Model-based analysis and synthesis for security of cyber-physical systems

Sayan Mitra, Geir Dullerud & Swarat Chaudhuri
University of Illinois at Urbana Champaign

NSA SOS Review meeting, October 2015

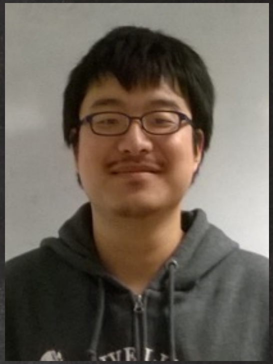
project team

Sayan Mitra
UIUC, ECE
Hybrid & Distributed systems

Geir Dullerud
UIUC, MechE
Control theory, hybrid systems

Swarat Chaudhuri
Rice University, CS
Programming Languages, Formal methods

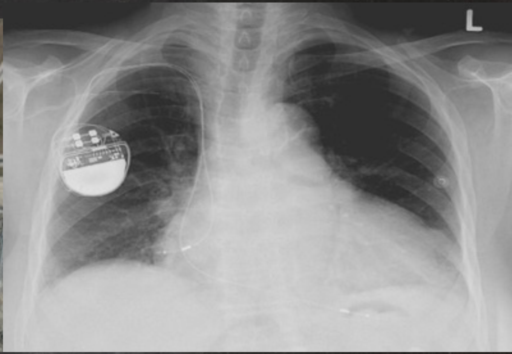
Zhenqi Huang
PhD student, ECE



Yu Wang
PhD student, MechE

cyber-physical systems





- engineering systems that bring together sensing, computation, and control
- autonomous, complex, and safety-critical
- many application areas: driving assist systems, driverless cars, embedded medical devices, surveillance drones

'I COULDN'T STOP IT'

Drone operator speaks out

STORY: PAGE 3

Crash involving self-driving Google car injures three employees

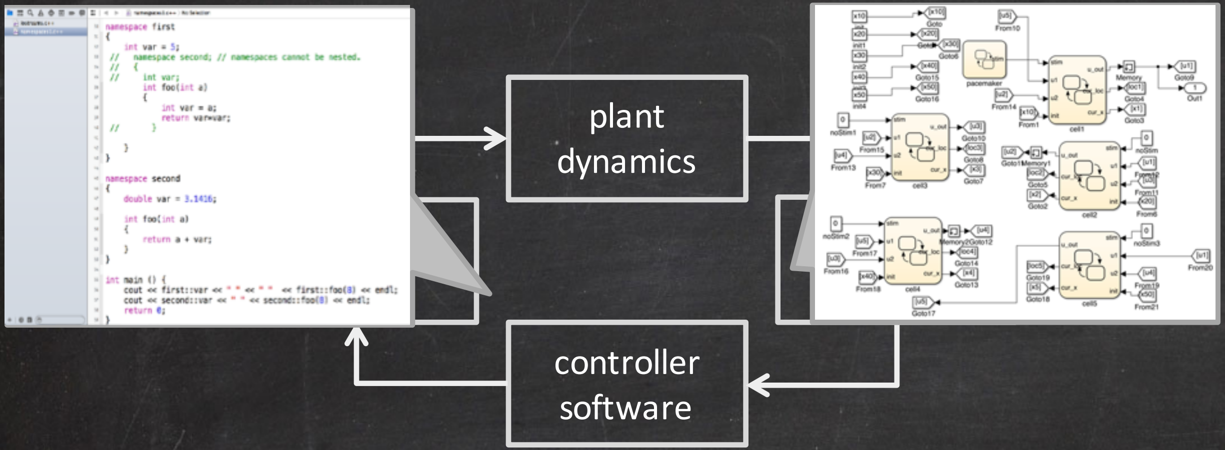
Driverless car hit while stationary in traffic by human driver travelling at 17mph in another vehicle, resulting in the first self-driving car injuries

"How can we design cyber-physical systems people can bet their lives on?" --- Jeannette Wing

foundational approach

- develop sound and relative complete algorithms for analysis and synthesis
  - powertrain control in vehicles
  - motion control in drones
- theory for optimality in distributed control while preserving privacy
  - distributed optimization
  - traffic networks
- robust control, formal methods, program analysis, and distributed systems theory

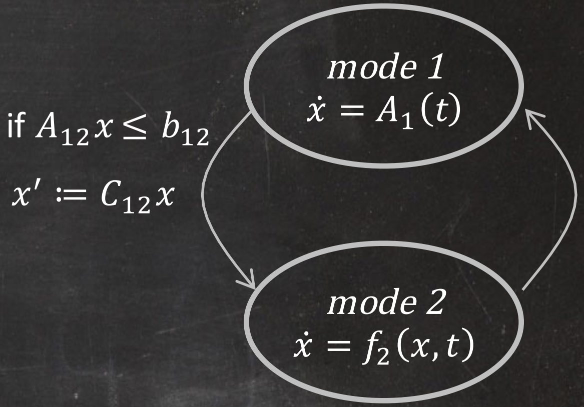
system design & properties



plant dynamics

controller software

```
namespace first
{
    int var = 5;
 //   namespace second; // namespaces cannot be nested.
 //   {
 //     int var;
        int foo(int a)
        {
            int var = a;
            return var*var;
 //       }

    }
}

namespace second
{
    double var = 3.1416;

    int foo(int a)
    {
        return a + var;
    }
}

int main () {
    cout << first::var << " " << " "  << first::foo(8) << endl;
    cout << second::var << " " << second::foo(8) << endl;
    return 0;
}
```



mode 1
$\dot{x} = A_1(t)$

if $A_{12}x \leq b_{12}$

$x' \coloneqq C_{12}x$

mode 2
$\dot{x} = f_2(x, t)$

hybrid systems models: mathematical model of CPS

differential equations & programs

discrete or continuous time

uncertainties: model parameters, disturbances, scheduling

- invariance and safety: “drone maintains safe separation to objects”
- stability, disturbance attenuation: “under sensor failures/attacks, air-fuel ratio maintained in required range”
- sensitivity: “individuals in a distributed control system maintain differential privacy ?”
- controllability: “does there exist a path for an attacker to make a power system unstable while avoiding detection ?”

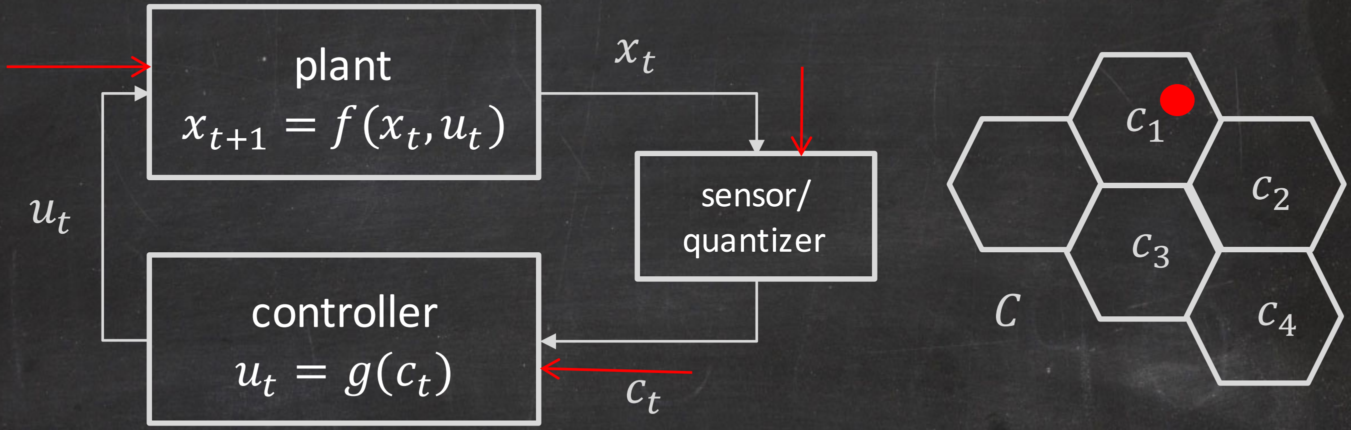
outline

- control synthesis
- privacy in cyber-physical systems
- challenge problems in verification

