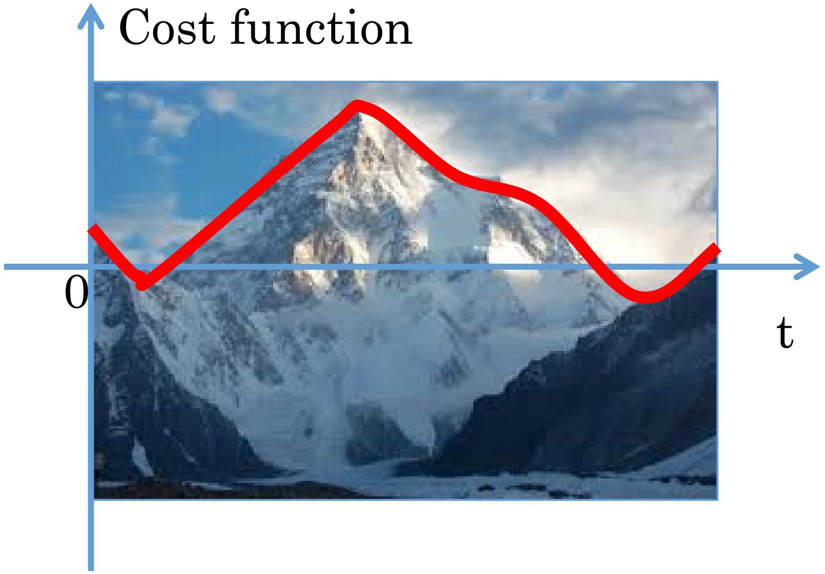
© Google Image search



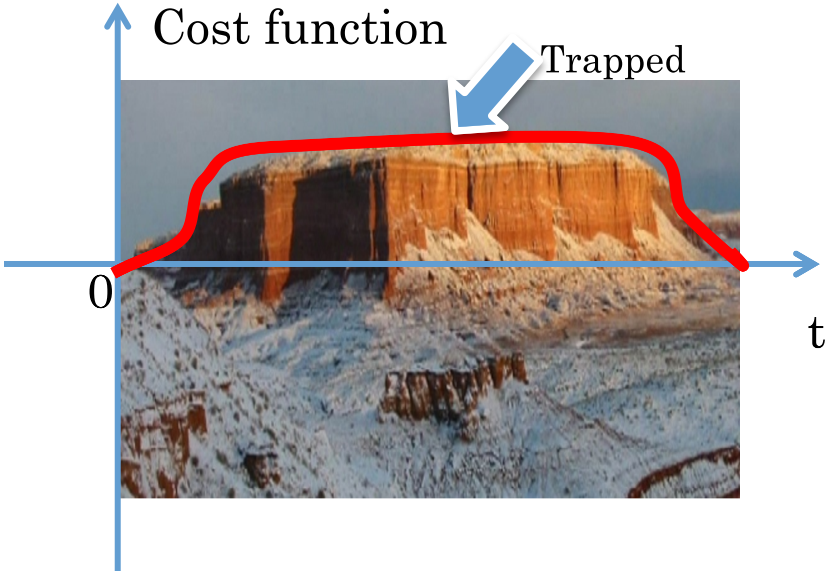
- Boolean structure
- Nonlinear system dynamics

In the Eyes of the Optimizer

- The performance of the optimizer relies on the landscape induced by the cost function



Good cost function



Bad cost function

How to Improve the Falsification Engine

- Simple ideas:

