Temporal logic control of continuous systems

George J. Pappas

Departments of ESE and CIS

University of Pennsylvania

pappasg@ee.upenn.edu

http://www.seas.upenn.edu/~pappasg

Penn
UNIVERSITY of PENNSYLVANIA

An invitation

Hybrid Systems : Computation and Control

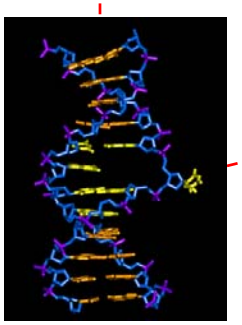
University of Pennsylvania

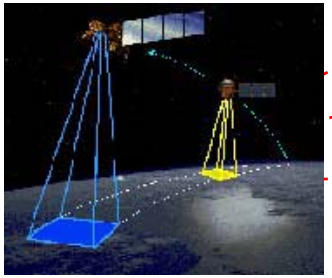
March 25-27, 2004

http://www.seas.upenn.edu/hybrid/HSCC04/

Penn
University of Pennsylvania

Multi-agent systems…





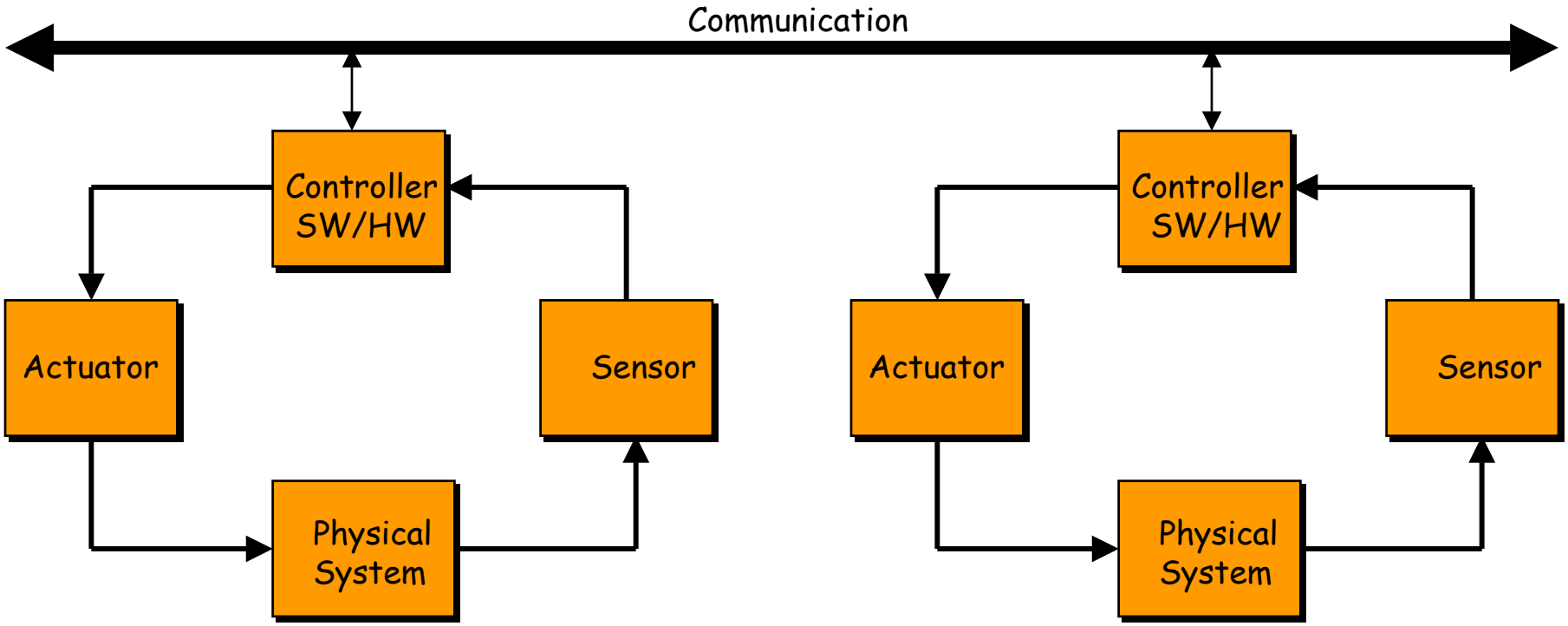
- Examples
  - Automated Highway Systems
  - Formation Flight
  - MEM arrays/Smart structures
  - Biochemical networks
  - Satellite clusters
- Application areas
  - Auto/Aerospace industry
  - Molecular Biology
  - Telecommunications
- Significance
  - key technologies
  - social impact
  - intellectual challenge

