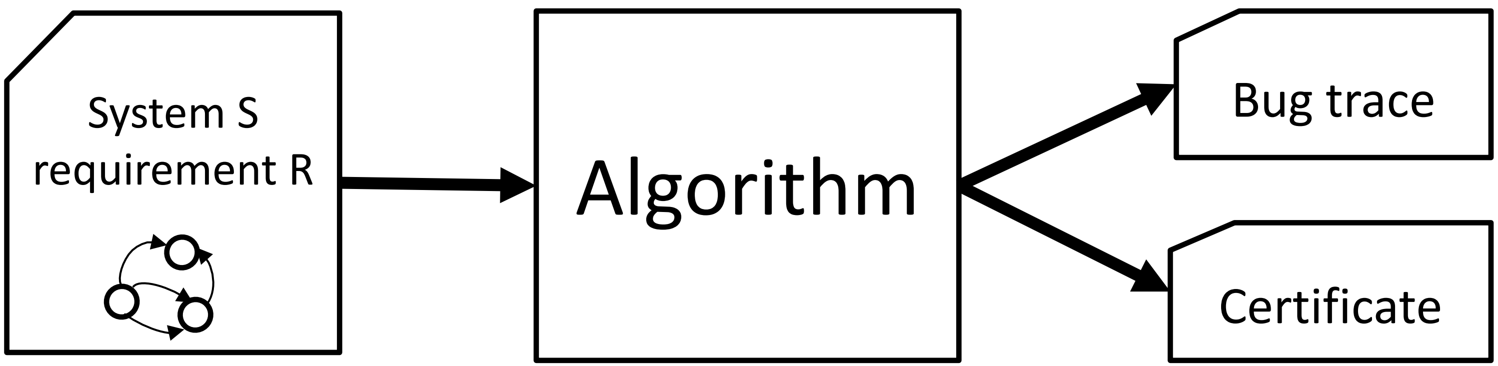
Lecture 7 and Tutorial 4: Simulation-driven Verification

Sayan Mitra
Electrical & Computer Engineering
Coordinated Science Laboratory
University of Illinois at Urbana Champaign

Safety verification problem



Is there a behavior of system S violating safety requirement R within time bound T?

Yes -> bug-trace -> design improvement

No -> safety proof -> certification

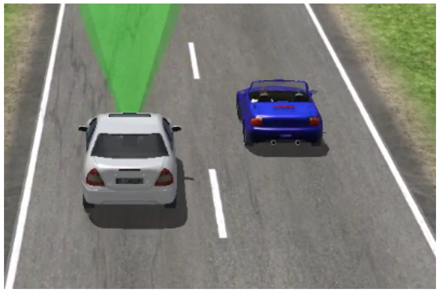
Safety verification problem

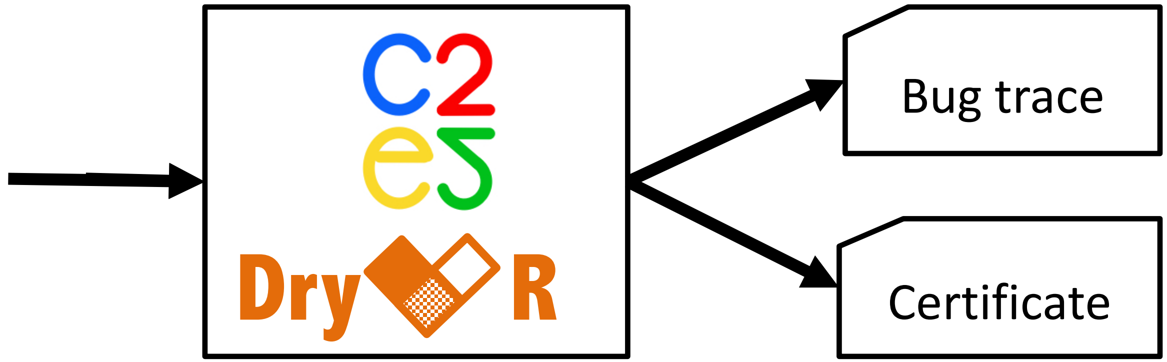
Is there a behavior of system S violating safety requirement R within time bound T?

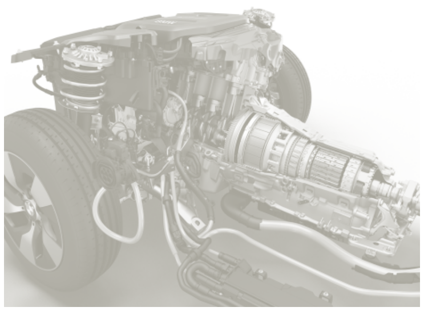
Yes -> bug-trace -> design improvement

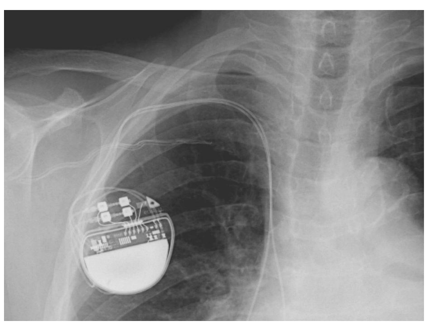
No -> safety proof -> certification

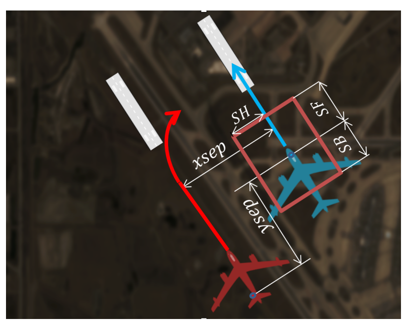
Safety verification problem

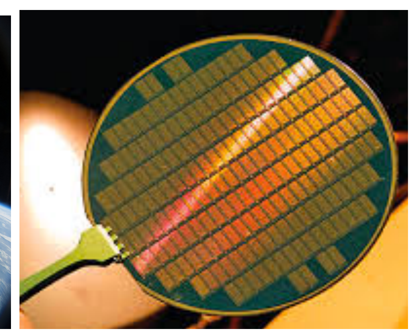






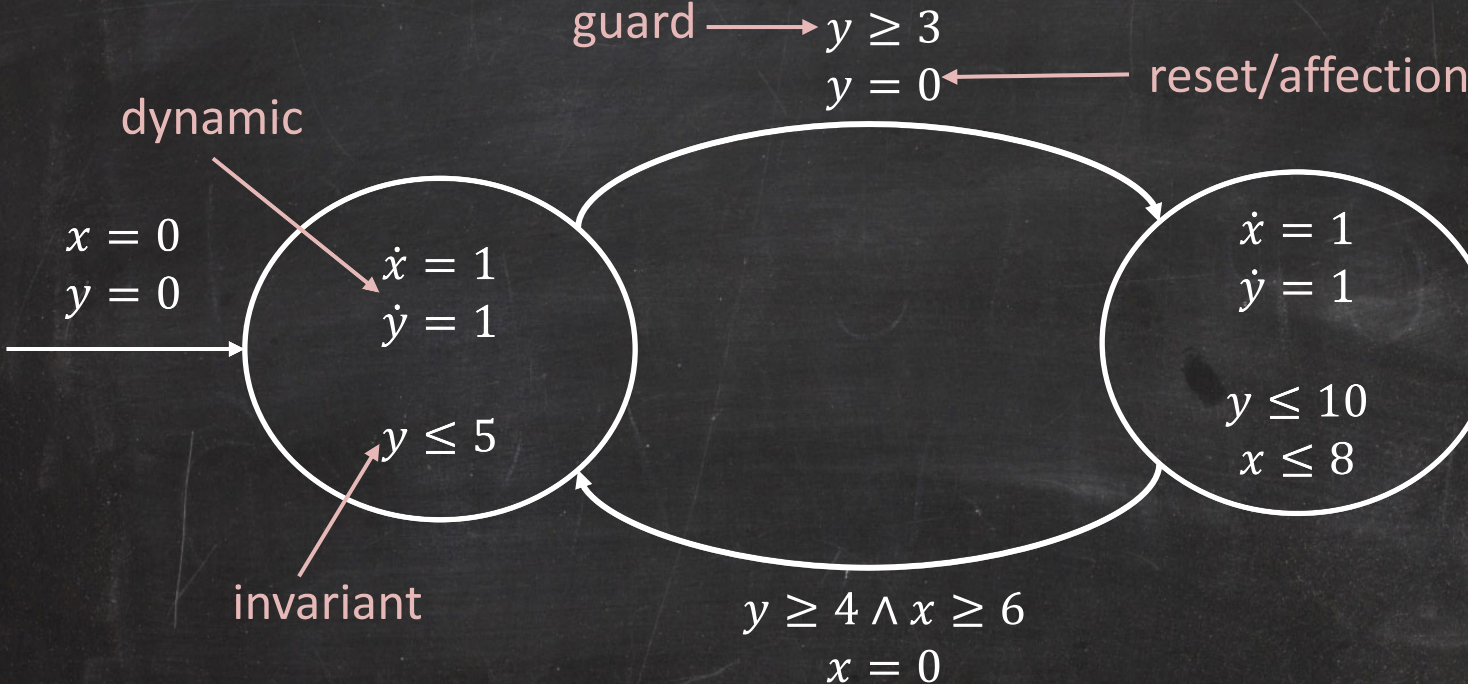




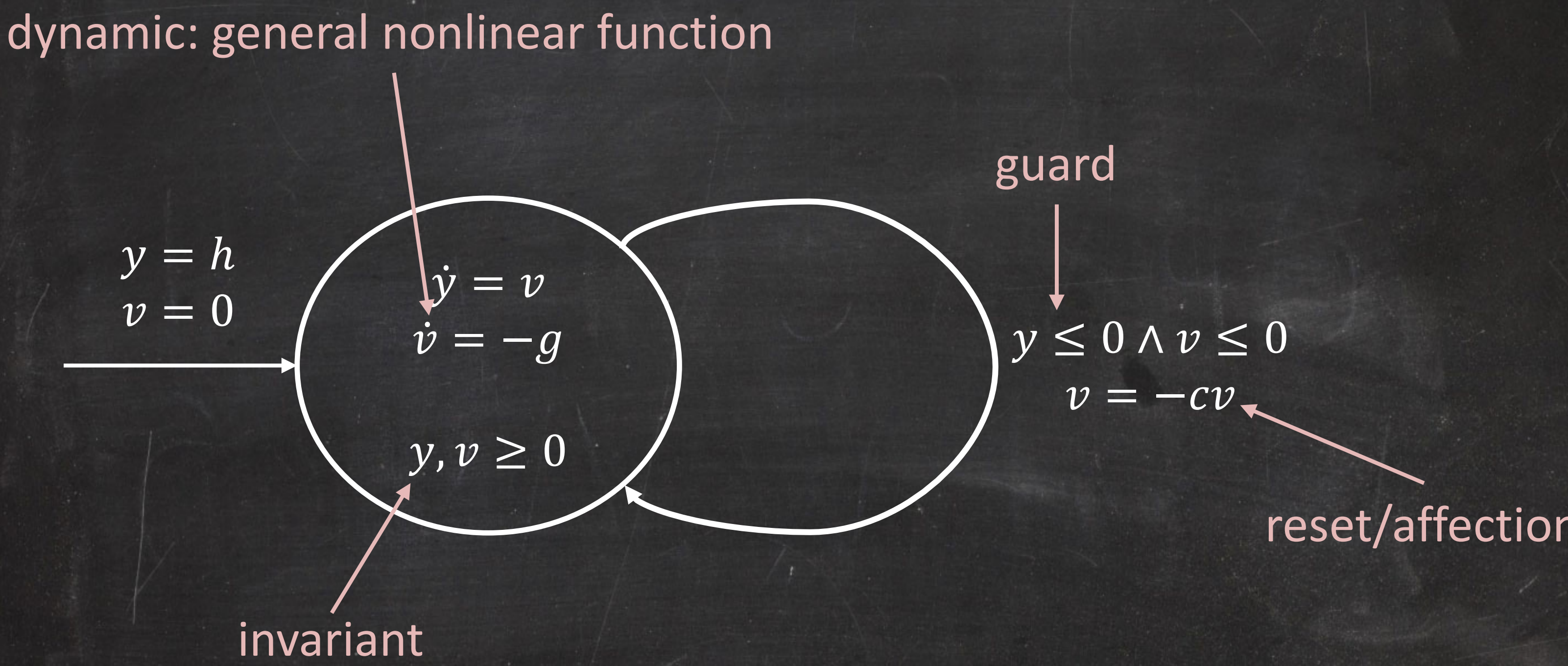


Slides by Sayan Mitra (mitras@illinois.edu)

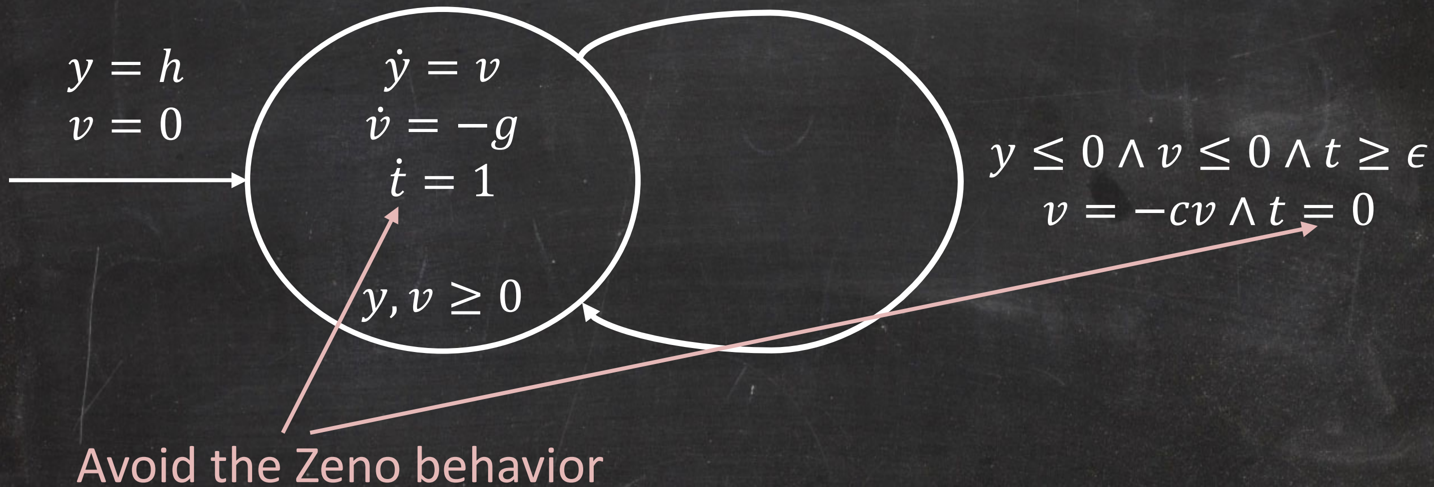
Recall: timed automata



Recall: bouncing ball



Recall: bouncing ball



Summary of C2E2

- Input: hyxml file
- Properties: initial set + unsafe set
- Simulate and/or verification
- Plotter