  - Tabu List + Stochastic Search

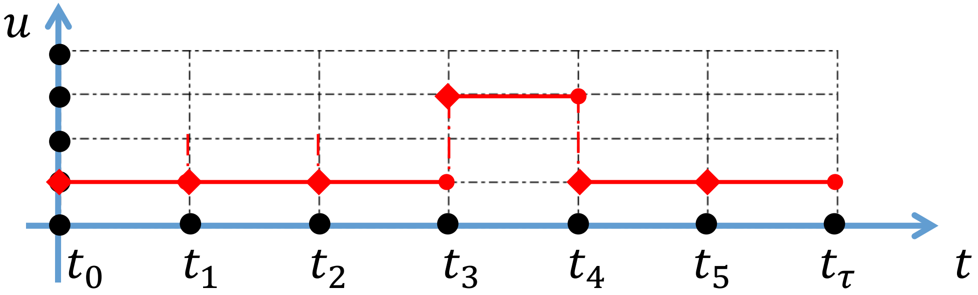
    Discretizing the input signals

  - Dynamic refinement of discretization

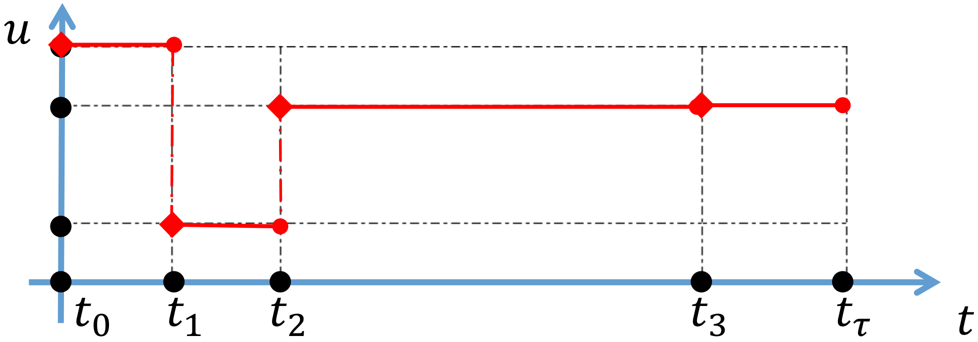
    No need to define "correct discretization strategy"

Discretization and Neighborhoods

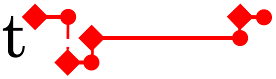
- Uniform

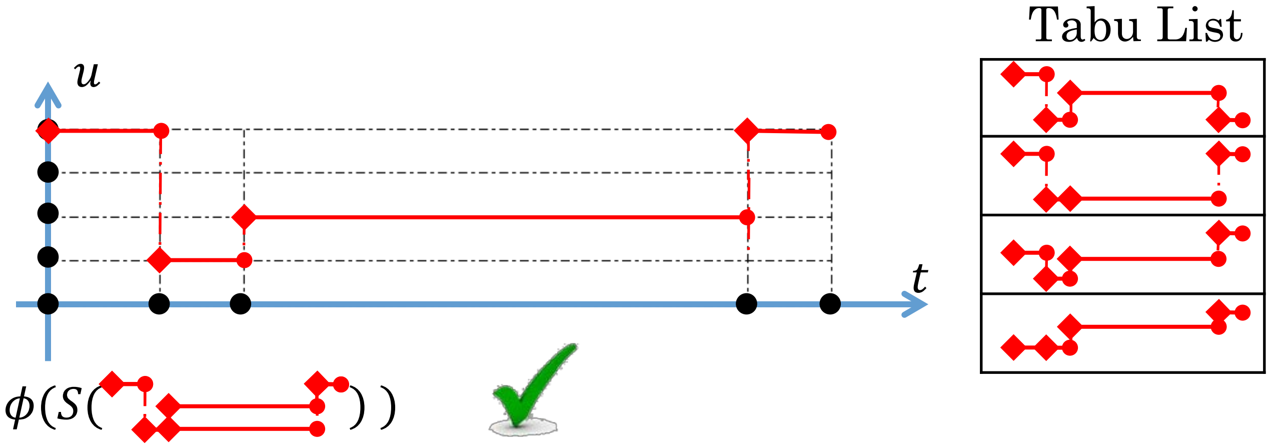


- Nonuniform



Tabu Search

- Basic Tabu search (For a given input )