PART I

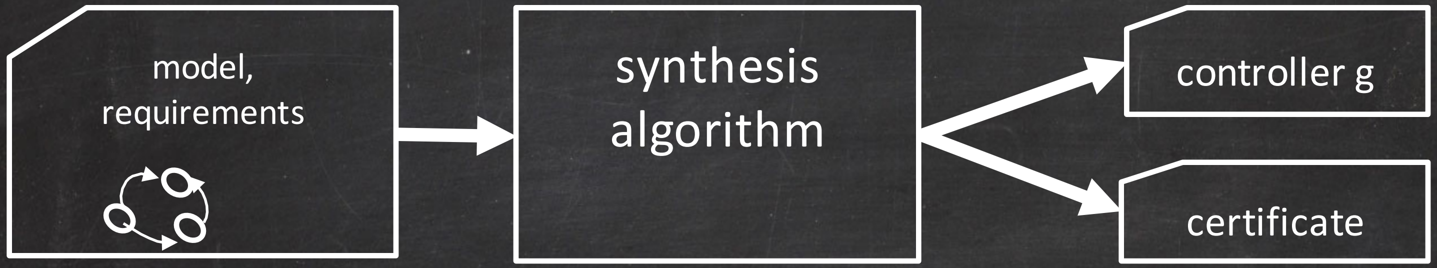
CONTROLLER SYNTHESIS WITH ADVERSARY

control system with quantized sensing



- measurements over finite bandwidth channel: quantized and sampled
- multi-point attack surface
- goal: synthesize controller with provable guarantees (certificates)

synthesis problem as search


model, requirements

synthesis algorithm

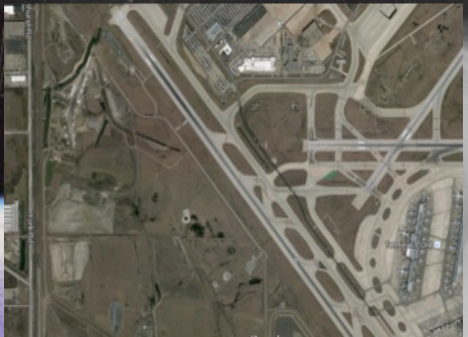
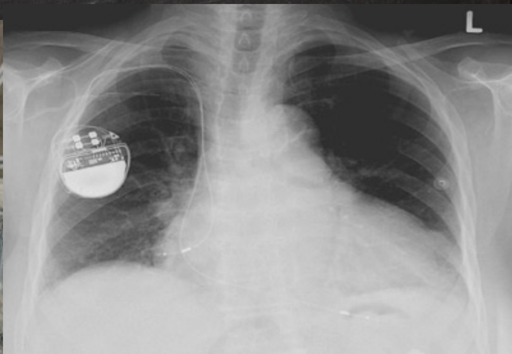
controller g

certificate

given a system *model*, *quantization*, *init, safe* and *goal*, find control *g(.)* such that all behaviors are safe and reach goal

- yes (controller strategy function *g*)
- no (impossibility certificate “no controller exists”)



inductive synthesis rules [Huang et al. CDC 15]

Find $g: \boldsymbol{C} \rightarrow U, \mathrm{V}: \mathrm{C} \rightarrow \mathbb{N}, k \in \mathbb{N}$ such that

- (control invariant)
  $\mathrm{V}(init) \leq k \wedge C' \subseteq post(C, g) \Rightarrow V(C) \geq V(C')$
- (safe) $V(C) \leq k \Rightarrow C \subseteq safe$
- (goal) $C \subseteq goal \Leftrightarrow V(C) = 0;$
- (progress)
  $C \subseteq inv \backslash \mathrm{goal} \wedge C' \subseteq post^{k}(C, g) \Rightarrow V > V(C')$

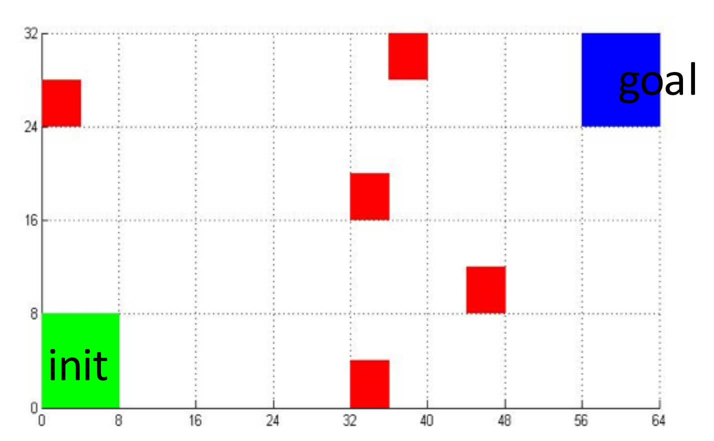
soundness and relative completeness of synthesis algorithm

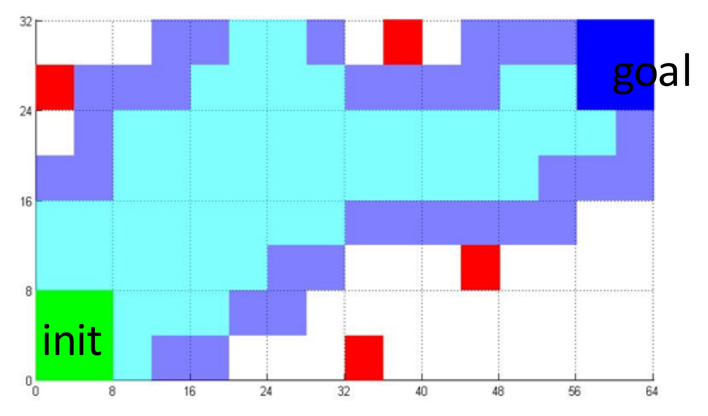
- Robustness: Given controller C and ranking function templates R, the problem M is robust if there exists $\epsilon > 0$ *:*
  - *exists $g \in C, V \in R$ such that for any problem M' that is ϵ-close to M, the g, V solves the synthesis problem for M' with some k, OR*
  - *for none of the problems M' that are ϵ-close to M, have solutions to the synthesis problem with any $g \in C, V \in R$*

- Theorem. If the synthesis problem M is (C,R)-robust, then there exists a sufficiently accurate computation of $\mathrm{post}(\mathrm{C}, \mathrm{g})$ to (a) either find control g and proof V or (b) give a proof that there exists no such controller in C, R.

application: path planning

- implemented using CVC4 SMT solver
- nonlinear vehicle navigation with noise and obstacles
- C: regions in x-y plane
- $V : C \rightarrow \mathbb{N}$
- 768 cells, 3072 real-valued variables, booleans, solved in less than 10 minutes





Light (under) and over (dark) approximation of post

linear dynamics with L2 attack budget

$Reach(x_0, u, t) = \{\ x \mid \exists a : x = \xi(x_0, u, at)\}$

$\mathrm{L}(x_0, u, t)$ is called adversarial leverage iff
$Reach(x_0, u, Adv, t) = Reach(x_0, u, O, t) \oplus L(x_0, u, t)$

For linear dynamics and L2-budget
$\mathrm{L}(x_0, u, t) = \{x \mid x^T W_t^{-1} x \le b\}$,
where $W_t = \sum_{s=0}^{t-1} A^{t-s-1} C C^T (A^T)^{t-s-1}$

Can be computed exactly and independently of x_0

adversarial leverage

For each $t \leq H$, generate $safe_t$ and $goal_t$ such that

- $safe_t \oplus L(t) = safe$
- $goal_t \oplus L(t) = goal$

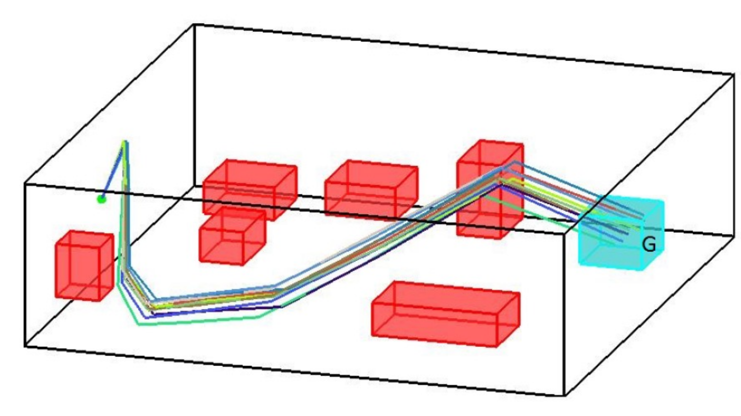
$safe_t, goal_t$ computed by conic programming

Check $\exists u \in Ctrl : \forall t, x_0 \in Init,\ Reach(Init, u, 0, t) \subseteq safe_t$ and $Reach(Init, u, 0, T) \subseteq Goal_T$