Outline

Introduction and C2E2 demo

Model-based sensitivity

- Simulation-driven verification algorithm
- Discrepancy function
- Matrix measure and sensitivity
- More examples

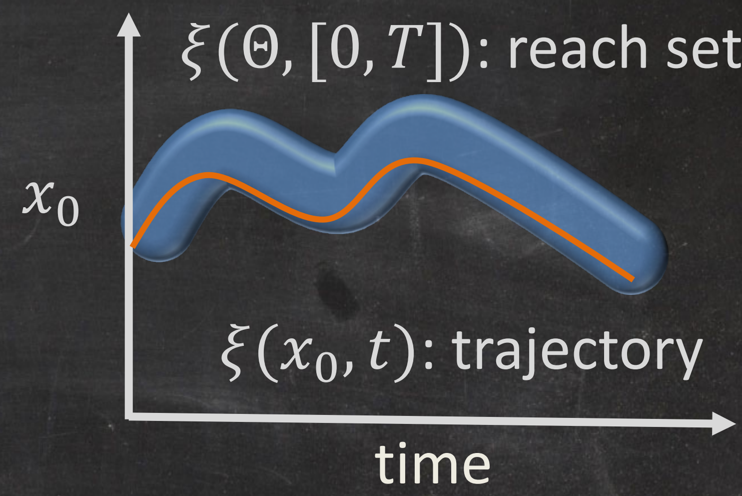
Next lecture on Thursday:

- New modeling questions with DryVR

System models and nonlinear dynamical model

nonlinear dynamical model

$$\dot{x}(t) = f\big(x(t)\big)$$
$$\Theta, \mathrm{U} \subseteq \mathbb{R}^n$$



Safety verification problem $\xi(\Theta, [0, T]) \cap U = \emptyset$?

Slides by Sayan Mitra (mitras@illinois.edu)

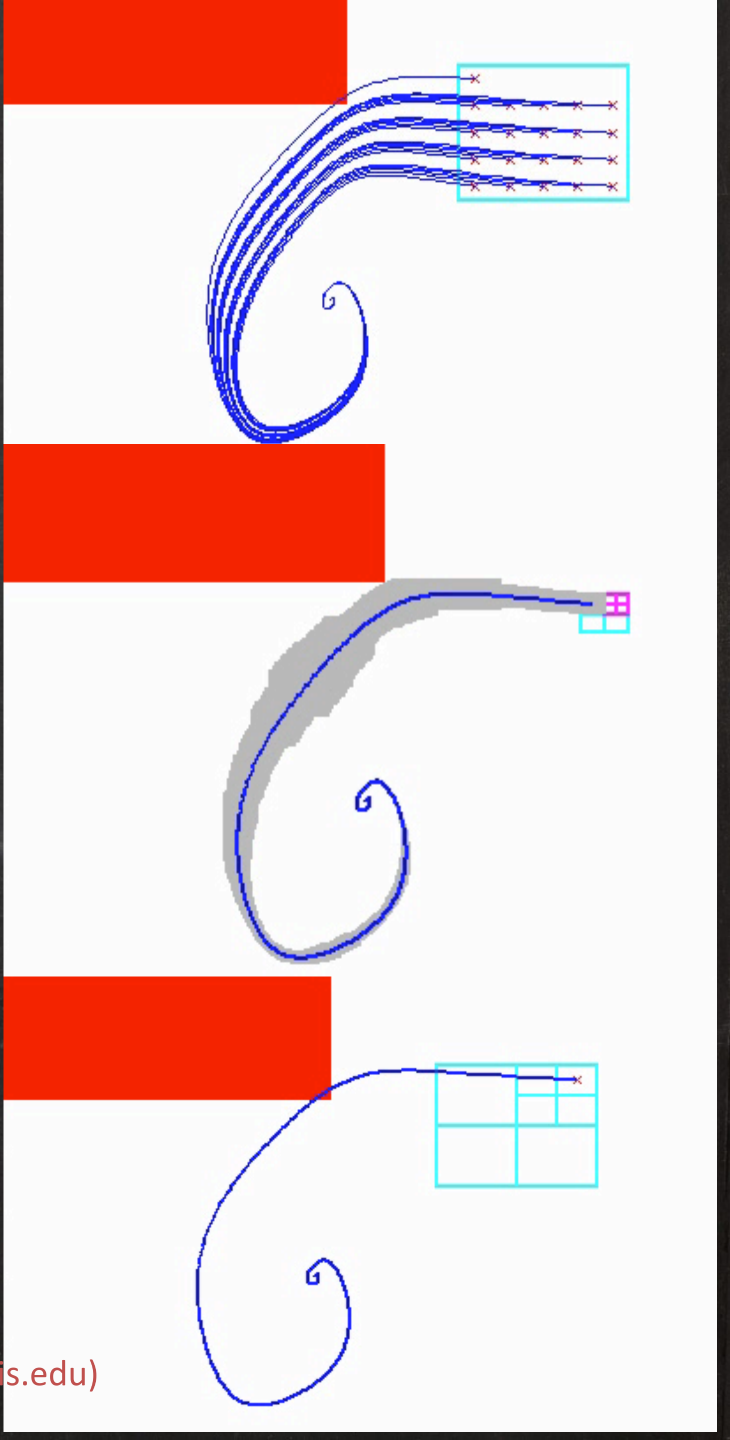
Simulations to safety proofs

- Given start Θ and target U
- Compute finite cover $\cup_i B(x_i, \delta) \supseteq \Theta$
- Simulate from the center x_0 of each cover to get $\xi(x_0, \{t_1, \ldots, t_k\})$
- **Bloat** simulation so that

$$\xi(x_0, .) \oplus \beta \supseteq \xi(B(x_0, \delta), [0, T])$$

- Check intersection/containment with U
- Refine cover if needed and repeat …

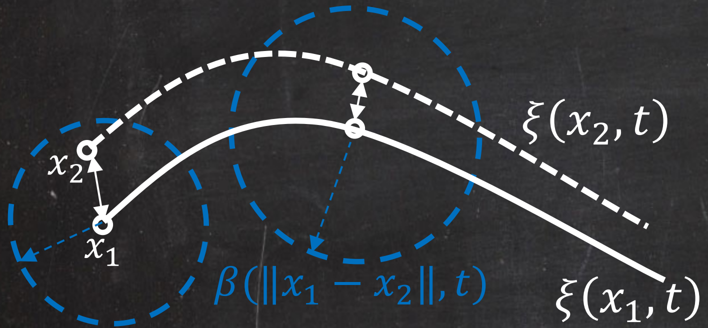
How to bloat or generalize simulations?



Brief history

2000	On Systematic Simulation of Open Continuous Systems	Kapinski et al.
2006	Verification using simulation	Girard and Pappas
2007	Robust Test Generation and Coverage for Hybrid Systems	Julius, Fainekos, et al.
2010	Breach, a toolbox for verification and parameter synthesis of hybrid systems.	Donzé
2013	Verification of annotated models from executions.	Duggirala, *Mitra,* Viswanathan

Main problem: How to quantify generalization?

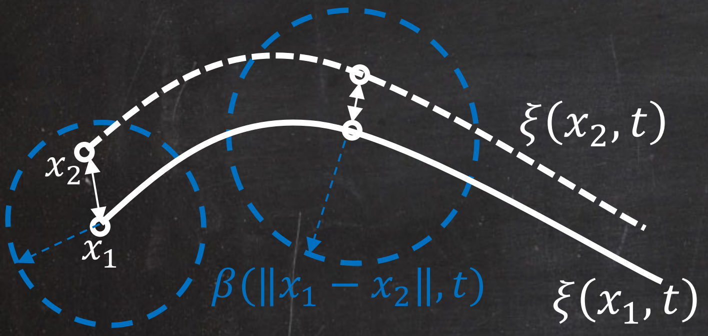


- Discrepancy formalizes generalization :
- Discrepancy is a continuous function β that bounds the distance between neighboring trajectories

$$\|\xi(x_1, t) - \xi(x_2, t)\| \leq \beta(\|x_1 - x_2\|, t),$$

- From a single simulation of $\xi(x_1, t)$ and discrepancy β we can over-approximate the reachtube

A simple example of discrepancy function



- If $f(x)$ has a Lipschitz constant L :

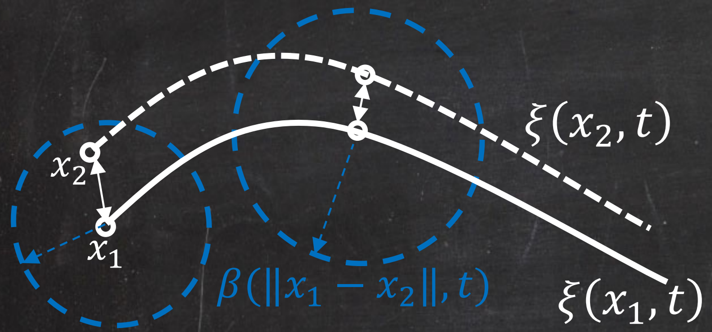
$$\forall x, y \in \mathbb{R}^n, \|f(x) - f(y)\| \leq L\|x - y\|$$

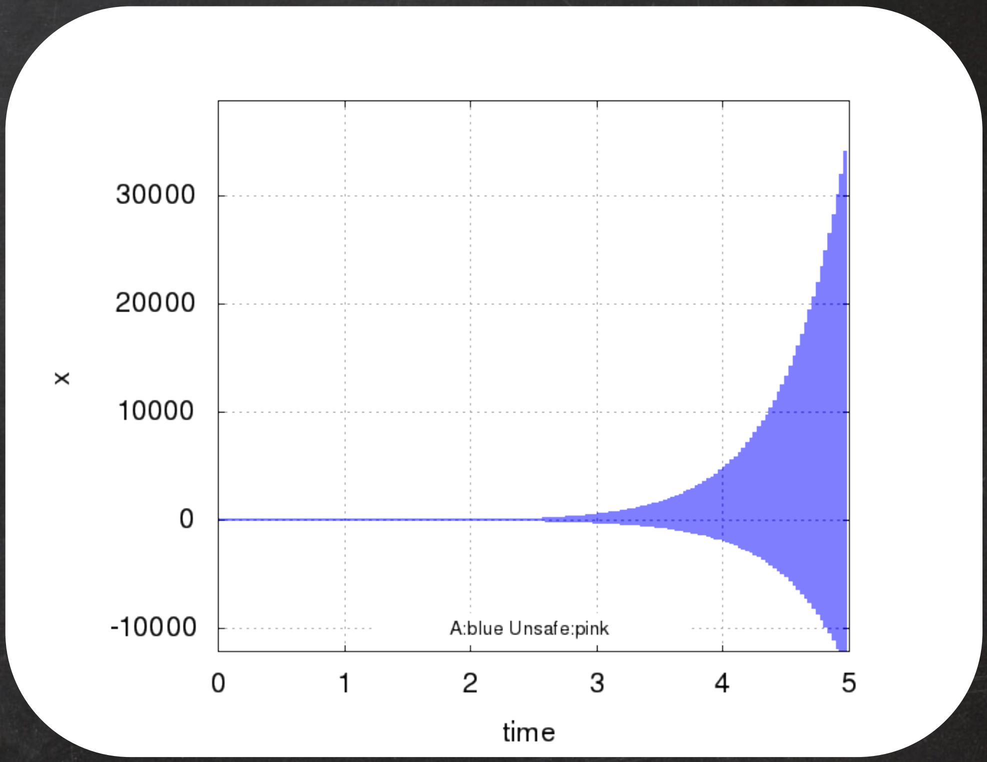
Example: $\dot{x} = -2x$, Lipschitz constant $L = 2$

- then a (bad) discrepancy function is

$$\|\xi(x_1, t) - \xi(x_2, t)\| \leq \|x_1 - x_2\|e^{Lt} = \beta(\|x_1 - x_2\|, t)$$

A simple example of discrepancy function

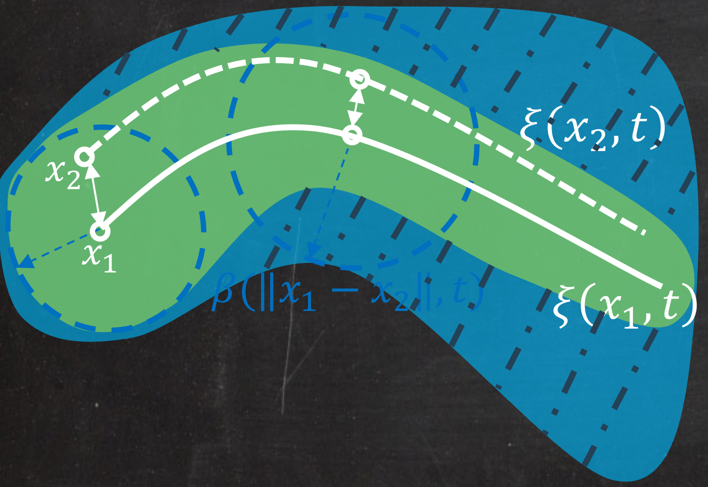




$\dot{x} = -2x$, Lipschitz constant $L = 2, \delta = 1$

Slides by Sayan Mitra (mitras@illinois.edu)

What is a good discrepancy ?