Penn
University of Pennsylvania

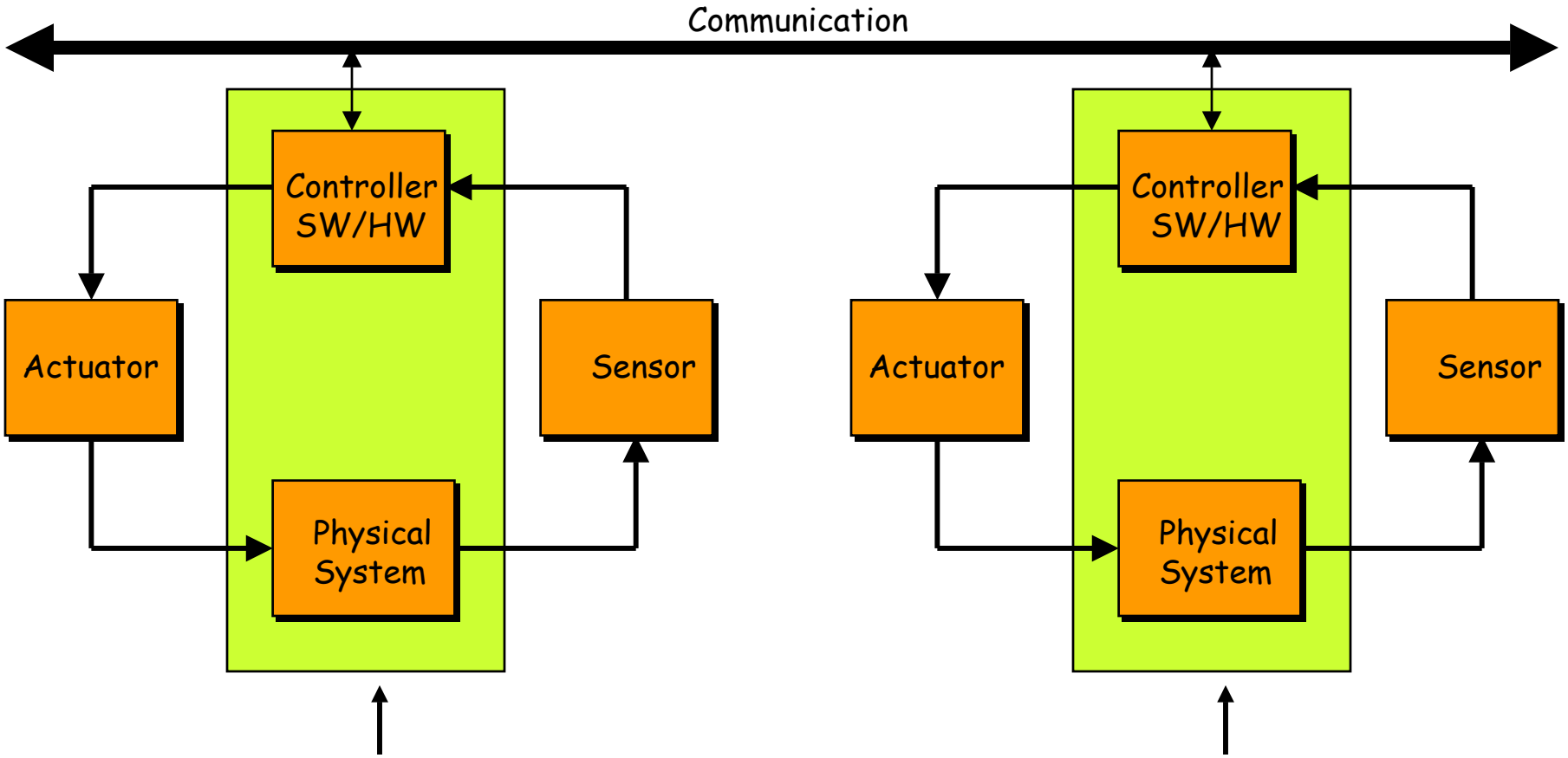
Multiple UAVs

Penn
University of Pennsylvania

Concurrent continuous systems…



Concurrent continuous systems…



Physical system is continuous, software is discrete

Agents are synchronized but only partially

Specification challenges

Continuous model

$$x_1(t+1) = A_1 x_1(t) + B_1 u_1(t)$$
$$x_2(t+1) = A_2 x_2(t) + B_2 u_2(t)$$
$$x_3(t+1) = A_3 x_3(t) + B_3 u_3(t)$$

Agents can exchange data if the they are physically close, say $|x_1 - x_2| \leq 50$