Theorem. Exists u that is adversary-free solution u $Reach(x_0, u, 0, t) \in Safe_t$ and $Reach(x_0, u, 0, t) \in Safe_t$ Iff u solves the control synthesis problem with adversary

planning under uncertainty

Autonomous helicopter (16D, 4 inputs)



$x_{t+1} = A_t x_t + B_t u_t + C_t a_t$

Adv: $\sum |a_i|^2 \leq b$: intrusion budget constraints

Ctr: $\sum c_i u_i \leq k$: actuation constraints

$Init$: Additive sensor attacks

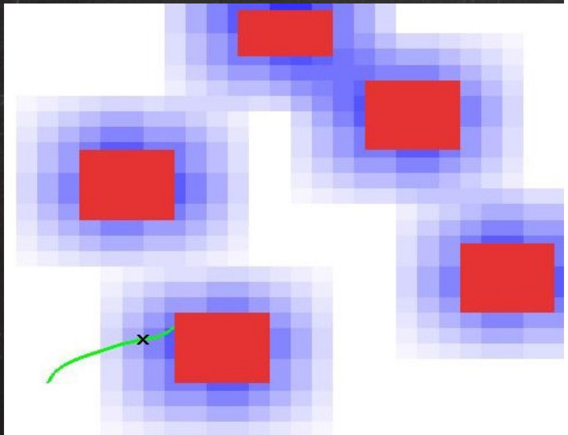
Synthesis of $Adv(b)$-proof control strategies

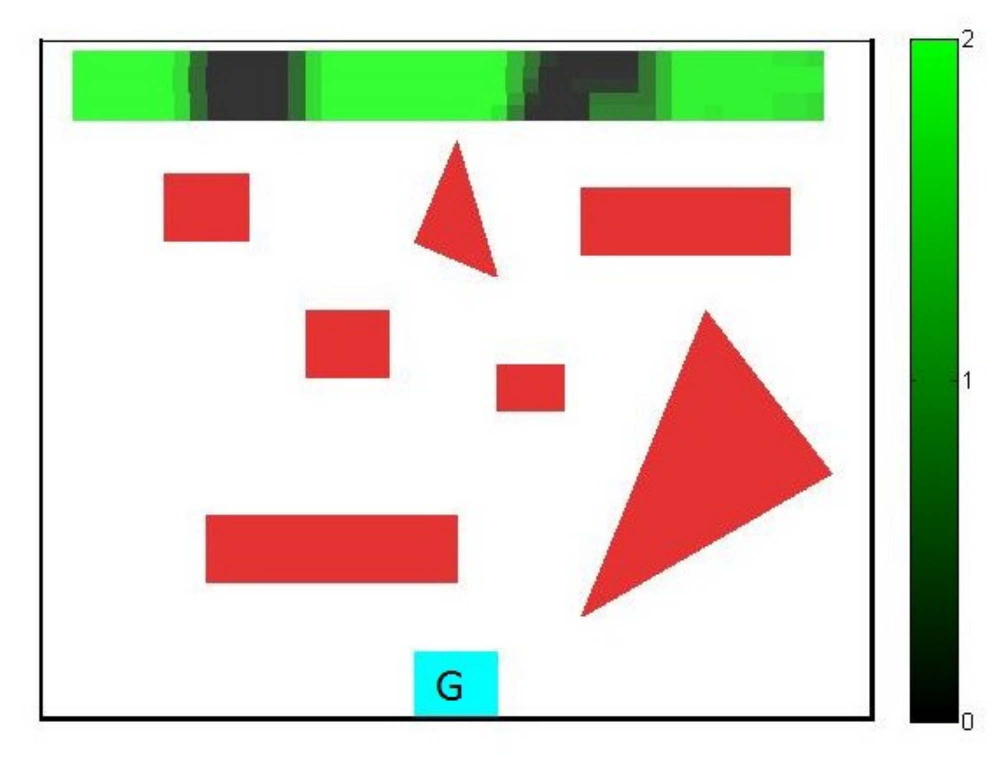
Find b_{crit} that makes synthesis impossible

Vulnerability classification of initial states

Attack synthesis: function: $\mathbb{R}^n \rightarrow Adv$ that reaches **unsafe**

T	ϕ_{safe}	ϕ_{goal}, Ctr	ϕ	Result	R.time (s)
40	16	4, 160	804	Unsat	2.79
80	44	4, 320	3844	Sat	35.22
320	24	4, 1280	8964	Sat	532.5
9	36	6, 72	402	Sat	24.5
12	24	6, 96	338	Sat	60.6
15	24	10, 96	576	Sat	158.8





summary and outlook

- we have developed a new class of synthesis algorithms for control systems under attacks with budget-constrained adversaries
  - algorithms can also give impossibility certificates
  - applications in motion planning under sensor attacks
- ongoing: switching based synthesis of attacks on that make power networks unstable while evading standard detection mechanisms (new collaboration with Prof. Saman Zonouz)

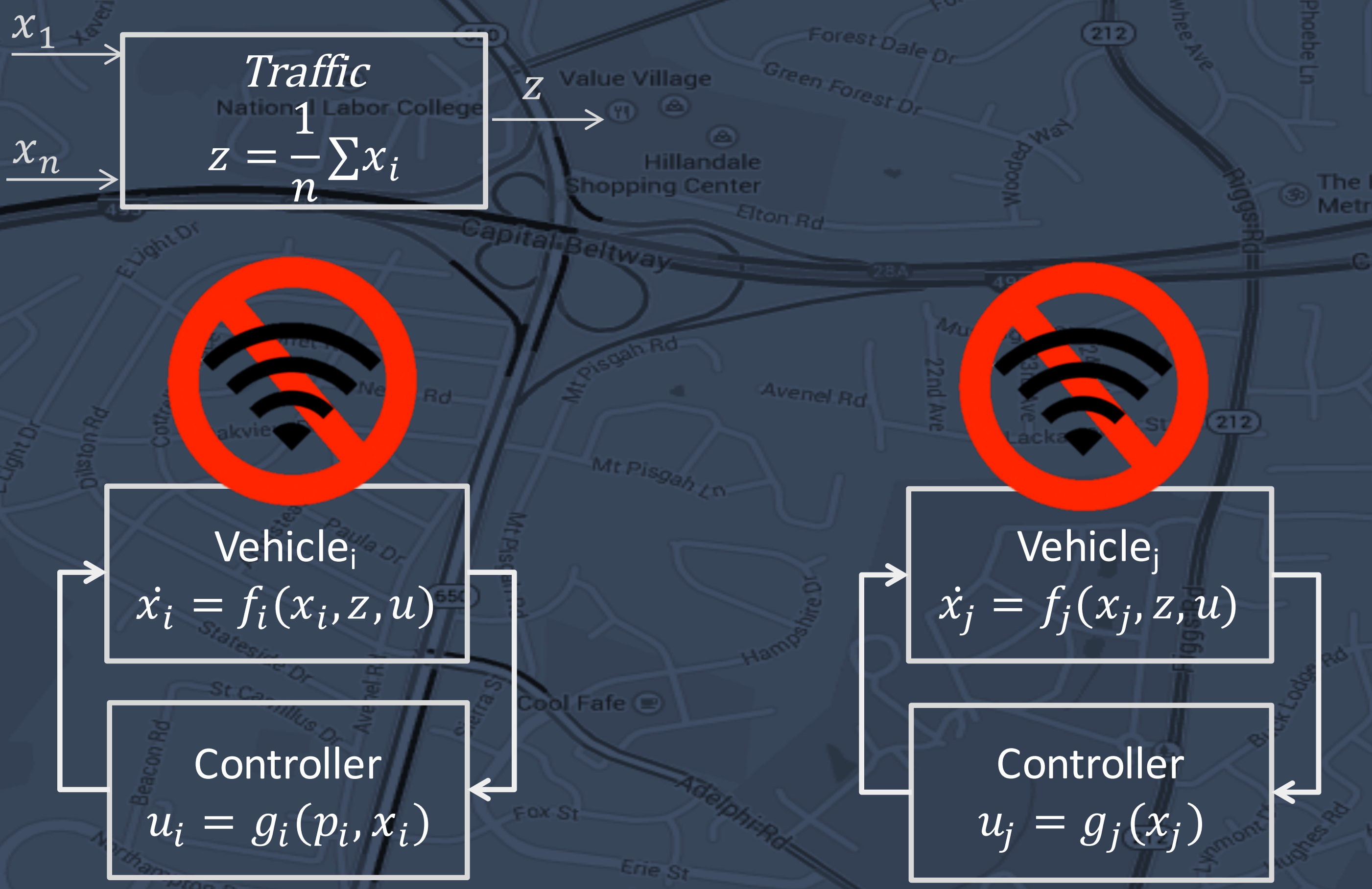
Part II

PRIVACY IN CYBER-PHYSICAL SYSTEMS CONTROL

[HiCons 2014] [CDC 2014] [ICDCN 2015]