General: Applies to general nonlinear f

Accurate: Small error in β

Effective: Computing β is fast (in practice)

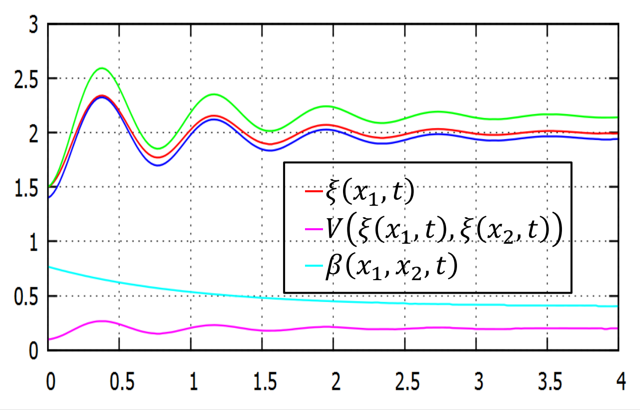
Discrepancy quantifies sensitivity

$\xi(B(x_0, \delta), [0, T]) \subseteq \xi(x_0, .) \oplus \beta$

reach set over-approximated by simulation and sensitivity

Definition. $\beta: \mathbb{R}^{2n} \times \mathbb{R}^{\geq 0} \rightarrow \mathbb{R}^{\geq 0}$ defines a discrepancy of the system if for any two states x_1 and $x_2 \in X$, for any t,

- $|\xi(x_1, t) - \xi(x_2, t)| \leq \beta(x_1, x_2, t)$ and
- $\beta \rightarrow 0$ as $x_1 \rightarrow x_2$



Computing discrepancy

$|\xi(x_1, t) - \xi(x_2, t)| \leq e^{Lt}|x_1 - x_2|$

L: Lipschitz constant of *f(.)*

$\dot{x} = -2x$ Lipschitz constant L=2

$|\xi(x_1, t) - \xi(x_2, t)| \leq e^{\mu t}|x_1 - x_2|$

μ: Matrix measure of Jacobian J_f

$$\mu_p(\mathrm{A}) = \lim_{t \to 0^+} \frac{\left|\left| I + tA \right|\right|_p - \left|\left| I \right|\right|_p}{t}$$

$\mu_p = -2$ for above linear system

Matrix measure for $A \in \mathbb{R}^{n \times n}$

Matrix norm

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}$$

$$\|A\|_2 = \sqrt{\lambda_{max}(A^T A)}$$

Matrix measure [Dahlquist 59]:

$$\mu(A) = \lim_{t \to 0^+} \frac{\|I + tA\| - \|I\|}{t}$$

2-norm: $\mu(A) = \lambda_{max}\left(\frac{A+A^T}{2}\right)$

Computing μ

Vector norm	Induced matrix norm	Matrix measure
$\lvert x\rvert_1 = \Sigma\lvert x_j\rvert$	$\lVert A\rVert_1 = \max_j \Sigma_i \lvert a_{ij}\rvert$	$\mu_1(A) = \max_j (a_{jj} + \Sigma_{i\neq j} \lvert a_{ij}\rvert)$
$\lvert x\rvert_2 = \sqrt{\Sigma x_j^2}$	$\lVert A\rVert_2 = \sqrt{\max_j \lambda_j(A^T A)}$	$\mu_2(A) = \max_j \frac{1}{2}(\lambda_j(A + A^T))$
$\lvert x\rvert_\infty = \max_j \lvert x_j\rvert$	$\lVert A\rVert_\infty = \max_i \Sigma_j \lvert a_{ij}\rvert$	$\mu_\infty(A) = \max_i (a_{ii} + \Sigma_{i\neq j} \lvert a_{ij}\rvert)$

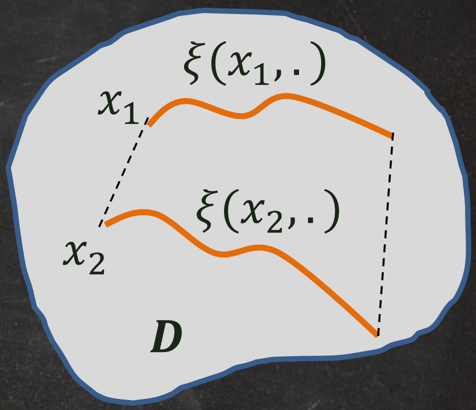
Table from: Reachability Analysis of Nonlinear Systems Using Matrix Measures [Maidens and Arcak, 2015]

Matrix measures can be used to compute discrepancy

Theorem [Sontag 10]: For any $\mathcal{D} \subseteq \mathbb{R}^n$,if the matrix measure of the Jacobian $\mu\big(J(t,x)\big) \leq c$ over $\mathcal{D}$, and all trajectories starting from the line remains in $\mathcal{D}$ then the solutions satisfies:

$$|\xi(x_1,t) \; - \xi(x_2,t) \mid \leq |\boldsymbol{x_1} - \boldsymbol{x_2}|e^{ct}$$

- That is, $|x_1 - x_2|e^{ct}$ is a discrepancy function
- Here J is the Jacobian of $f(x)$
- This c can be negative and is usually much smaller than the Lipschitz constant

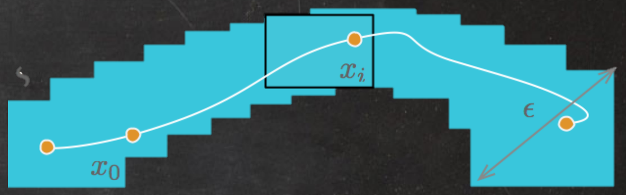


Strategies for computing μ

- Define $y(t) = \xi(x_1, t) - \xi(x_2, t)$
- Let interval matrix **A** be such that for all $x \in D, J_f(x) \in \boldsymbol{A}$,
- Then $\dot{y}(t) = A(t)y(t)$, for some A(t) $\in \boldsymbol{A}$
- This gives discrepancy $\beta\left(||x_1 - x_2||_M, t\right) = ||x_1 - x_2||_M e^{\frac{\gamma^*}{2}t}$, where $\gamma^* = \min \gamma$ s.t. $A^T M + MA \preccurlyeq \gamma M, \forall A \in \boldsymbol{A}$ --- (*)
- Solving (*)
  - Fix $M = I,\ \gamma^* = \lambda_{max}(A + A^T) + error$

Simulation ⊕ β → Reachtubes

$\boldsymbol{simulation(x_0, h, \epsilon, T)}$ of gives sequence $S_0, \dots, S_k$: $dia(S_i) \le \epsilon$ & at any time $t \in [ih, (i+1)h]$, solution $\xi(x_0, t) \in S_i$.



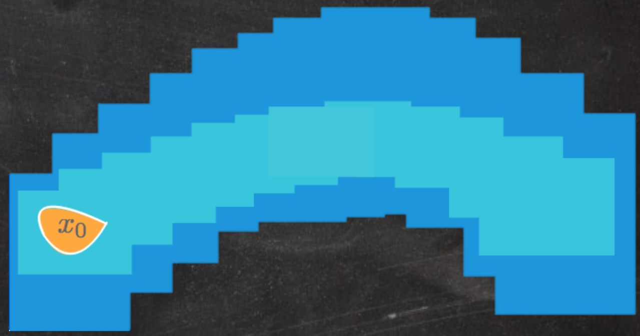
$\langle S_0, \dots, S_k, \epsilon_1 \rangle \leftarrow valSim(x_0, T, f)$

For each $i \in [k]$, $\epsilon_2 \leftarrow \sup_{t \in T_i, x, x' \in B_\delta(x_0)} \beta(x_1, x_2, t)$

$R_i \leftarrow B_{\epsilon_2}(S_i)$

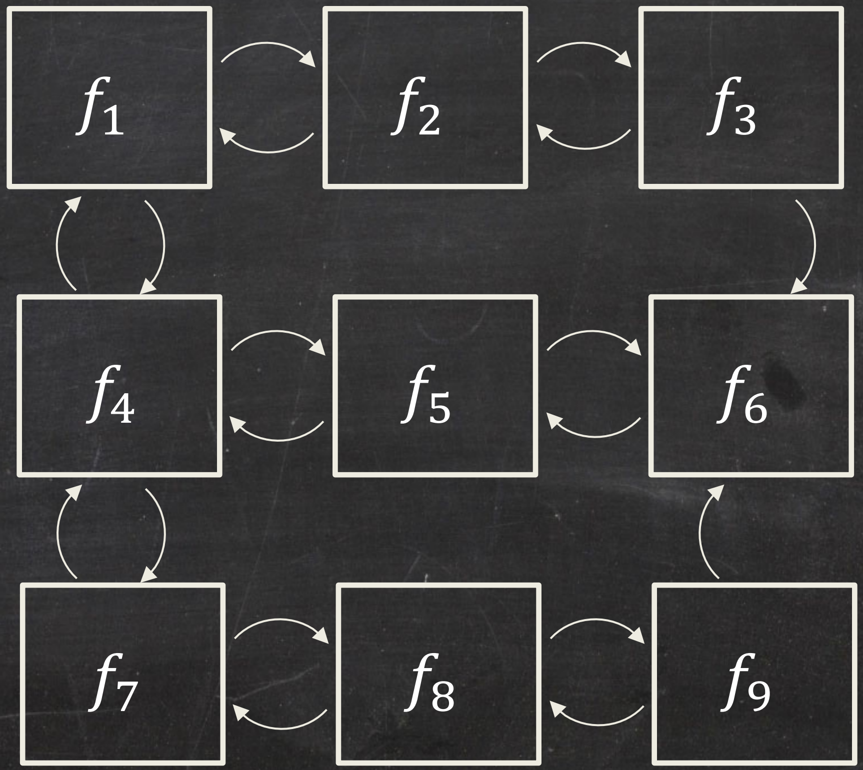
Example 1: $\dot{v} = \frac{1}{2}(v^2 + w^2); \dot{w} = -v$

- $J_f(v, w) = \begin{bmatrix} v & w \\ -1 & 0 \end{bmatrix}$
- $\gamma^* = 1.0178$ upper-bound on eigen values of the symmetric part of $J_f(v, w)$ over $\boldsymbol{D} = [-2, -1] \times [2,3]$
- $||\xi(x_1, t) - \xi(x_2, t)|| \le ||x_1 - x_2|| e^{1.0178t}$ while $x \in \boldsymbol{D}$
- Uniform in all directions



Example 2: $\dot{x} = \begin{bmatrix} 0 & 3 \\ -1 & 0 \end{bmatrix} x$; Eigenvalues $\pm\sqrt{3}\, i$

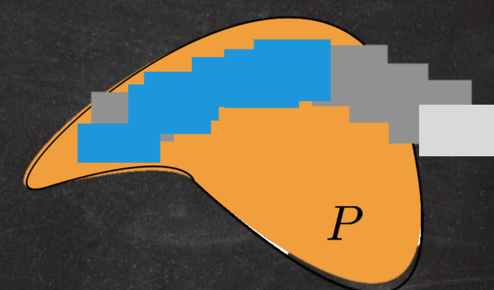
Hybrid models



Hybrid Reachtubes

Track & propagate *may* and *must* fragments of reachtube

$$\mathit{tagRegion}(R, P) = \begin{cases} must & R \subseteq P \\ may & R \cap P \neq \emptyset \\ not & R \cap P = \emptyset \end{cases}$$

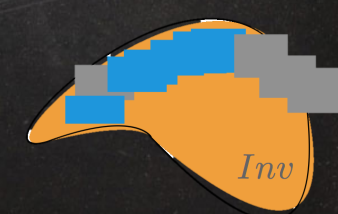


$\mathit{invariantPrefix}(\psi, S) =$

$\langle R_0, tag_0, \ldots, R_m, tag_m \rangle$, such that either

$tag_i = must$ if all the $R_j's$ before it are must

$tag_i = may$ if all the $R_j's$ before it are at least may and at least one of them is not must



Guarantees for bounded invariance verification using discreapancy

Theorem. (Soundness). If Algorithm returns safe or unsafe, then A is safe or unsafe.

Definition Given HA $A \ = \langle V, Loc, A, D, T \ \rangle$, an **$\epsilon$-perturbation** of A is a new HA A' that is identical except, $\Theta' = B_\epsilon(\Theta)$, $\forall \ \ell \in Loc, Inv' = B_\epsilon(Inv)$ (b) a $\in$ A, $Guard_a = B_\epsilon(Guard_a)$.

A is **robustly safe** iff $\exists \epsilon > 0$, such that A' is safe for U_ϵ upto time bound T, and transition bound N. Robustly unsafe iff $\exists \ \epsilon < 0$ such that A' is safe for U_ϵ.

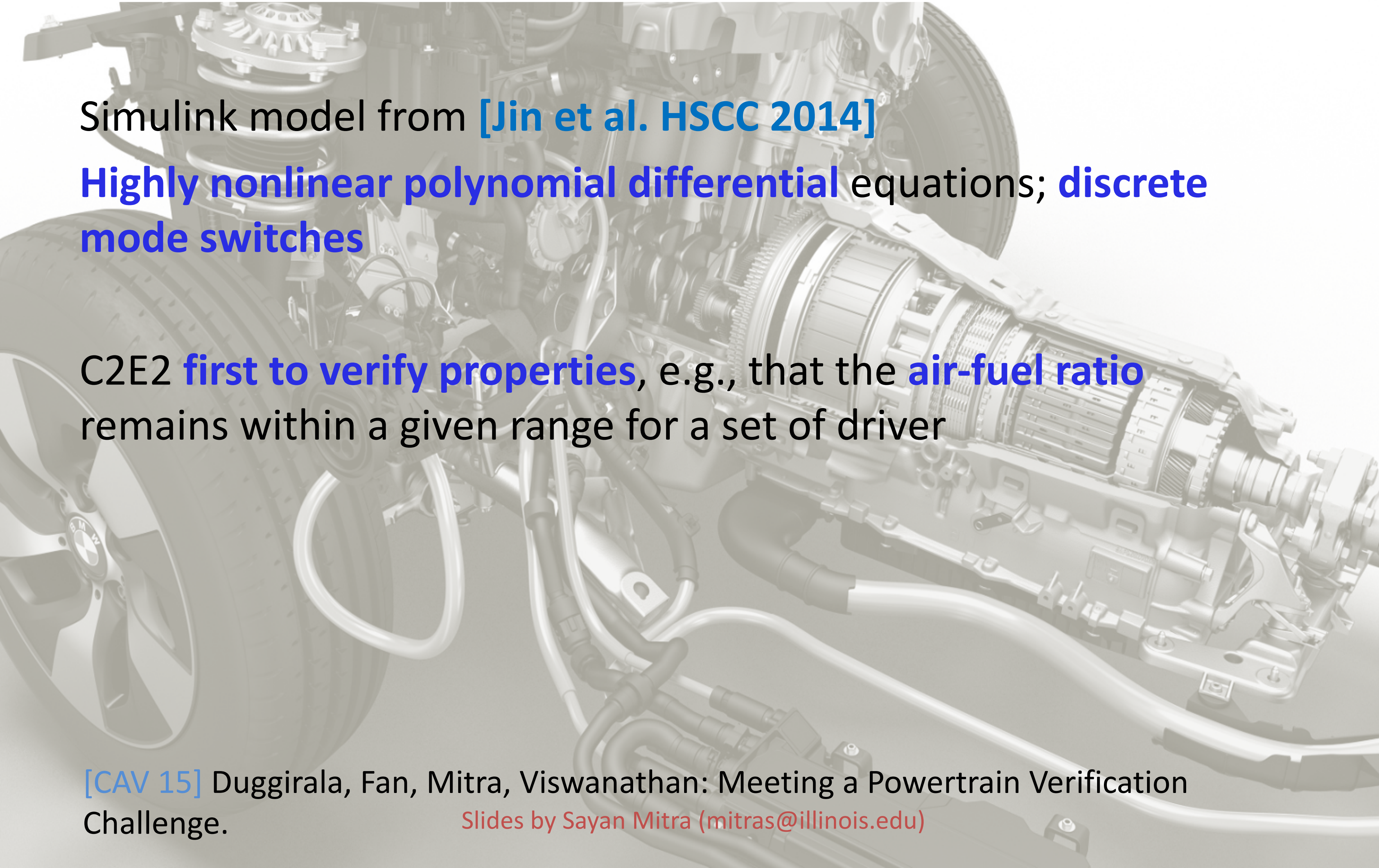
Theorem. (Relative Completeness) Algorithm always terminates whenever the A is either robustly safe or robustly unsafe.