Tabu list is to avoid revisiting neighbors

- Problem
  - Too many neighbors

Stochastic Local Tabu Search

- Stochastically choose a subset of neighbors

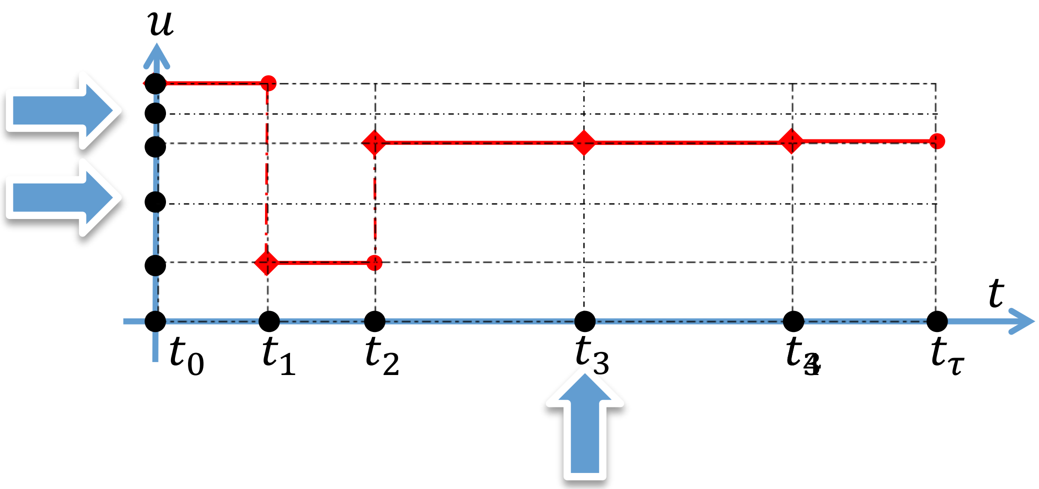
© Google Image search

- Random restarts
  - Jump out of local optimum or escape slow convergence.

- Simulated annealing-like feature
  - Seed next iteration using sub-optimal neighbors with a small probability

SEARCH SPACE REFINEMENT HEURISTICS

- Naïvely halve the discretization step size for both time and values
- Randomly refine input domain
- Refine input domain largest gap
- Refine time domain largest gap



Theoretic Guarantee Result

- Theorem 1
  - If the given system S has an input $\boldsymbol{u}^*$ that **robustly violates** the property φ, *then as the choice for the parameters of max local improvements, max refinements, and max restarts tend to* ∞, with a suitable refinement scheme, the probability that the search algorithm finds an input $\boldsymbol{u}'$ such that $\varphi(\boldsymbol{u}', \boldsymbol{y}') < 0$, where $\boldsymbol{y}' = S(\boldsymbol{u}')$, tends to 1.

- Definition (Robust Violation)
  - $\boldsymbol{y} = S(\boldsymbol{u}) \wedge \varphi(\boldsymbol{u}, \boldsymbol{y}) < 0$

    $\Rightarrow \forall\, \boldsymbol{u}' \in \mathrm{NB}_{\delta,\epsilon}(\boldsymbol{u}) | \boldsymbol{y}' = S(\boldsymbol{u}') \wedge \varphi(\boldsymbol{u}', \boldsymbol{y}') < 0$