- Participants share private information for social benefit
- Unfettered sharing can expose users in unexpected ways
- Adding noise to private information can give privacy by sacrificing some accuracy
- Privacy–accuracy trade-off in database

agents sharing no location data



x_1

x_n

Traffic

$$z = \frac{1}{n}\sum x_i$$

z

Vehicle$_i$

$$\dot{x}_i = f_i(x_i, z, u)$$

Controller

$$u_i = g_i(p_i, x_i)$$

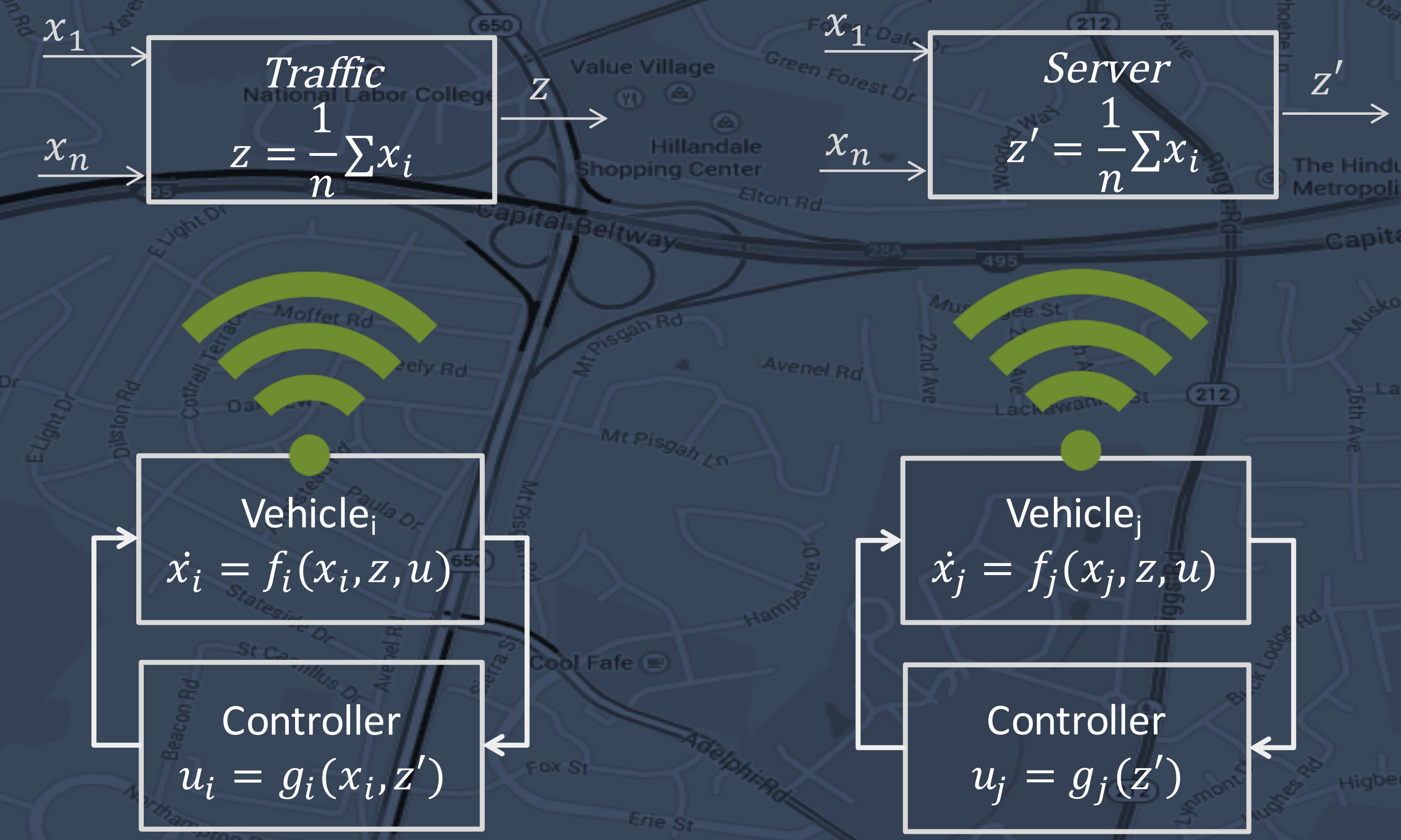
Vehicle$_j$

$$\dot{x}_j = f_j(x_j, z, u)$$

Controller

$$u_j = g_j(x_j)$$

agents sharing complete location data



x_1

x_n

Traffic

$$z = \frac{1}{n}\sum x_i$$

z

x_1

x_n

Server

$$z' = \frac{1}{n}\sum x_i$$

z'

Vehicle$_i$

$$\dot{x}_i = f_i(x_i, z, u)$$

Controller

$$u_i = g_i(x_i, z')$$

Vehicle$_j$

$$\dot{x}_j = f_j(x_j, z, u)$$

Controller

$$u_j = g_j(z')$$

M^*

better distributed control while protecting private location data

Obs: observation stream (location data) of the system bounded by time T

Sensitive data: location way points of all agents $g = \{g_1, \dots, g_n\}$

g and g' be two sequences location waypoints that are identical except g_i and g_i'. The system is **differentially private** iff

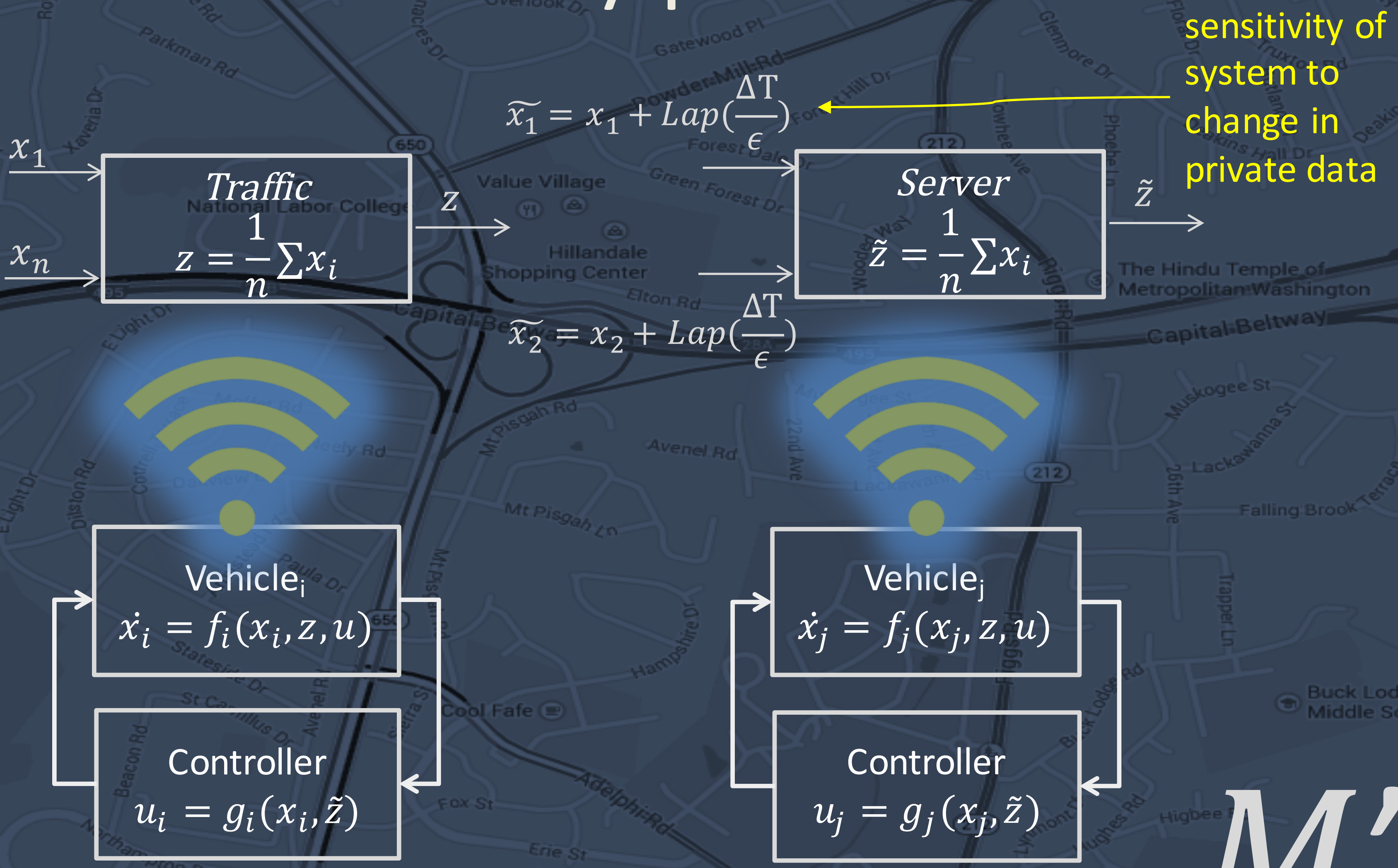
$$\frac{P[g\ leads\ to\ Obs]}{P[g'\ leads\ to\ Obs]} \leq e^{|g_i - g_i'|}$$