Compare execute check engine

static-dynamic analysis of nonlinear hybrid models

Slides by Sayan Mitra (mitras@illinois.edu)

Powertrain control verification benchmark



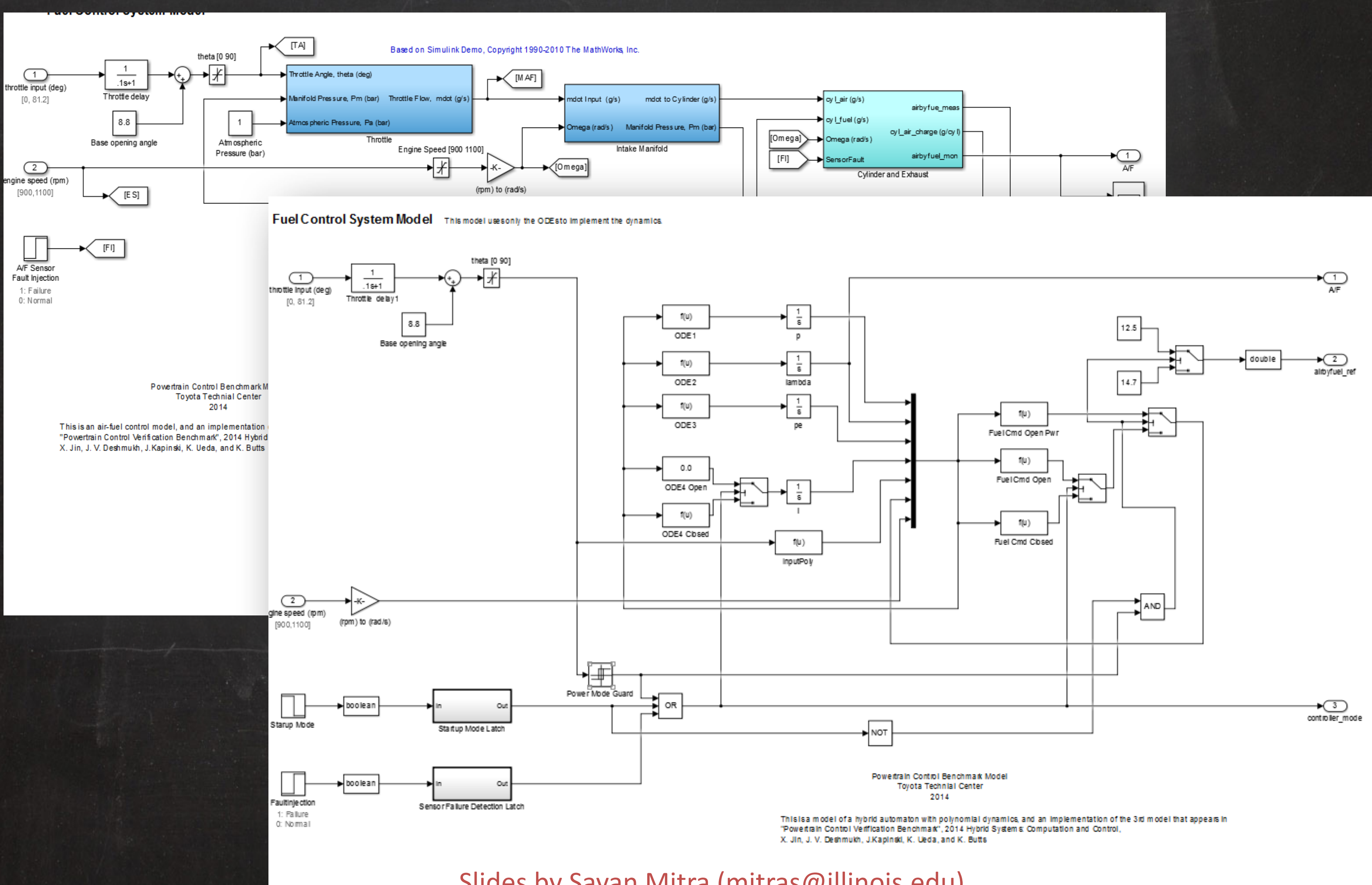
Simulink model from **[Jin et al. HSCC 2014]**

Highly nonlinear polynomial differential equations; **discrete mode switches**

C2E2 **first to verify properties**, e.g., that the **air-fuel ratio** remains within a given range for a set of driver

[CAV 15] Duggirala, Fan, Mitra, Viswanathan: Meeting a Powertrain Verification Challenge.

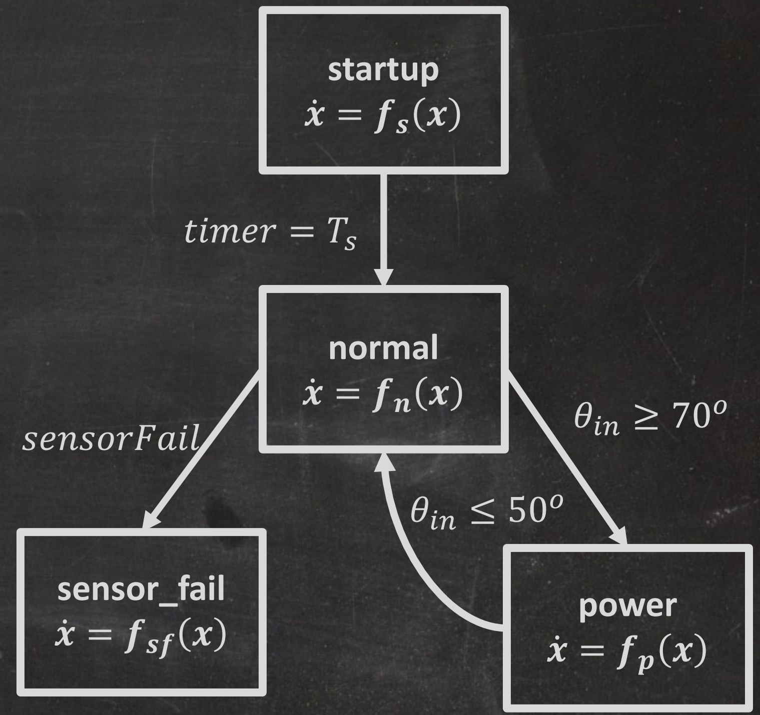
Benchmark Simulink models



Slides by Sayan Mitra (mitras@illinois.edu)

Polynomial hybrid automaton

Variable	Description
θ_{in}	Throttle angle
p	Intake manifold pressure
λ	Air/Fuel ratio
p_e	Intake manifold pressure estimate
i	Integrator state, control variable



$$\dot{\theta} = 10(\theta_{in} - \theta)$$

$$\dot{p} = c_1(2\theta(c_{20}p^2 + c_{21}p + c_{22}) - c_{12}(c_2 + c_3\omega p + c_4\omega p^2 + c_5\omega p^2))$$

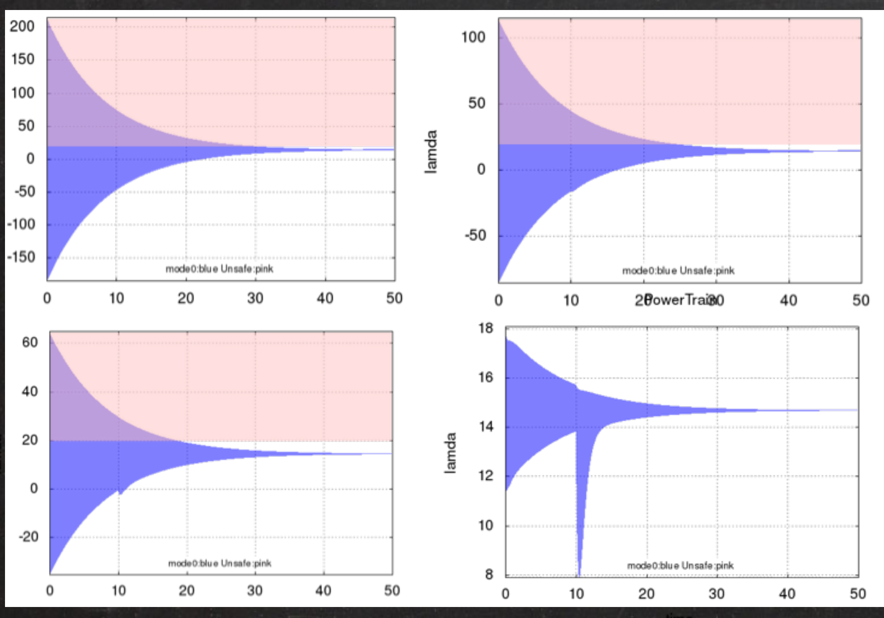
$$\dot{\lambda} = c_{26}(c_{15} + c_{16}c_{25}F_c + c_{17}c_{25}^2F_c^2 + c_{18}\dot{m}_c + c_{19}\dot{m}_c c_{25}F_c - \lambda)$$

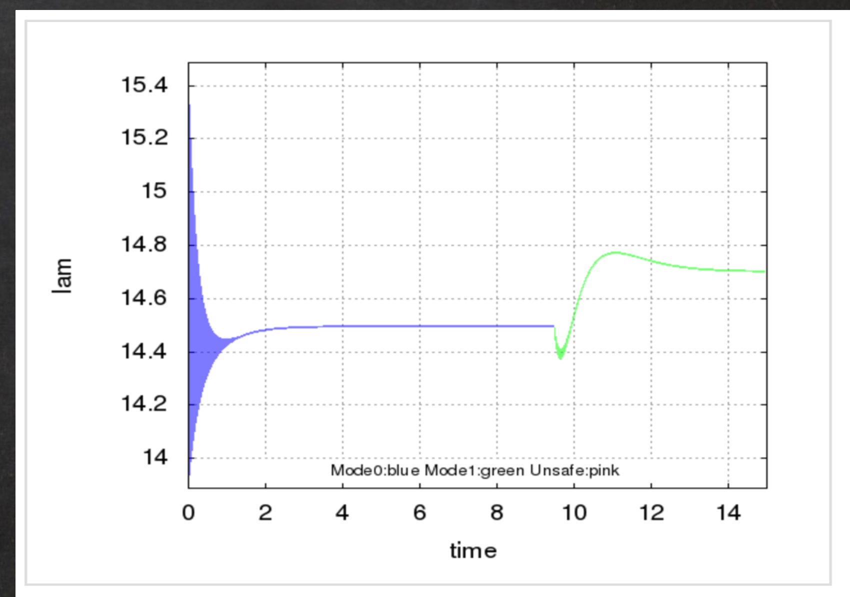
$$\dot{p_e} = c_1\left(2c_{23}\theta(c_{20}p^2 + c_{21}p + c_{22}) - (c_2 + c_3\omega p + c_4\omega p^2 + c_5\omega p^2)\right)$$

$$\dot{i} = c_{14}(c_{24}\lambda - c_{11})$$

Refinements in action: air-fuel ratio range

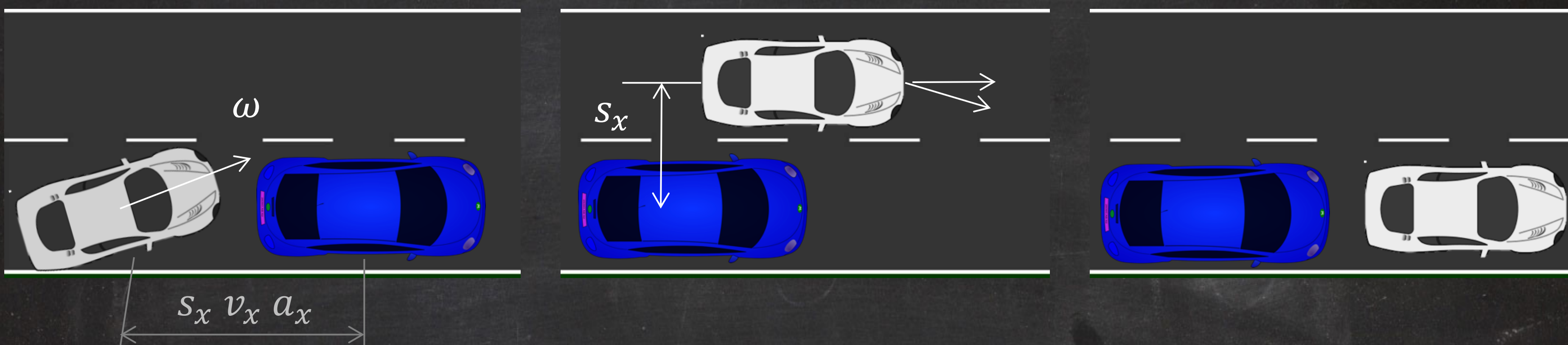
Requirement: Air-Fuel ratio λ contained in interval $[0.9\lambda_{ref}, 1.02\lambda_{ref}]$ for different initial conditions &throttle inputs