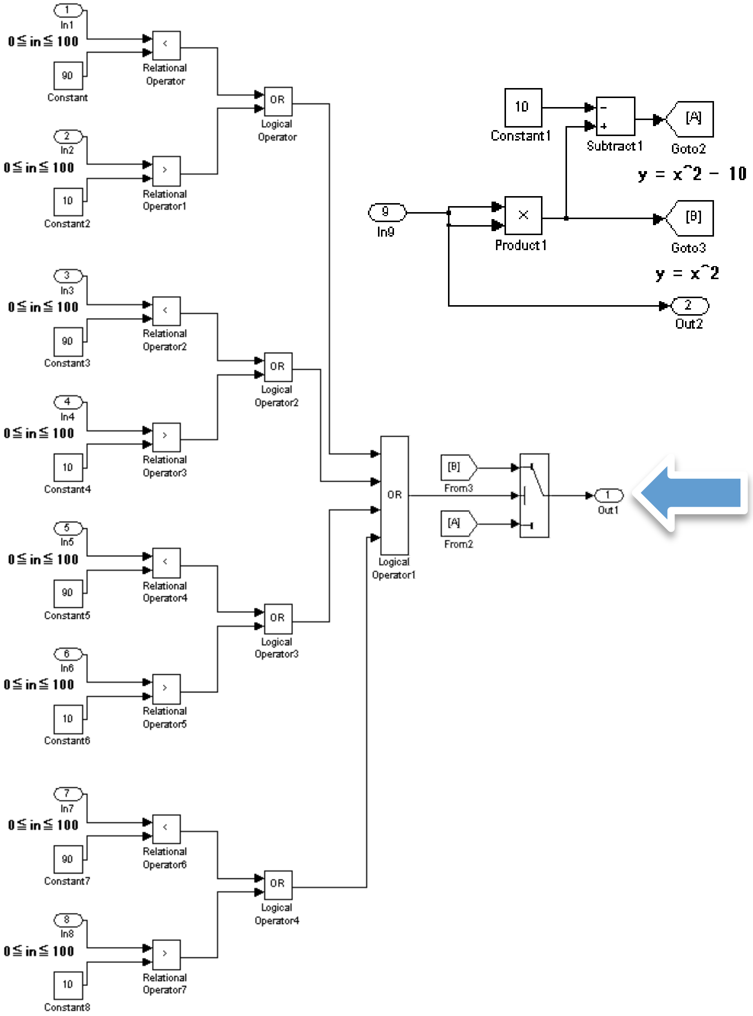
Experimental Results

- Mode-specific Reference Selection Model (MRS)
- Check property

  Output1 < -8
- Why it is hard?

$$\bigwedge_{i\in[1,4]} ((w^{2i}(t) > 90) \wedge (w^{2i-1}(t) < 10)$$

$P(error) \cong 10^{-8}$



Experimental Results (Continued)

- SITAR (No refinement)

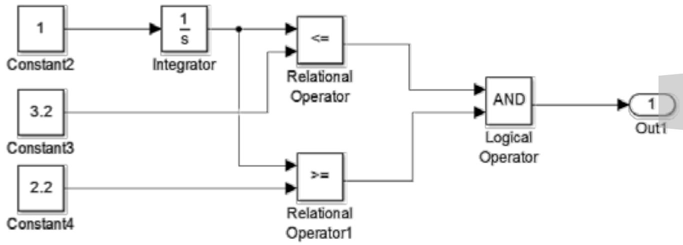
Initial Discretization	#(input disc. pt.)	#(time disc. pt.)	Time (sec)	Num (Sim)	Falsified
NonUniform	35	3	50	233	✓
Uniform	35	3	241	2058	✓

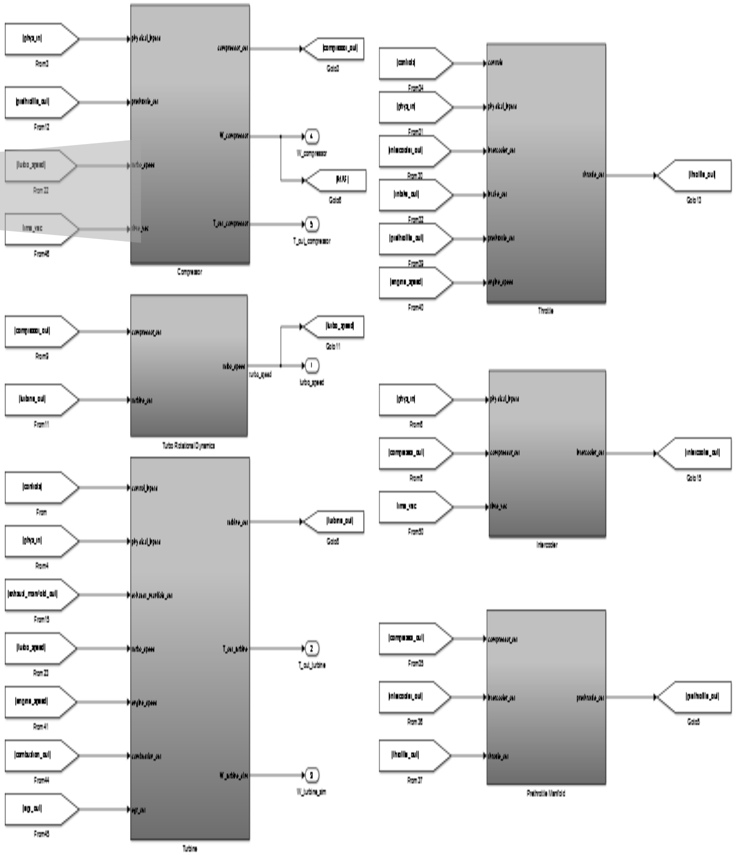
- S-TaLiRo

#(disc. pt.)	Time (sec)	Num (Sim)	Falsified
40	745	1000	✗
40	2121	3000	✗

Experimental Results (Continued)