Cost of privacy: $\sup_{g,i} E[Cost(g, M^*) - Cost(g, M')]$

Worst case loss of efficiency (over all location waypoints of any agent) for using differentially private sharing

What is the cost of privacy in distributed control?

differentially private control



$\widetilde{x_1} = x_1 + Lap(\frac{\Delta T}{\epsilon})$

sensitivity of system to change in private data

x_1

x_n

Traffic

$z = \frac{1}{n}\sum x_i$

z

Server

$\tilde{z} = \frac{1}{n}\sum x_i$

$\tilde{z}$

$\widetilde{x_2} = x_2 + Lap(\frac{\Delta T}{\epsilon})$

Vehicle$_i$

$\dot{x}_i = f_i(x_i, z, u)$

Controller

$u_i = g_i(x_i, \tilde{z})$

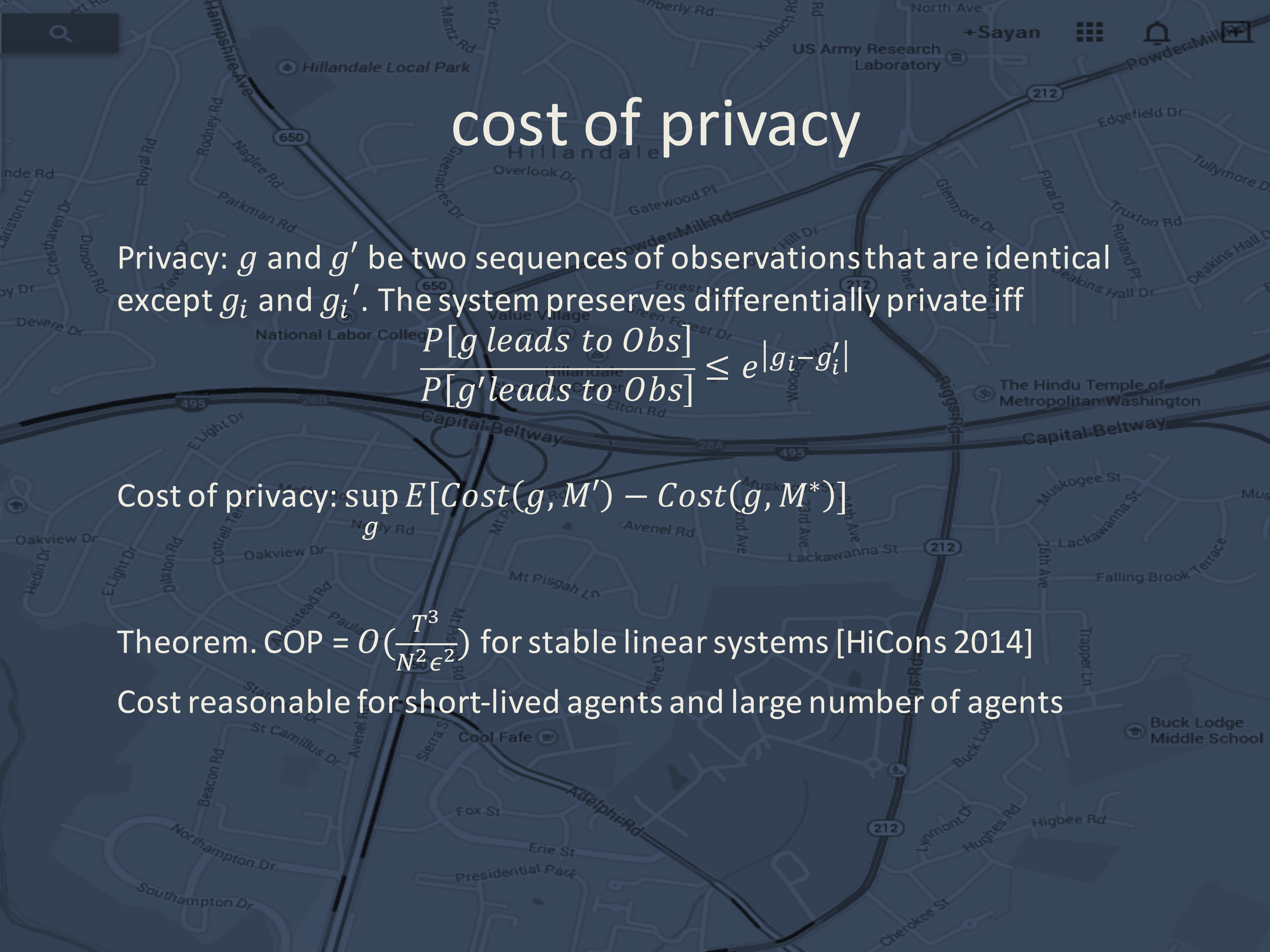
Vehicle$_j$

$\dot{x}_j = f_j(x_j, z, u)$

Controller

$u_j = g_j(x_j, \tilde{z})$

M'



cost of privacy

Privacy: g and g' be two sequences of observations that are identical except g_i and $g_i{}'$. The system preserves differentially private iff

$$\frac{P[g\ leads\ to\ Obs]}{P[g'\ leads\ to\ Obs]} \leq e^{|g_i - g_i'|}$$

Cost of privacy: $\sup_{g} E[Cost(g, M') - Cost(g, M^*)]$

Theorem. COP = $O(\frac{T^3}{N^2 \epsilon^2})$ for stable linear systems [HiCons 2014]

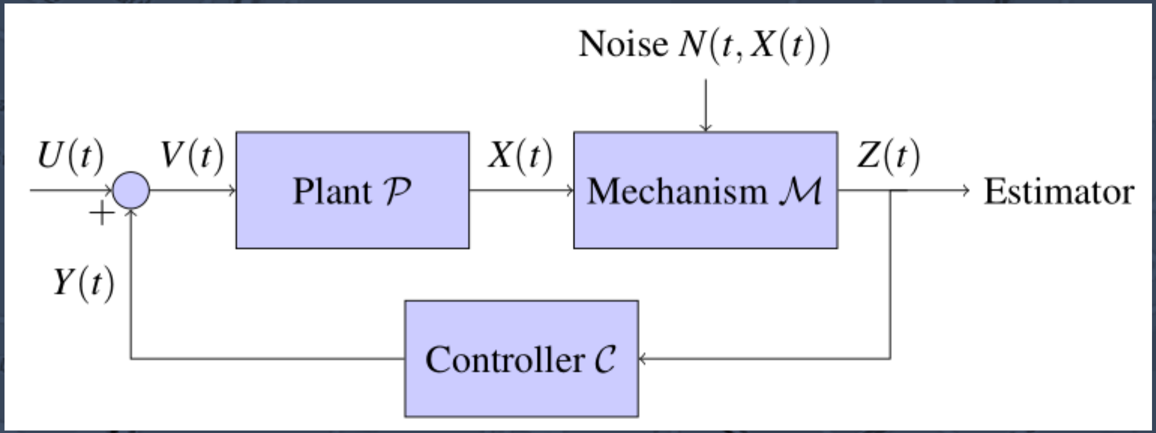
Cost reasonable for short-lived agents and large number of agents