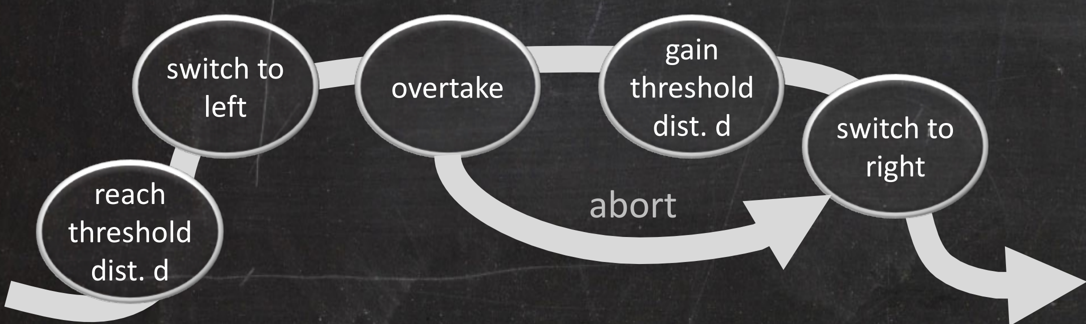




Slides by Sayan Mitra (mitras@illinois.edu)

An auto-pass controller



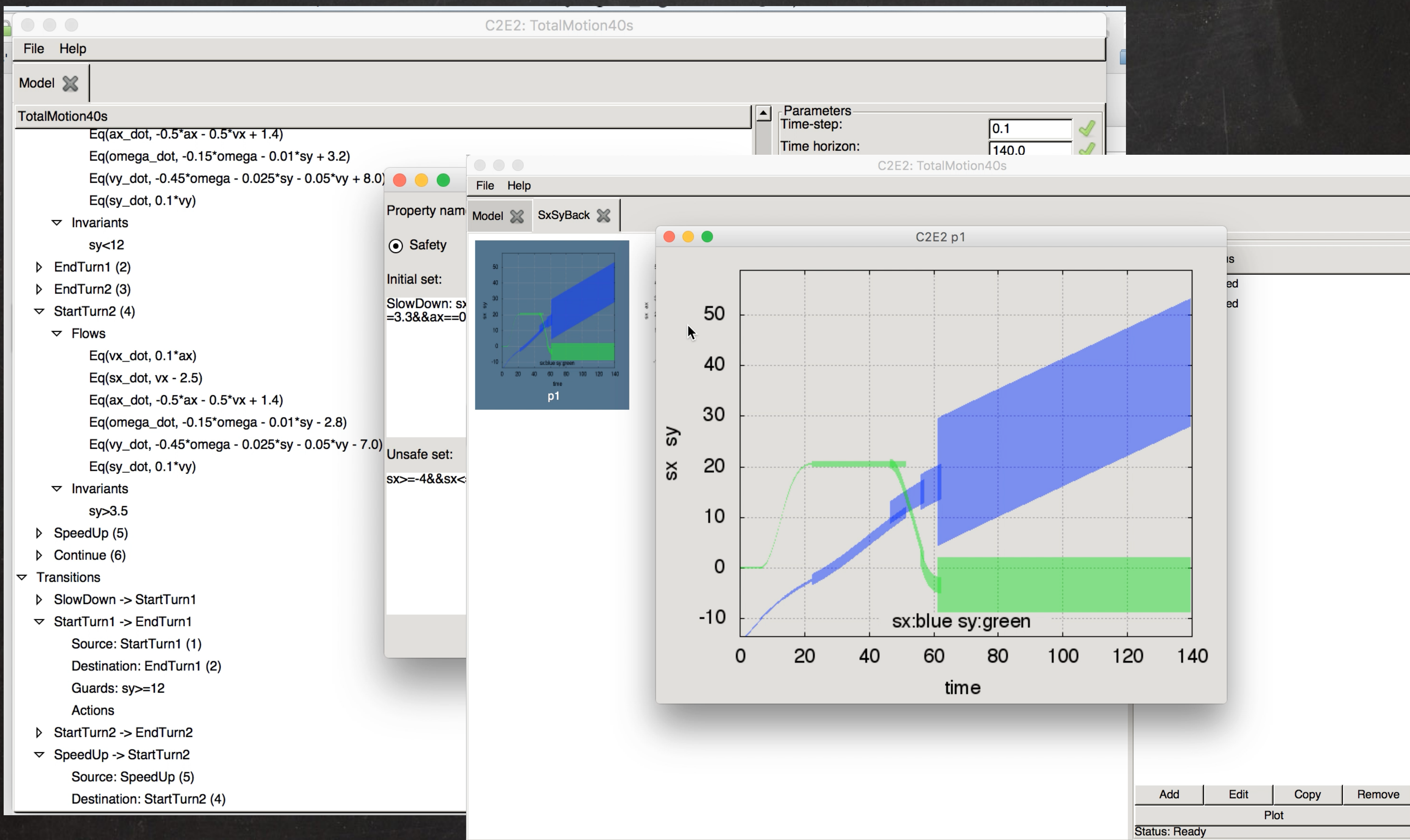


Given a controller and a safe separation requirement, we would like to check that the system is safe with respect to

a) range of initial relative positions
b) range of possible speeds
c) range road friction conditions
d) possible behaviors of “other” car
e) range of design parameters

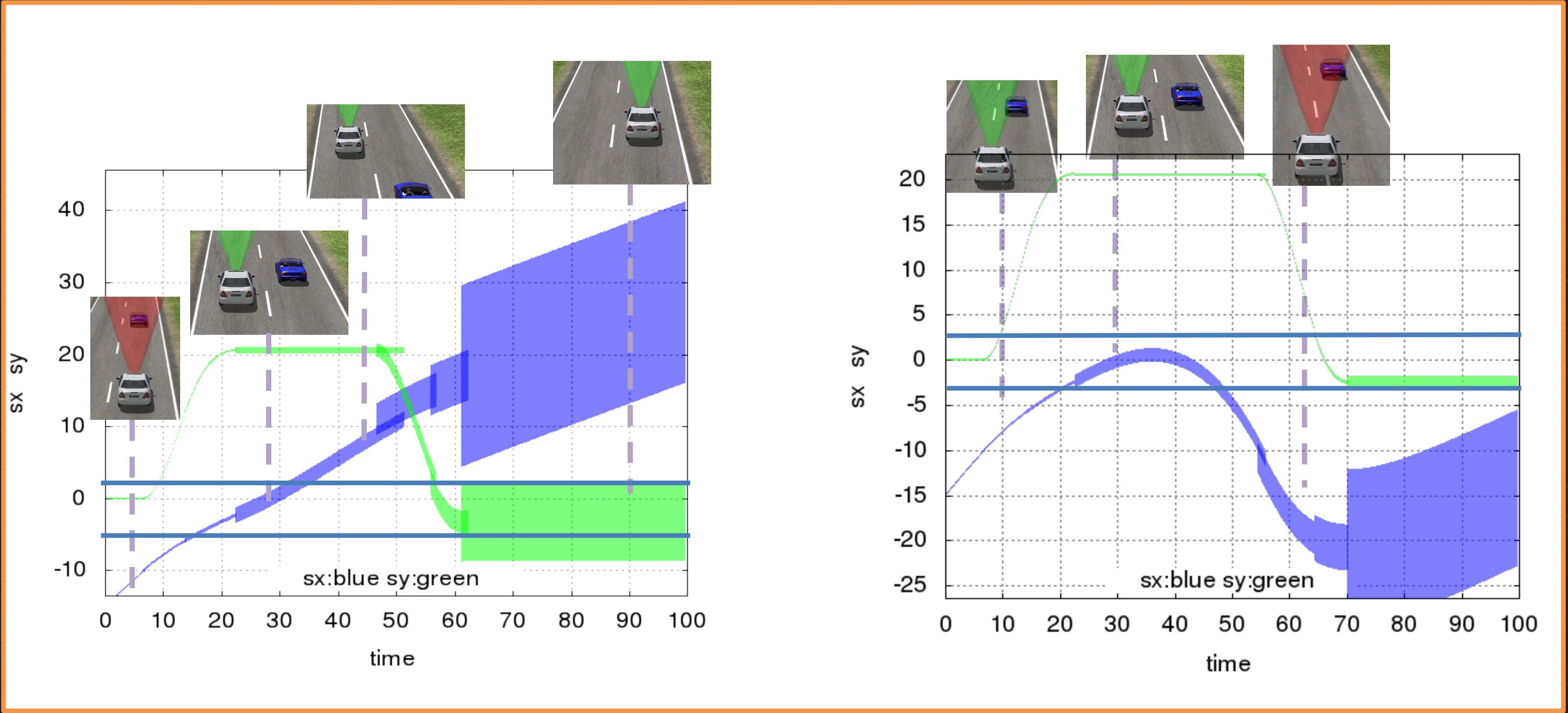
Slides by Sayan Mitra (mitras@illinois.edu)

C2E2: Tool for nonlinear hybrid system verification

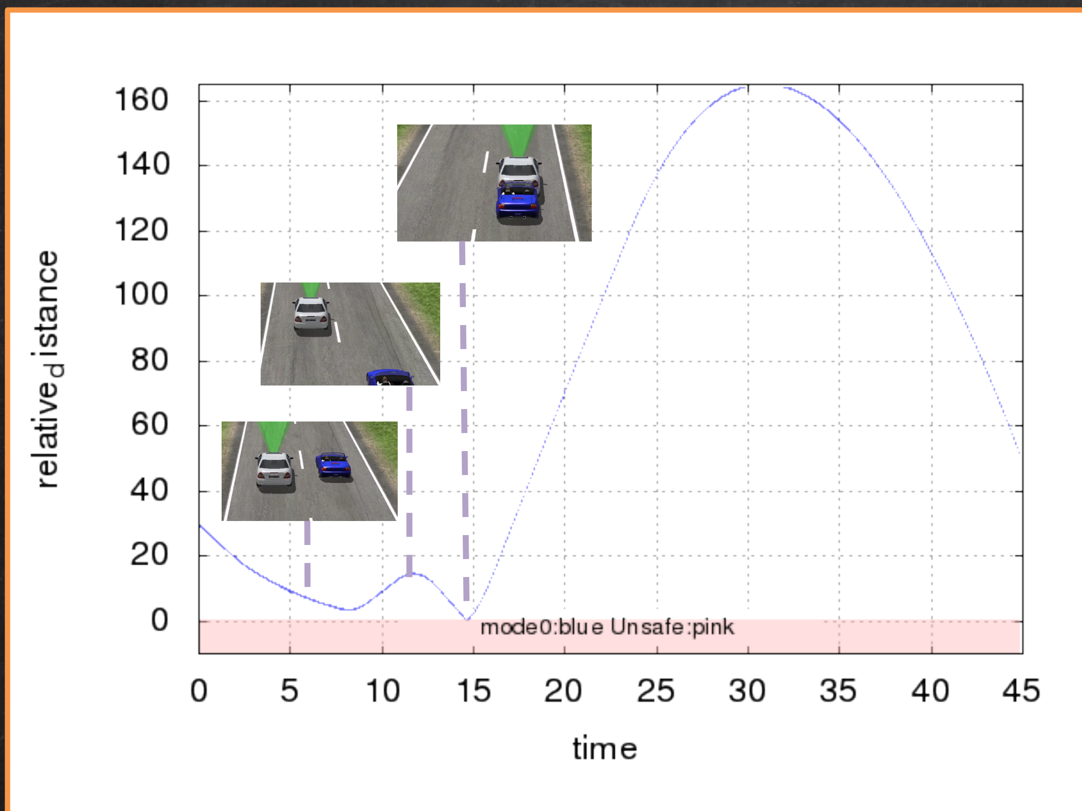


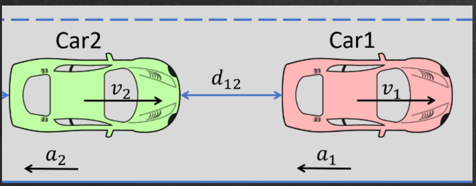
Slides by Sayan Mitra (mitras@illinois.edu)

An auto-pass controller



Debugging systems with high-fidelity models





Homework problem

Initial Set

Time Bound: 10s

Unsafe Set

Mode:Const_Const

Flow:

$$\dot{s}_1 = v_1$$
$$\dot{v}_1 = 0$$
$$\dot{s}_2 = v_2$$
$$\dot{v}_2 = 0$$
$$\dot{t} = 1$$

Inv: $t \leq 1$

Guard: $t \geq c_1$

Mode:Brake_Const

Flow:

$$\dot{s}_1 = v_1$$
$$\dot{v}_1 = -2v_1$$
$$\dot{s}_2 = v_2$$
$$\dot{v}_2 = 0$$
$$\dot{t} = 1$$

Inv: $s_1 - s_2 \geq 10$

Guard: $s_1 - s_2 \leq c_3$

Reset: $t = 0$

Mode:Brake_Const

Flow:

$$\dot{s}_1 = v_1$$
$$\dot{v}_1 = -2v_1$$
$$\dot{s}_2 = v_2$$
$$\dot{v}_2 = 0$$
$$\dot{t} = 1$$