- Rate Detection (RD)





- Check Property

The decrease rate is within $[\zeta_1, \ \zeta_2]$ in a given time window $[\tau_1, \tau_2]$

Experimental Results (Continued)

- SITAR (With refinement)

Initial Discretization	#(input disc. pt.)	#(time disc. pt.)	Time (sec)	Num (Sim)	Falsified
NonUniform	3	**2***	17	206	✓
Uniform	3	**3***	47	575	✓
Uniform	3	**4***	28	349	✓

* (allow refinement of discretization points)

- S-TaLiRo

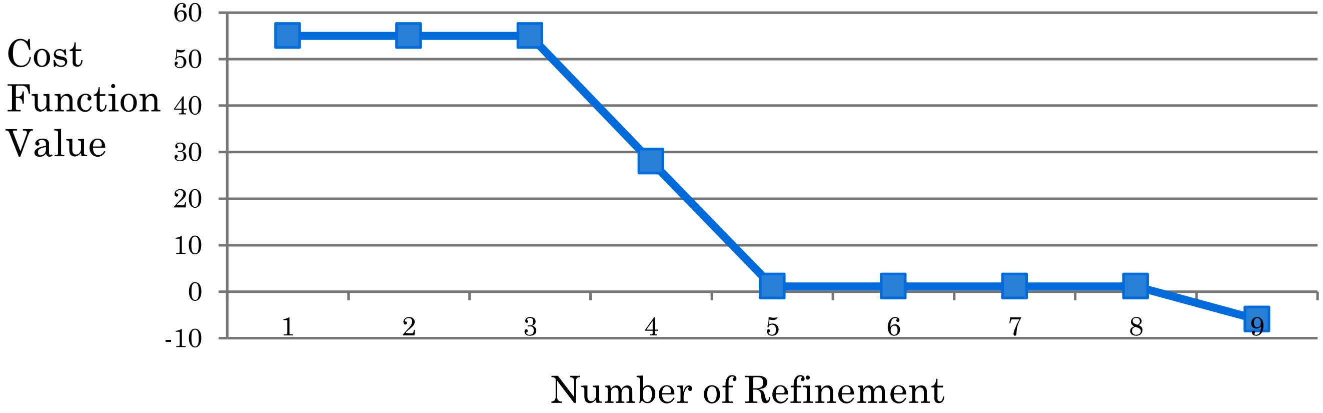
#(disc. pt.)	Time (sec)	Num (Sim)	Falsified
2	141	2000	✗
4	141	2000	✗
8	1	17	✓

EXPERIMENTAL RESULTS (CONTINUED)

- SITAR

Initial Discretization	#(input disc. pt.)	#(time disc. pt.)	Time (sec)	Num (Sim)	Falsified
NonUniform	3	**2***	17	206	✓
Uniform	**3**	**3***	**47**	**575**	✓
Uniform	3	**4***	28	349	✓

- Cost function value decreased during refinement



Experimental Results (Continued)

- Toyota prototype model: Powertrain Air Control (PTAC) System
  - 2 Electronic Control Units (ECU)
  - High fidelity plant model
- Check property: the overshoot $< \pi$
- SITAR (Without refinement)

Initial Discretization	#(input disc. pt.)	#(time disc. pt.)	Time (sec)	Num (Sim)	Falsified
Uniform	3	3	8784	39	✓

- S-TaLiRo

#(disc. pt.)	Time (sec)	Num (Sim)	Falsified
6	26568	71	✓

Discussion and future work

- Lessons learnt
  - Simple ideas sometimes work surprisingly well
  - Adaptive refinement balancing the efficiency and effectiveness

- Future work
  - Add coverage metric for the input sequence space
  - Used advanced spatial data structure for Tabu List
  - Consider model structure to inform refinement decisions

THANKS FOR YOUR ATTENTION