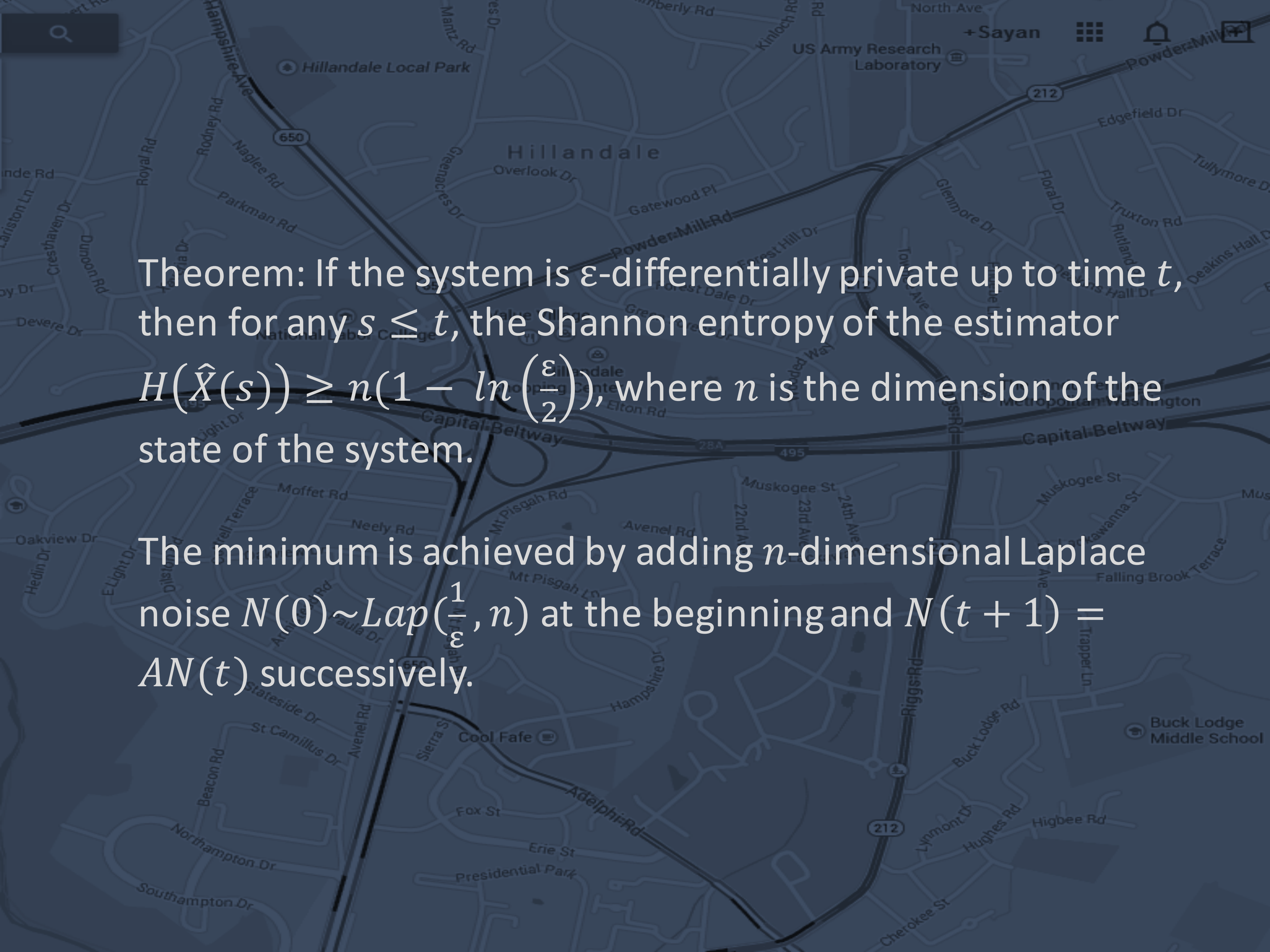
lower-bound on estimation accuracy [Wang et al. CDC 2014]

suppose adversary estimates the initial system state from observations

minimal mean square estimator: $\hat{X}(t) = \mathbb{E}[X(0)|Z(t), \ldots, Z(0)]$

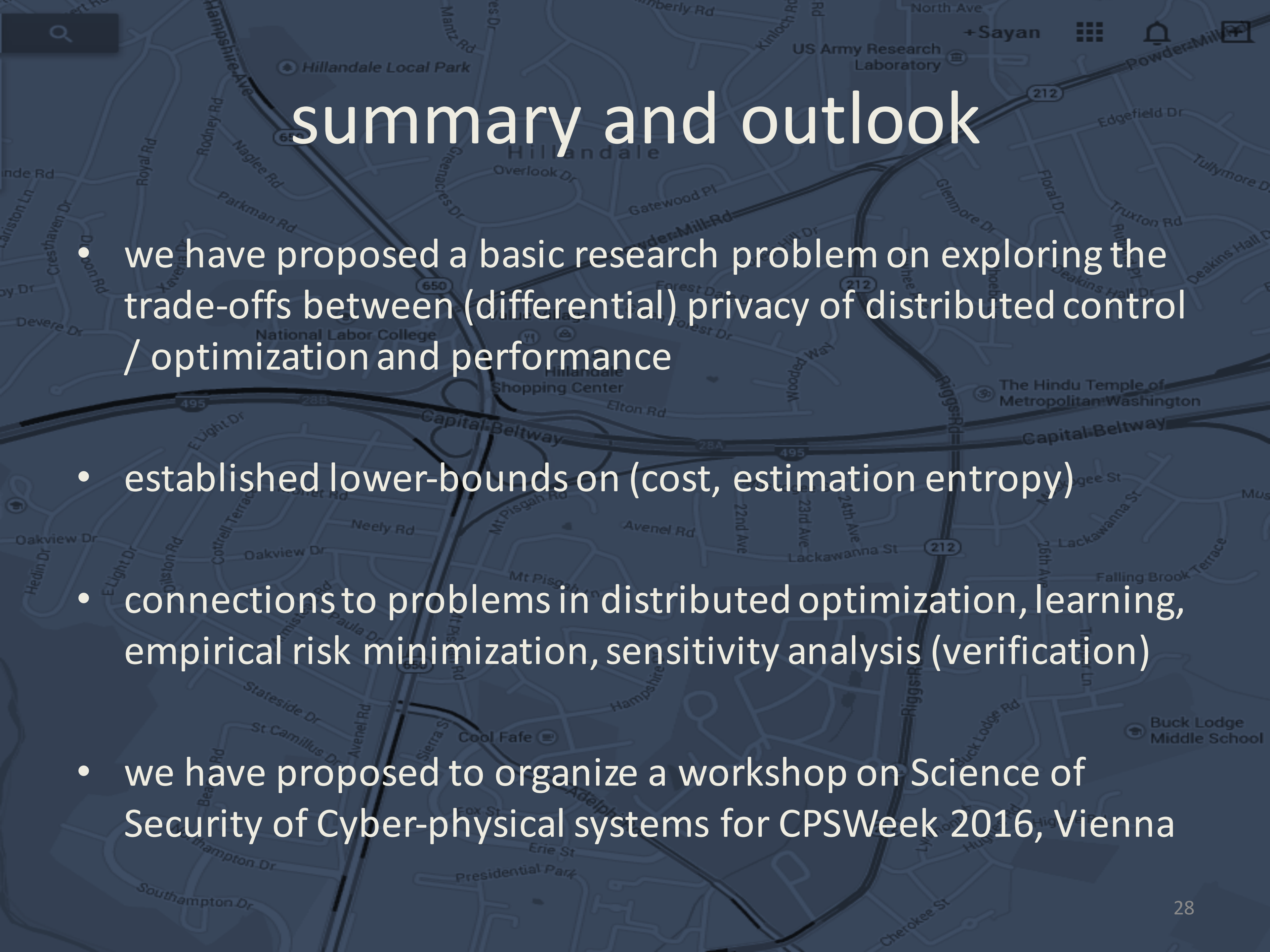
accuracy of this estimation process at time t ∈ N is measured by the entropy of the sequence $H(\hat{X}(t))$





Theorem: If the system is ε-differentially private up to time t, then for any $s \leq t$, the Shannon entropy of the estimator $H(\hat{X}(s)) \geq n(1- \ln\left(\frac{\varepsilon}{2}\right))$, where n is the dimension of the state of the system.

The minimum is achieved by adding n-dimensional Laplace noise $N(0) \sim Lap(\frac{1}{\varepsilon}, n)$ at the beginning and $N(t+1) = AN(t)$ successively.