Inv: $t \leq 0.4$

Reaction time

Guard: $t \geq c_2$

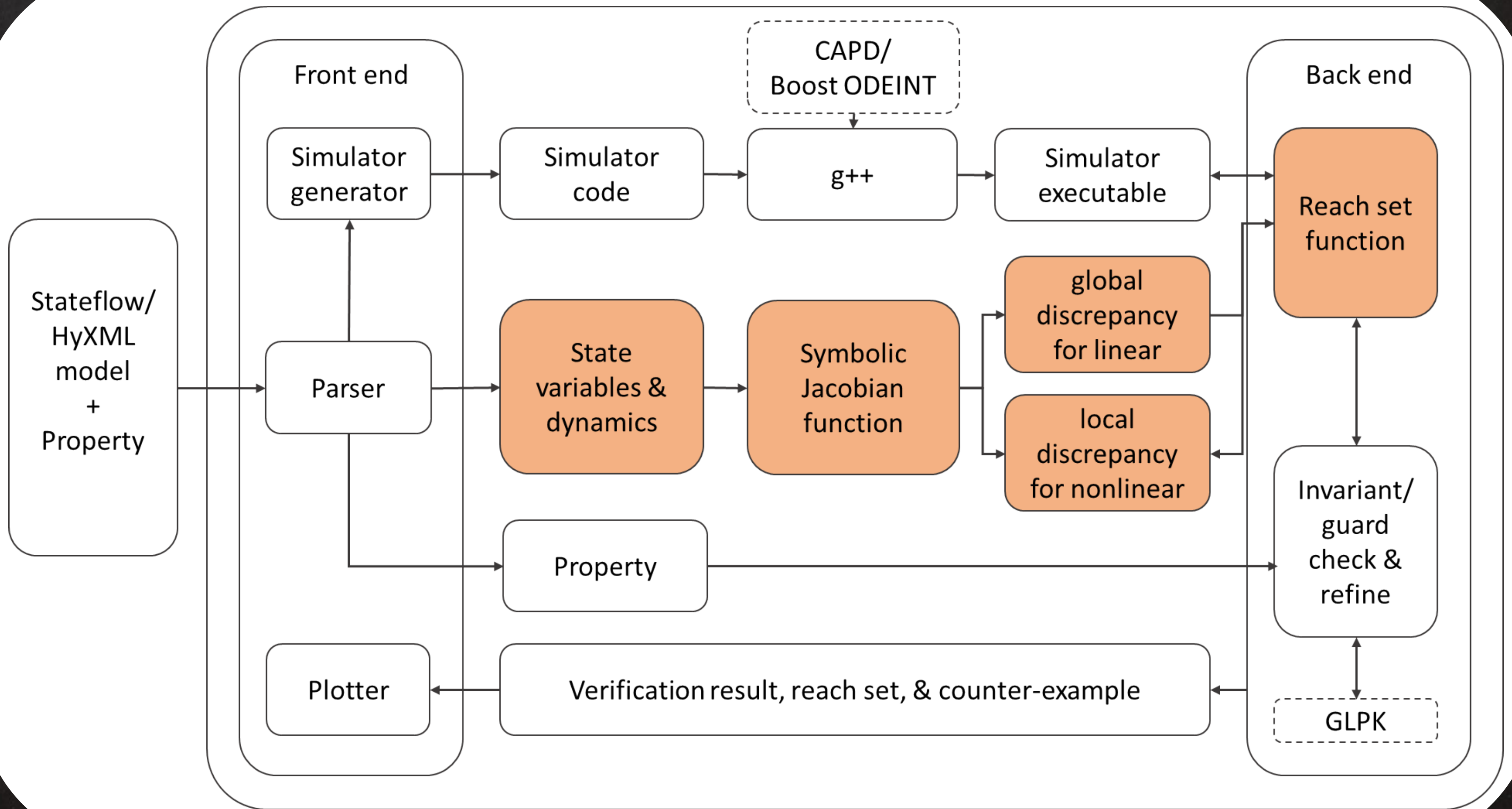
Mode:Brake_Brake

Flow:

$$\dot{s}_1 = v_1$$
$$\dot{v}_1 = -2v_1$$
$$\dot{s}_2 = v_2$$
$$\dot{v}_2 = -3v_2$$
$$\dot{t} = 1$$

Inv: $s_1 - s_2 \geq 0$

C2E2 Architecture



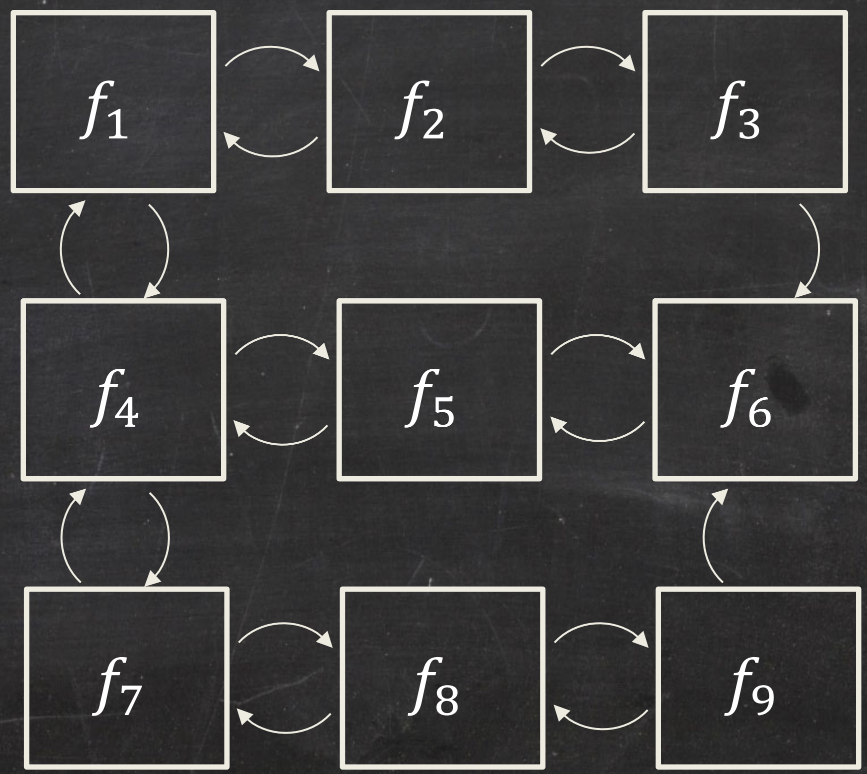
More features

- Log file to debug
- Plotted pictures are saved in the work-dir folder
- Command line version

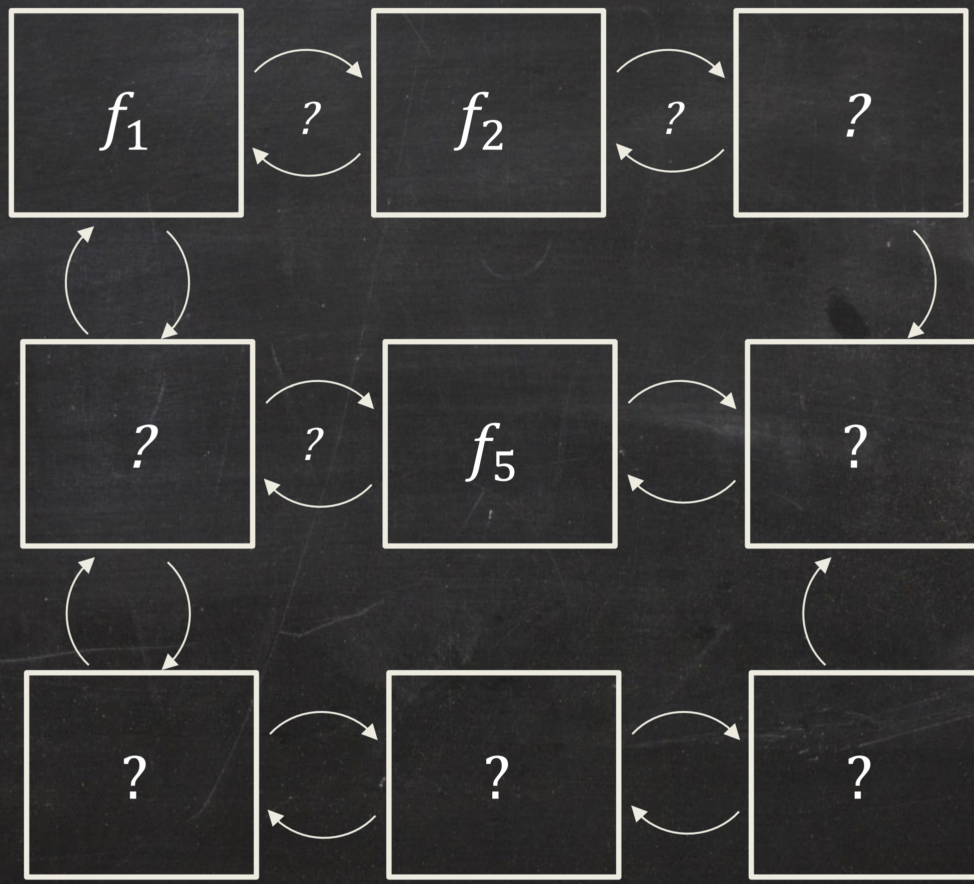
What we don't know

- Sample efficiency of the algorithms
  - Towards that [Girard Pappas 2006]
  - **[Fan et al. EmSoft 2016] [Liberzon Mitra 2016]**
- Unbounded initial set and time horizon
- How to verify open models?
  - $\dot{x}(t) = f\big(x(t), u(t)\big),\ x_0 \in \Theta\ u \in \mathcal{U}$
  - Ongoing work with $\mathcal{U} = \{u_1, .., u_k\}$
- More general models with uncertainty

Hybrid models



Models closer to reality



Slides by Sayan Mitra (mitras@illinois.edu)

“All models are wrong, some are useful”

DryVR

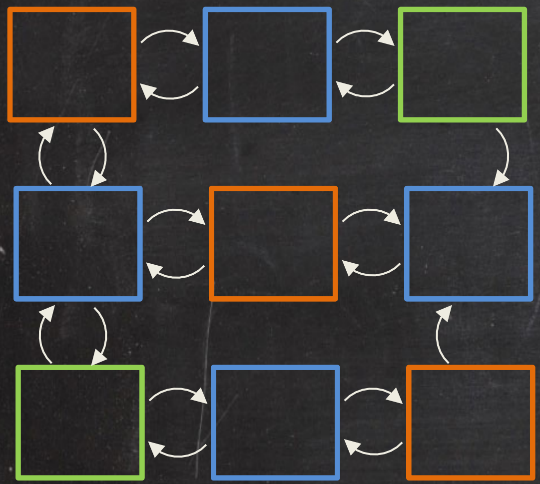
Gain serenity to accept models as they are

https://github.com/qibolun/DryVR

Slides by Sayan Mitra (mitras@illinois.edu)

A new view of knowledge in hybrid models

Complete information of switching structure



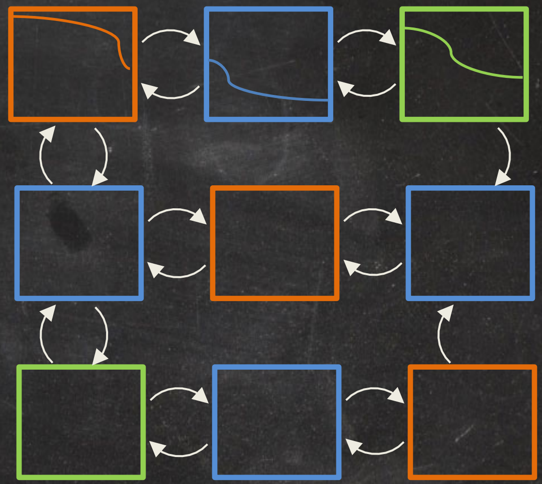
Transitions are time-triggered, possibly nondeterministic: one-clock timed automaton

Executable access to mode dynamics

+

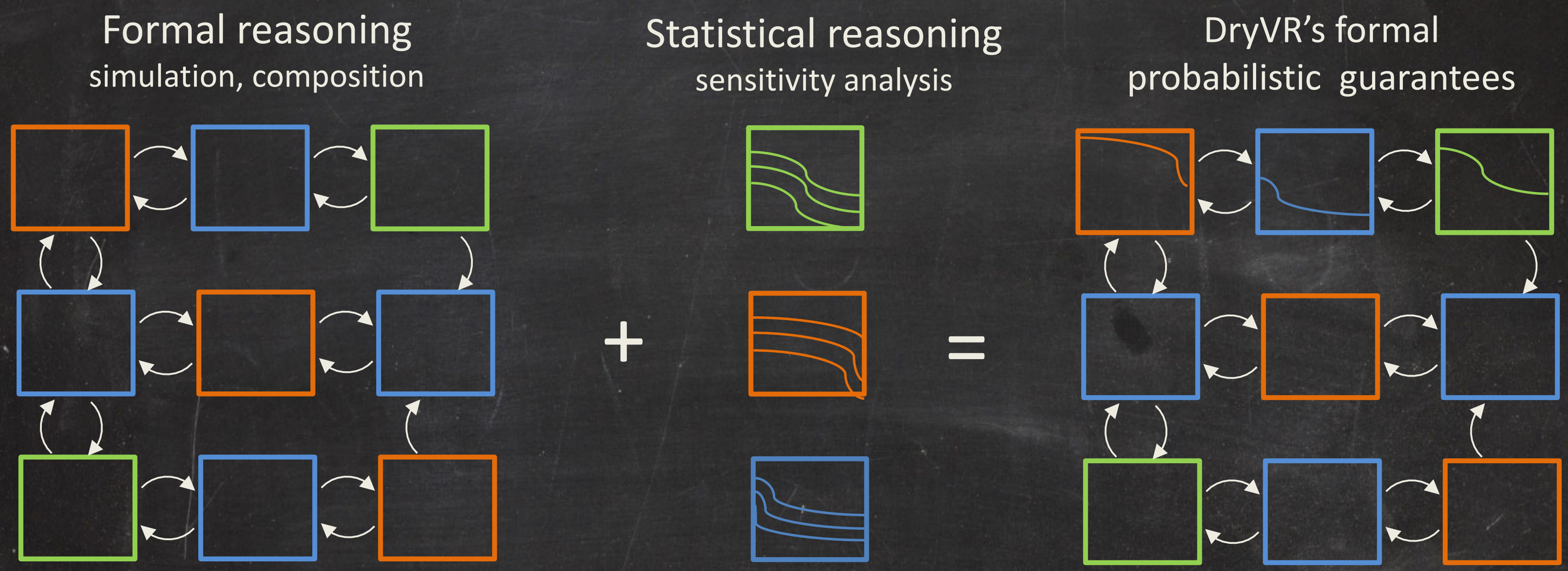
=

DryVR's Executable hybrid model



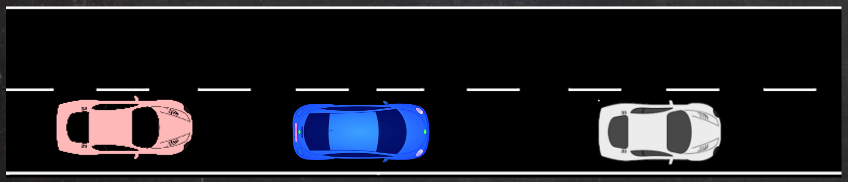
Slides by Sayan Mitra (mitras@illinois.edu)