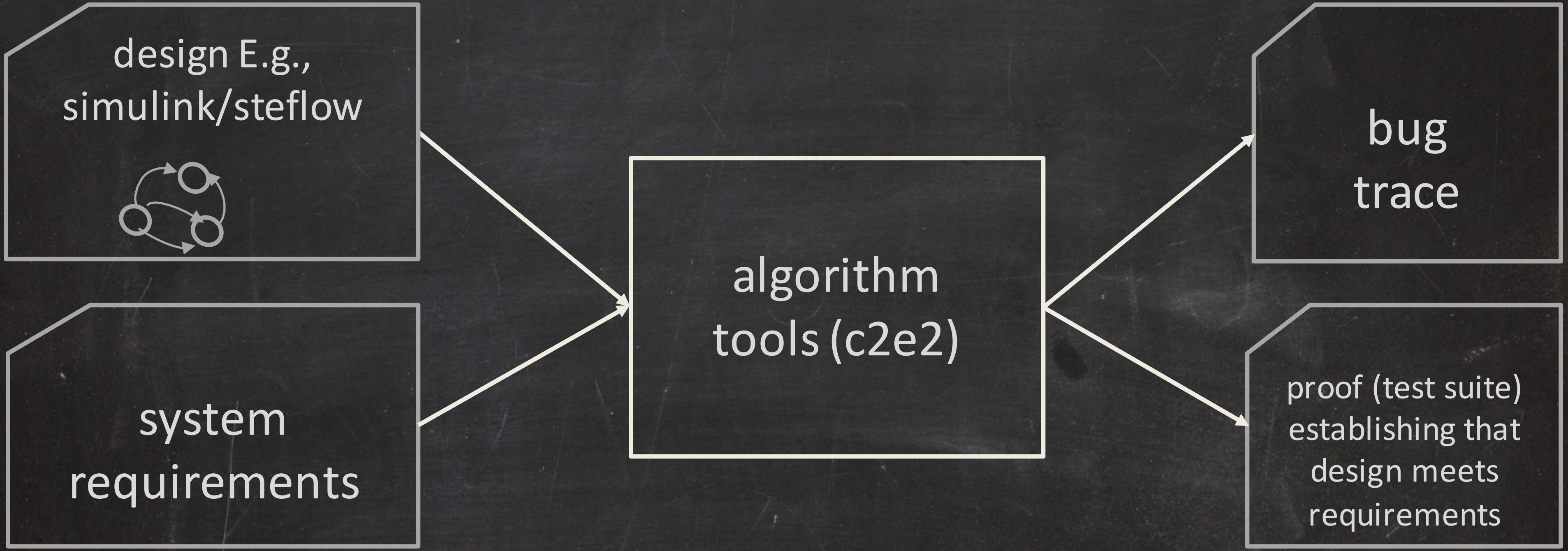
summary and outlook

- we have proposed a basic research problem on exploring the trade-offs between (differential) privacy of distributed control / optimization and performance
- established lower-bounds on (cost, estimation entropy)
- connections to problems in distributed optimization, learning, empirical risk minimization, sensitivity analysis (verification)
- we have proposed to organize a workshop on Science of Security of Cyber-physical systems for CPSWeek 2016, Vienna

Part III

MEETING CPS VERIFICATION CHALLENGES

verification problem



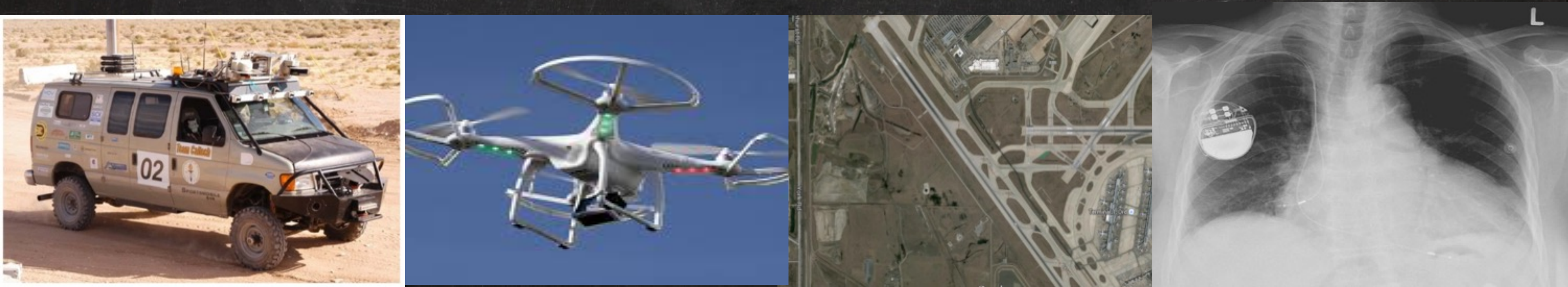
design E.g.,
simulink/steflow

system
requirements

algorithm
tools (c2e2)

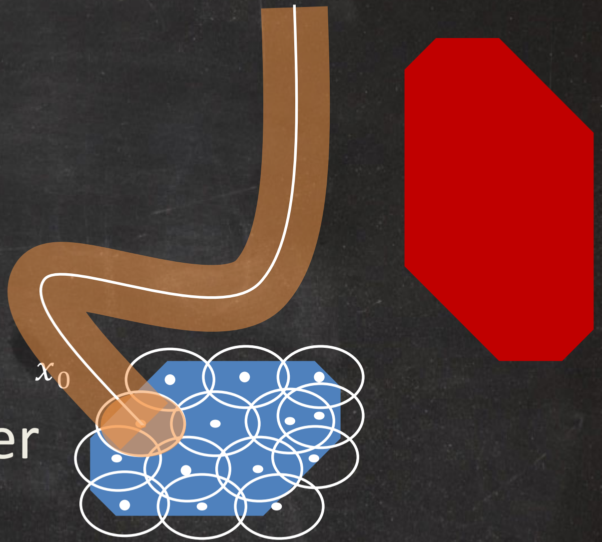
bug
trace

proof (test suite)
establishing that
design meets
requirements



strategy: combine concrete numerical simulations with symbolic analysis

- given start S and target T
- compute finite cover of initial set
- **numerically simulate** from center x_0 of each cover
- **symbolically bloat** simulation so bloated tube contains all trajectories from the cover
- Union = over-approximation of reach set
- Check intersection/containment with T
- Refine



- symbolic bloat computed from static analysis of models; this is related to sensitivity [HSCC 2014] [ATVA 2015]

sound & relatively complete

Theorem. (Soundness). Given hybrid automaton A, initial set Θ, unsafe set U, time bound T, bound on discrete transitions N, if the algorithm 1 returns safe or unsafe, then A is safe or unsafe.

Definition (Robust Safety). Given HA $A = \langle V, Loc, A, D, T \rangle$, an ϵ-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.

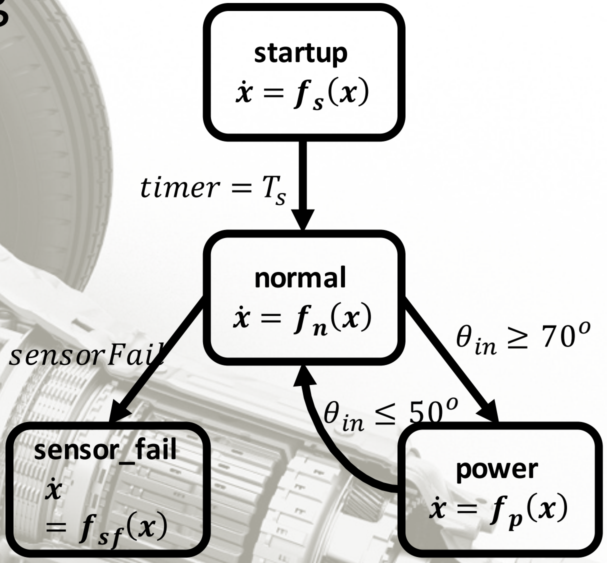
Theorem. (Relative Completeness) The algorithm will always terminate whenever the system is either robustly safe or robustly unsafe.

application 1: powertrain verification

powertrain design is a critical piece for meeting fuel efficiency and emissions targets for automotive industry

simulink model of a powertrain control benchmarks presented by ***Toyota*** *[ATVA, HSCC2014] as a verification challenge.*