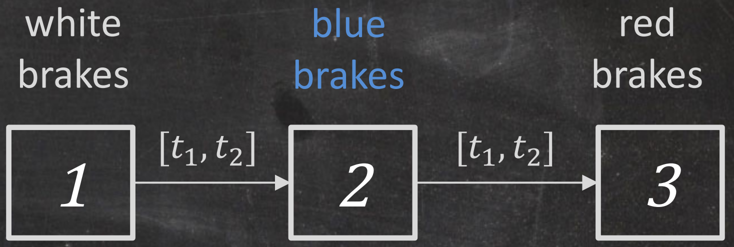
A new view of knowledge in hybrid models



Slides by Sayan Mitra (mitras@illinois.edu)

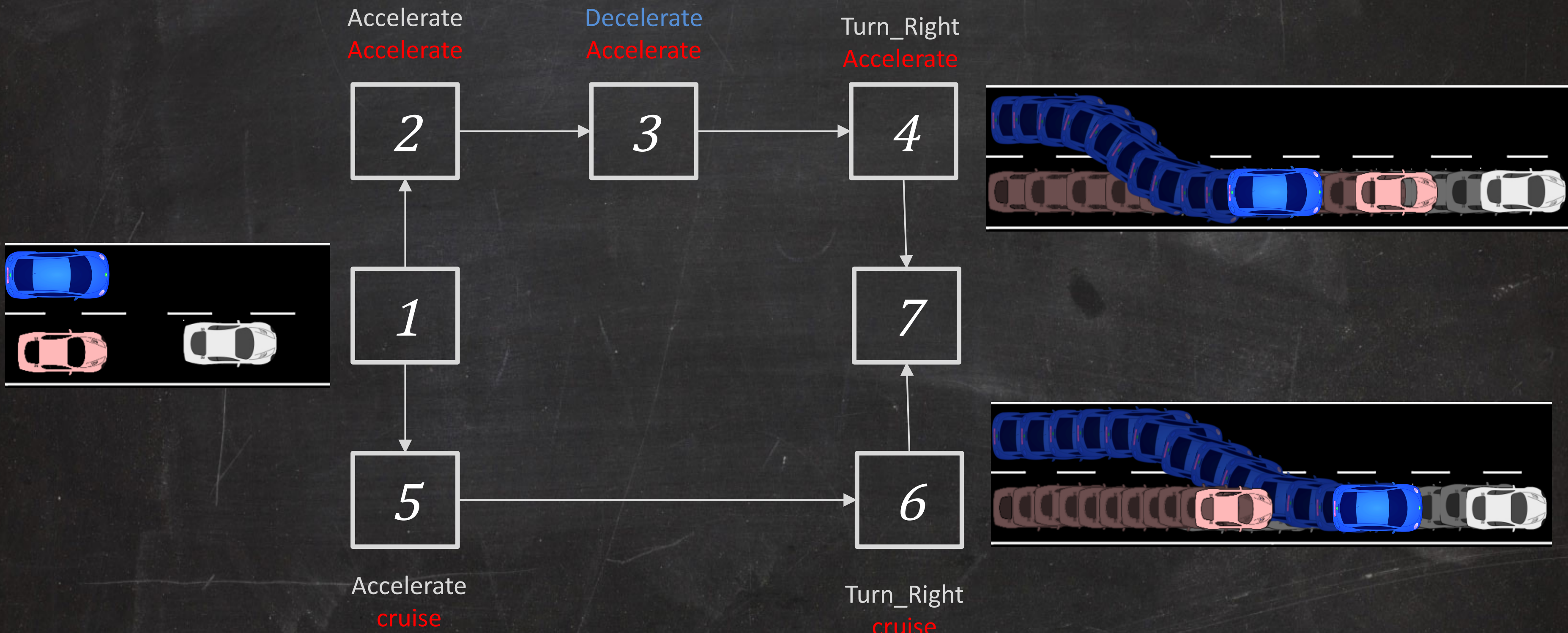
DryVR model for Automatic Emergency Breaking





Slides by Sayan Mitra (mitras@illinois.edu)

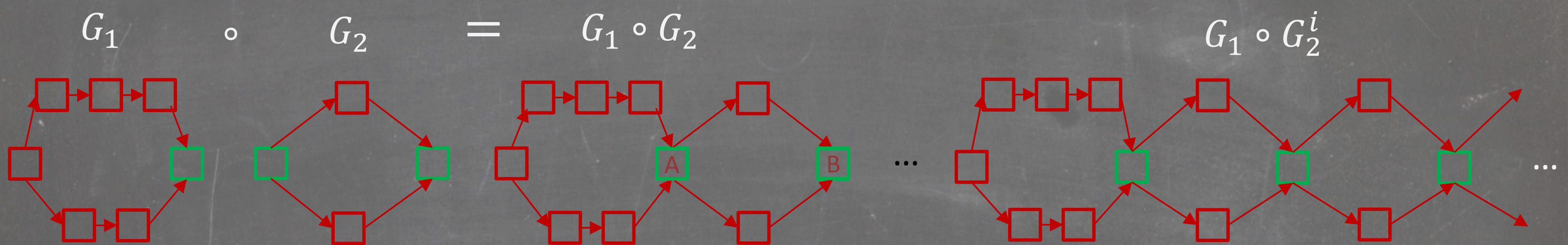
DryVR model for auto-pass



Slides by Sayan Mitra (mitras@illinois.edu)

Composition for unbounded time analysis

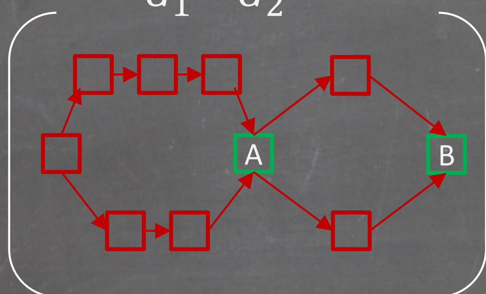
If $\boldsymbol{Reach|B} \subseteq \boldsymbol{Reach|A}$ then

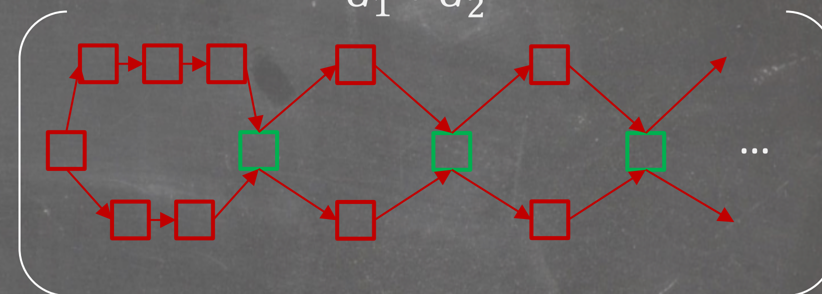


Composition for unbounded time analysis

If $Reach|B \subseteq Reach|A$ then

$G_1 \circ G_2$ $\qquad\qquad$ $G_1 \circ G_2^i$

$Reach$ 

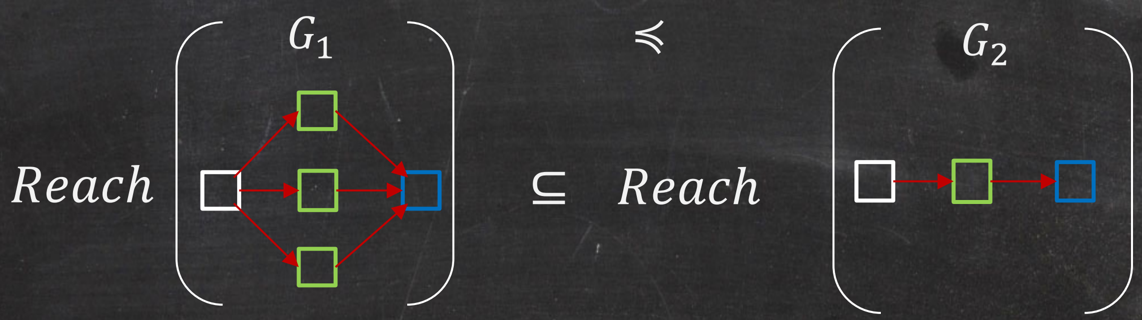
$\supseteq$ $Reach$ 

Reasoning about behavior containment

Trace containment $G_1 \preccurlyeq G_2$

Trajectory containment $\mathcal{TL}_1 \preccurlyeq \mathcal{TL}_2$

If $\Theta_1 \subseteq \Theta_2, G_1 \preccurlyeq G_2, \mathcal{TL}_1 \preccurlyeq \mathcal{TL}_2$, then



Learning discrepancy from black-box

Assume a form of the discrepancy

Global exponential discrepancy

$$\beta(x_1, x_2, t) = |x_1 - x_2| K e^{\gamma t}$$

Others piece-wise exponential, polynomial

For any pair of trajectories τ_1 and τ_2 in mode □

$\forall t \in [0, T], |\tau_1(t) - \tau_2(t)|$
$\leq |\tau_1(0) - \tau_2(0)| K e^{\gamma t}$

$$\forall t, \ln \frac{|\tau_1(t) - \tau_2(t)|}{|\tau_1(0) - \tau_2(0)|} \leq \gamma t + \ln K$$

Familiar problem of learning linear separators

Learning linear separators

For a subset $\Gamma \subseteq \mathbb{R}\times\mathbb{R}$, a linear separator is a pair $(a, b) \in \mathbb{R}^2$ such that $\forall(x, y) \in \Gamma, x \leq ay + b$

Algorithm:

1. Draw k pairs $(x_1, y_1), \dots, (x_k, y_k)$ from Γ according to $\mathcal{D}$.

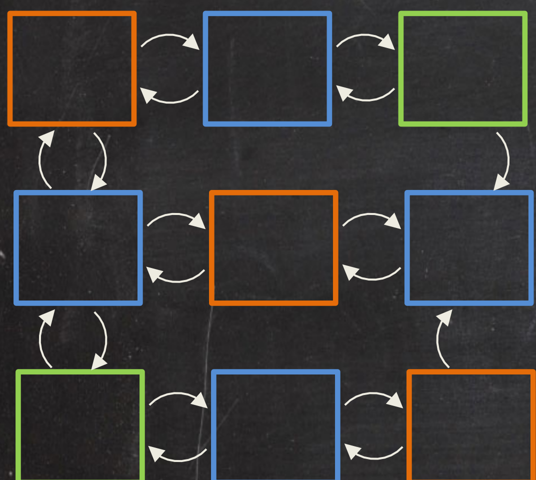
2. Find $(a, b) \in \mathbb{R}^2$ s.t. $x_i \leq ay_i + b$ for all $i \in \{1, \dots, k\}$.

Proposition [Valiant 84]: Let $\epsilon, \delta \in \mathbb{R}^+$. If $k \geq \frac{1}{\epsilon} \ln \frac{1}{\delta}$ then with probability $1 - \delta$, the above algorithm finds (a, b) such that $err_{\mathcal{D}}(a, b) = \mathcal{D}(\{(x, y) \in \Gamma \mid x > ay + b\}) < \epsilon$.

Experience: 96% accuracy for 10 trajectories, >99.9% for 20

DryVR

Complete information of switching structure



Model file specifies vertices, edges, labels

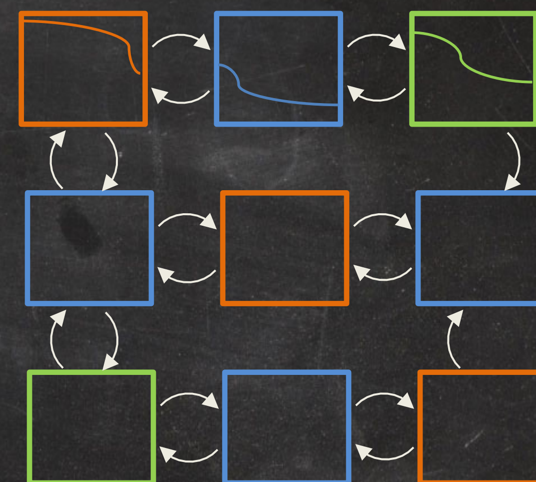
+

Executable access to mode dynamics

Simulate function takes as input mode, initial state, and time horizon

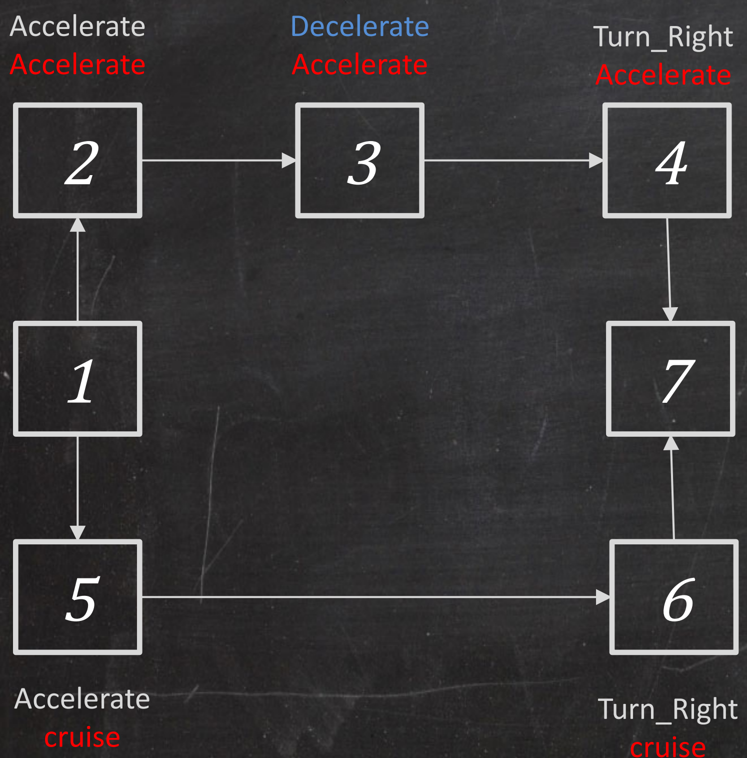
=

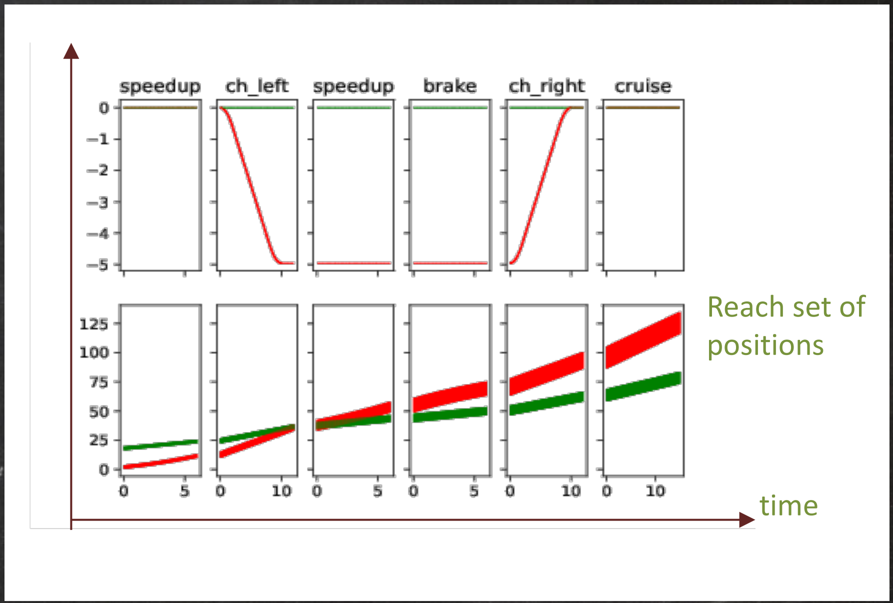
DryVR's Executable hybrid model



Slides by Sayan Mitra (mitras@illinois.edu)