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

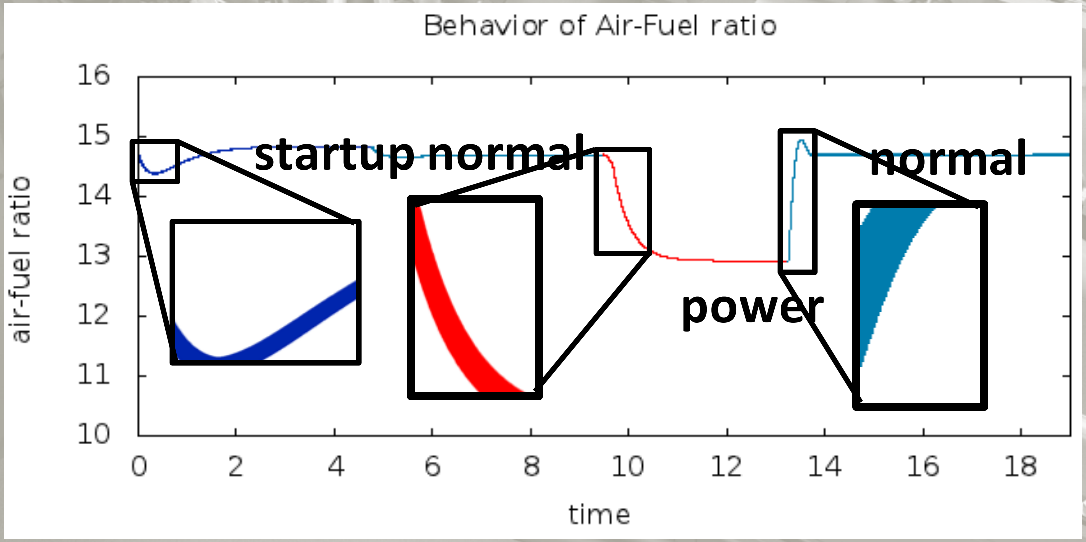


application 1: powertrain verification

our tool C2E2 is the **first to verify air-fuel ratio** remains within required range for a set of driver behaviors

analysis is mostly **automatic**. **project took less than 2 months**

[CAV 2015] [ARCH 2015 award winning paper]

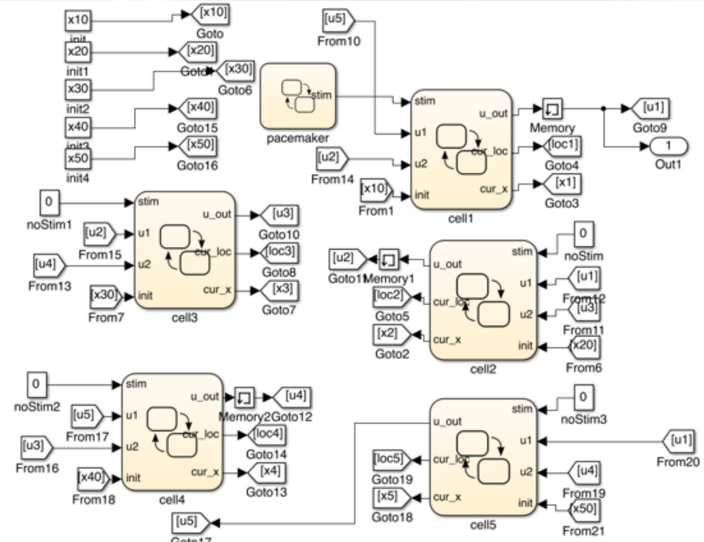


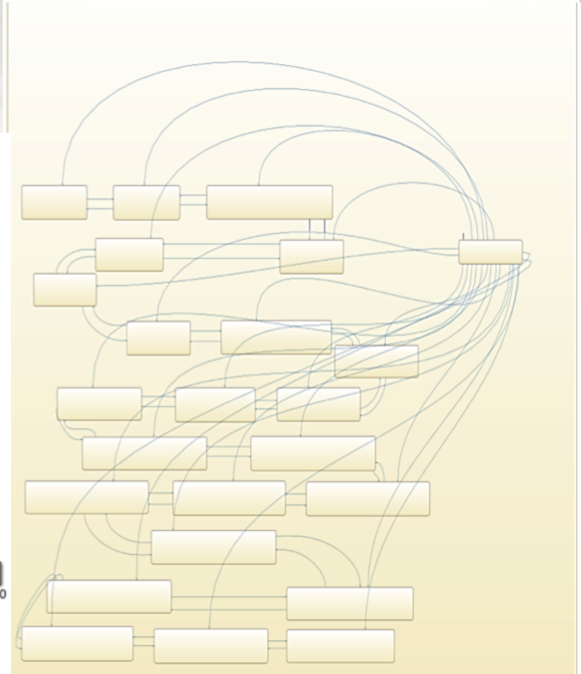
application 2: pacemaker verification

2M medical devices recalled in the past decade; 24 % owing to software defects

challenge problem: verify properties of a pacemaker composed with a model of cardiac tissue

composition of many identical cells: millions of modes, nonlinear differential equations; compositional analysis





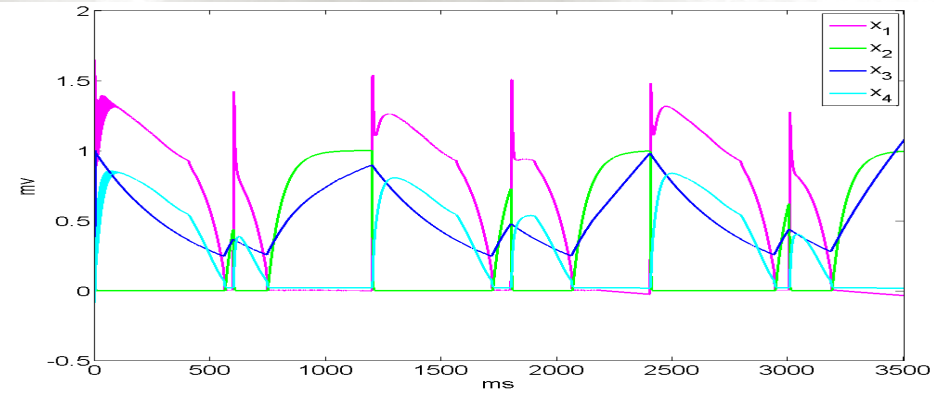
```
state1
du:
u_dot=-0.0025000000000000*u+D*(u1+u2-2*u)/(h*h)+stim;
v_dot=-0.0166666666666667*v+0.0166666666666667;
w_dot=-0.0726392130750601*u-0.0050000000000000*w+0.0050000000000000;
s_dot=0.0325954614796371*u-0.3657376929266330*s+0.0078827602517302;
u_out=u;
cur_x[0] = u;
cur_x[1] = v;
cur_x[2] = w;
cur_x[3] = s;
cur_loc=1;

[u<0.0032252252252252]  2
[u>=0.0032252252252252]

state2
du:
u_dot=-0.0025934648787471*u+0.0000003014452846+D*(u1+u2-2*u)/(h*h)+stim;
v_dot=-0.0166666666666667*v+0.0166666666666667;
w_dot=-0.0726392130750601*u-0.0050000000000000*w+0.0050000000000000;
s_dot=0.0342238163406254*u-0.3657376929266330*s+0.0078775084405570;
u_out=u;
cur_x[0] = u;
cur_x[1] = v;
cur_x[2] = w;
cur_x[3] = s;
cur_loc=2;

[u>=0.0059]  1
[u<0.0059]  1

state3
du:
u_dot=-99.8500002249997323*u+0.5891000013299984+D*(u1+u2-2*u)/(h*h)+stim;
v_dot=-166.6666666666666003*u-0.9486956521739127*v+157.9710144927535680*u*v+0.9999999999999996;
w_dot=1.2857135714285342*u-0.0050000000000000*w-0.0030142814285712;
s_dot=0.0000003317449342*u-0.3657376929266330*s+0.0080794269996715;
u_out=u;
cur_x[0] = u;
cur_x[1] = v;
cur_x[2] = w;
cur_x[3] = s;
cur_loc=3;
```

application 2: pacemaker verification

new algorithm for compositionally computing symbolic bloat using ideas from input-to-state stability [Huang & Mitra, HSCC 2014]

first to verify this class of models [Huang et al. CAV 2014]

synthesize pacemaker parameters that prevent pacemaker induced tachycardia [Huang et. al. IEEE Design and Test]



Nodes	Thresh	Sims	Run time (s)	Property
3	2	16	104.8	TRUE
3	1.65	16	103.8	TRUE
5	2	3	208	TRUE
5	1.65	5	281.6	TRUE
5	1.5	NA	63.4	FALSE
8	2	3	240.1	TRUE
8	1.65	73	2376.5	TRUE

summary

- we have developed algorithms and a software tool for verification of a general class of cyber-physical system models
  - applied it to meet several verification challenges

- establishes connection between formal verification, synthesis, and privacy of cyber-physical systems