Reachability analysis

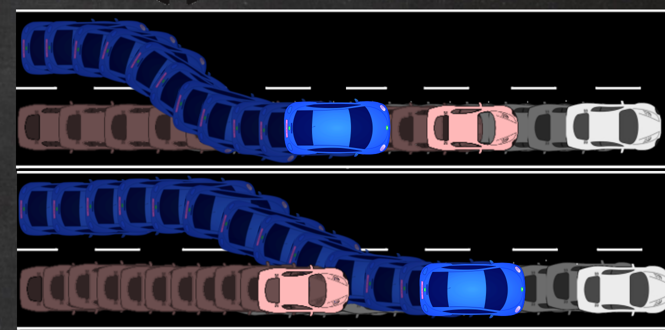


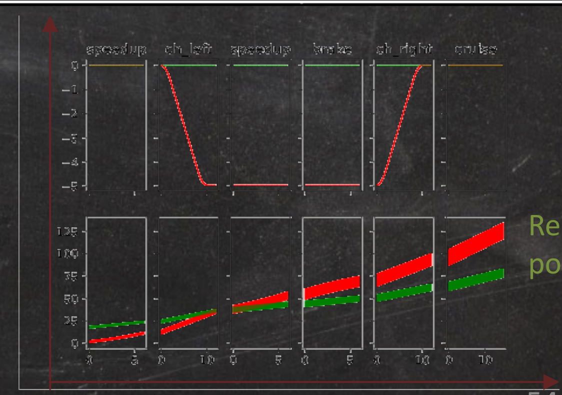


Automotive maneuvers

Model	Time horizon	Unsafe set	# Refinement	Safe	Run time
Auto-passing	50	Collision	4	✔	208s
	50	Collision	5	✘	152s
Lane-merge	50	Collision	0	✔	55s
	50	Collision	0	✘	38s
Lane-merge-highway	50	Collision	4	✔	197s
	50	Collision	0	✘	21s
Powertrain	80	Air/Fuel out of bound	2	✔	217s
Automatic transmission	50	Engine speed too high	2	✔	109s

https://github.com/qibolun/DryVR

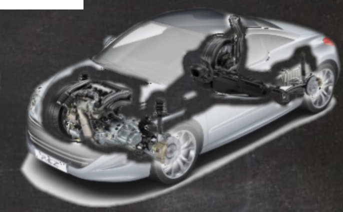


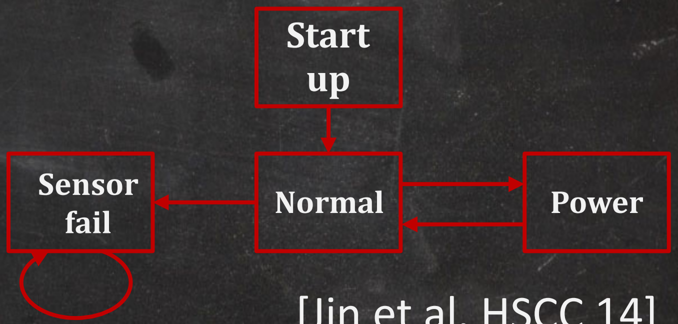


Case studies: Engine control

Model	Time horizon	Unsafe set	# Refinement	Safe	Run time
Auto-passing	50	Collision	4	✔	208s
	50	Collision	5	✘	152s
Lane-merge	50	Collision	0	✔	55s
	50	Collision	0	✘	38s
Lane-merge-highway	50	Collision	4	✔	197s
	50	Collision	0	✘	21s
Powertrain	80	Air/Fuel out of bound	2	✔	217s
Automatic transmission	50	Engine speed too high	2	✔	109s

https://github.com/qibolun/DryVR



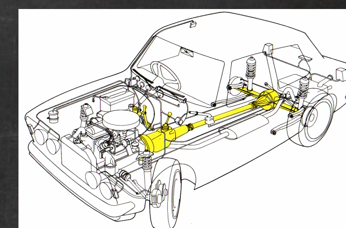


[Jin et al. HSCC 14]

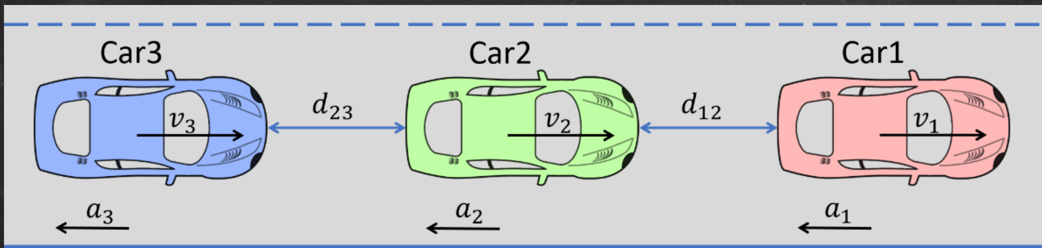
Case studies: transmission control

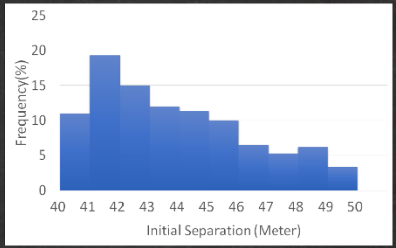
Model	Time horizon	Unsafe set	# Refinement	Safe	Run time
Auto-passing	50	Collision	4	✔	208s
	50	Collision	5	✘	152s
Lane-merge	50	Collision	0	✔	55s
	50	Collision	0	✘	38s
Lane-merge-highway	50	Collision	4	✔	197s
	50	Collision	0	✘	21s
Powertrain	80	Air/Fuel out of bound	2	✔	217s
Automatic transmission	50	Engine speed too high	2	✔	109s

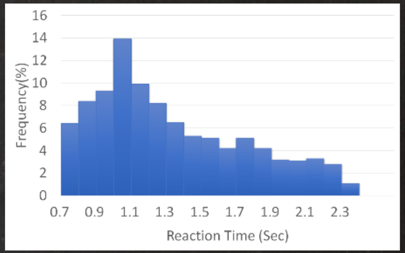
https://github.com/qibolun/DryVR



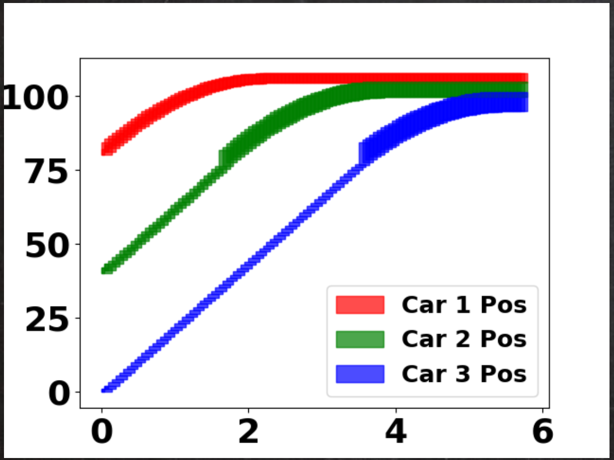
Automated Risk / ASIL Analysis

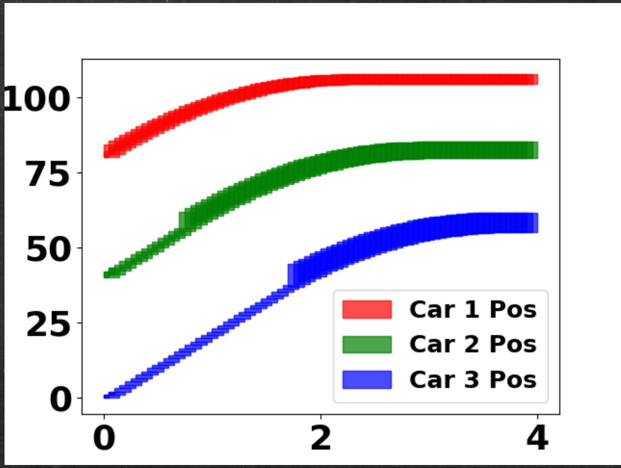


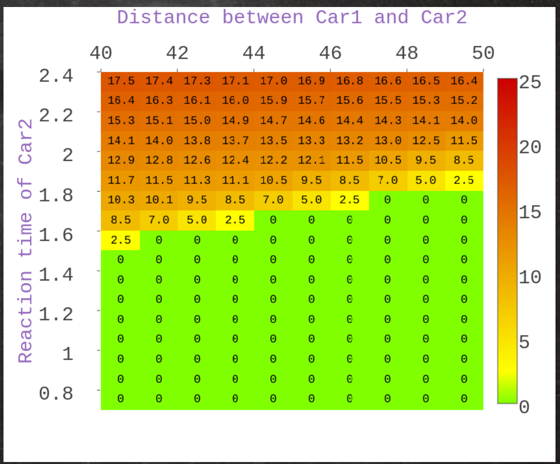


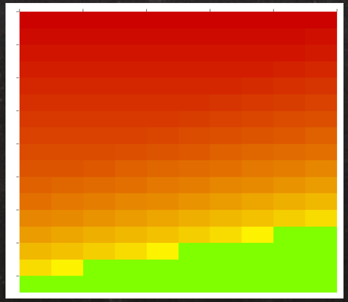


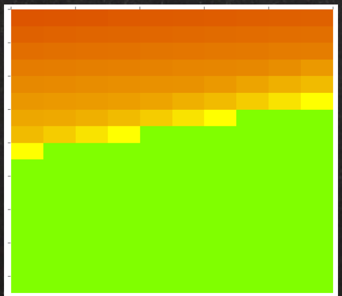
Risk = Probability x Severity

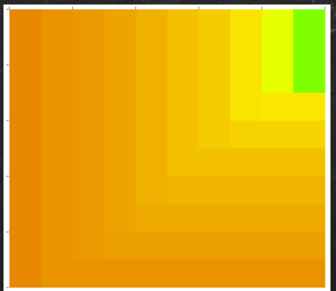


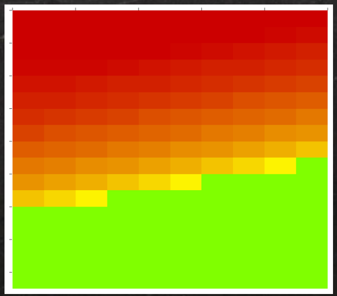












Conclusion

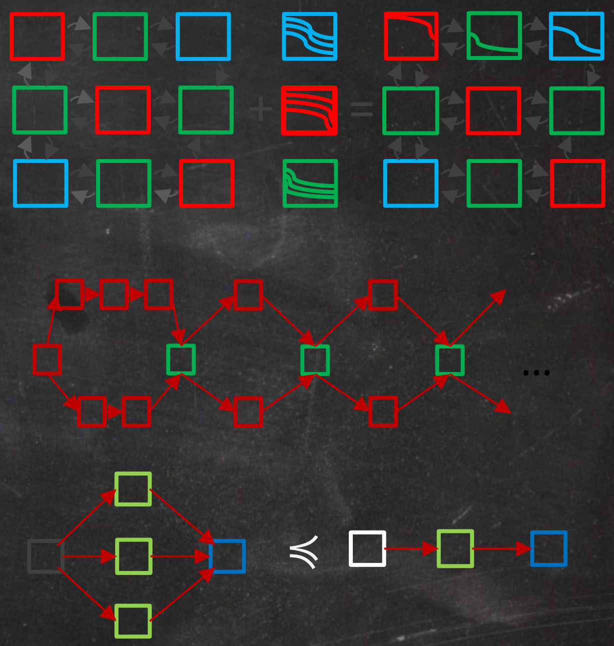
A fresh perspective (DryVR's model) on modeling hybrid systems

- white box transition graph + black box simulator
- Case studies ADAS / AV

Enables types of static-dynamic analysis

- Black-box => discrepancy functions with probabilistic guarantees
- Bounded verification [Sound and relatively complete]
- Proof rules for sequential composition for unbounded time verification and behavior containment

Future: More expressive white boxes, synthesis, monitoring,



Conclusions

Simulation data + sensitivity from models => algorithms => sound & complete invariance verification

Try C2E2 and DryVR give feedback, built on!

Examples available: Satellites to transistors

Several open questions about handling models with uncertainty and precise characterization of efficiency

This work is supported by grants form the United States National Science Foundation (NSF)