Hybrid specification

Move freely within a specified region S

Avoid collisions with each other, say $|x_1 - x_2| \geq 5$

Exchange data (no later than) every 60 time units

Specification challenges

Continuous model

$$x_1(t+1) = A_1 x_1(t) + B_1 u_1(t)$$
$$x_2(t+1) = A_2 x_2(t) + B_2 u_2(t)$$
$$x_3(t+1) = A_3 x_3(t) + B_3 u_3(t)$$

Agents can exchange data if the they are physically close, say $|x_1 - x_2| \leq 50$

Linear temporal logic (LTL) specification

Move freely but always within a specified region S

$$\varphi_1 := \square\, (x_1 \in S \wedge x_2 \in S \wedge x_3 \in S)$$

Avoid collisions with each other, say $|x_1 - x_2| \geq 5$

$$\varphi_2 := \square\, (|x_1 - x_2| \geq 5 \wedge |x_1 - x_3| \geq 5 \wedge |x_2 - x_3| \geq 5)$$

Exchange data (no later than) every 60 time units

$$\varphi_3 := \square\, (\Diamond_{60} |x_1 - x_2| \leq 50 \wedge \Diamond_{60} |x_1 - x_3| \leq 50 \wedge \Diamond_{60} |x_2 - x_3| \leq 50)$$

$$\varphi = \varphi_1 \wedge \varphi_2 \wedge \varphi_3$$

Penn
University of Pennsylvania

A verification problem

Given dynamical system S, and temporal logic formula φ

Basic verification problem

$$S \models \varphi$$

Two main approaches

Model checking : Algorithmic, restrictive
Deductive methods : Semi-automated, general

A synthesis problem

Given control system S, and temporal formula φ

Basic synthesis problem

$$S \parallel C \models \varphi$$

Controller is necessarily a hybrid system...

Composition semantics can be defined...

Technical outline

Symbolic transition systems

Emphasis on region algebras and finite bisimulations

Temporal logic verification of linear dynamical systems

Emphasis on finite bisimulations using order-minimality

Temporal logic synthesis of linear control systems

Emphasis on finite bisimulations using Brunovsky canonical forms

Symbolic transition systems*

A symbolic transition system

$$S = (Q, \delta, R, P, [\cdot])$$

consists of

A (in)finite set of states	Q	Finite sets, reals
A (in)finite set of regions	R	BDDs, polyhedra
A finite set of observables	$P \subset R$	
The transition function	$\delta : Q \to 2^Q$	Possibly nondeterministic
The extension function	$[\cdot] : R \to 2^Q$	

Symbolic transition systems are equipped with a **region algebra** of sets

*T.A. Henzinger, R. Majumdar, J.F. Raskin, ***A classification of symbolic transition systems***, ACM Transactions on Computational Logic, June 2003.

Symbolic transition systems

Set of observables covers the state space $\bigcup_{p \in P} \langle p \rangle = Q$

For every observable p, there is a complementary observable $\bar{p}$

For regions s and t in R, there are **computable** regions for

$$[\mathrm{And}(s, t)] = [s] \bigcap [t]$$

$$[\mathrm{Diff}(s, t)] = [s] \setminus [t]$$

$$[\mathrm{Pre}(s)] = \{q \in Q \mid \exists q' \in \delta(q) \wedge q' \in s\}$$

Emptiness Empty(s) and membership Member(q,s) can be **decided**

Region Algebra $R_S = (P, \mathrm{Pre}, \mathrm{And}, \mathrm{Diff}, \mathrm{Empty})$

A continuous example

$$S = (Q, \delta, R, P, [\cdot])$$

$$x(t+1) = Ax(t)$$

$$\varphi := X_0 \wedge X_0 \Rightarrow \Diamond X_f$$

Symbolic Transition System S

$Q = R^n$

$x' \in \delta(x) \Leftrightarrow x' = Ax$

R = Semi-linear sets

$P = \{X_0, X_F, R^n \setminus (X_0 \bigcup X_F)\} \bigcup \bar{P}$

$Pre(s) = \{\text{states that reach } s\}$

Region algebra $R_S = (P, Pre, And, Diff, Empty)$

Linear temporal logic

Linear temporal logic syntax

The LTL formulas are defined inductively as follows

Atomic propositions

All observation symbols p are formulas

Boolean operators

If φ_1 and φ_2 are formulas then

$\varphi_1 \vee \varphi_2$ $\neg\varphi_1$

Temporal operators

If φ_1 and φ_2 are formulas then

$\varphi_1 \, U \, \varphi_2$ $\bigcirc \varphi_1$

Linear temporal logic

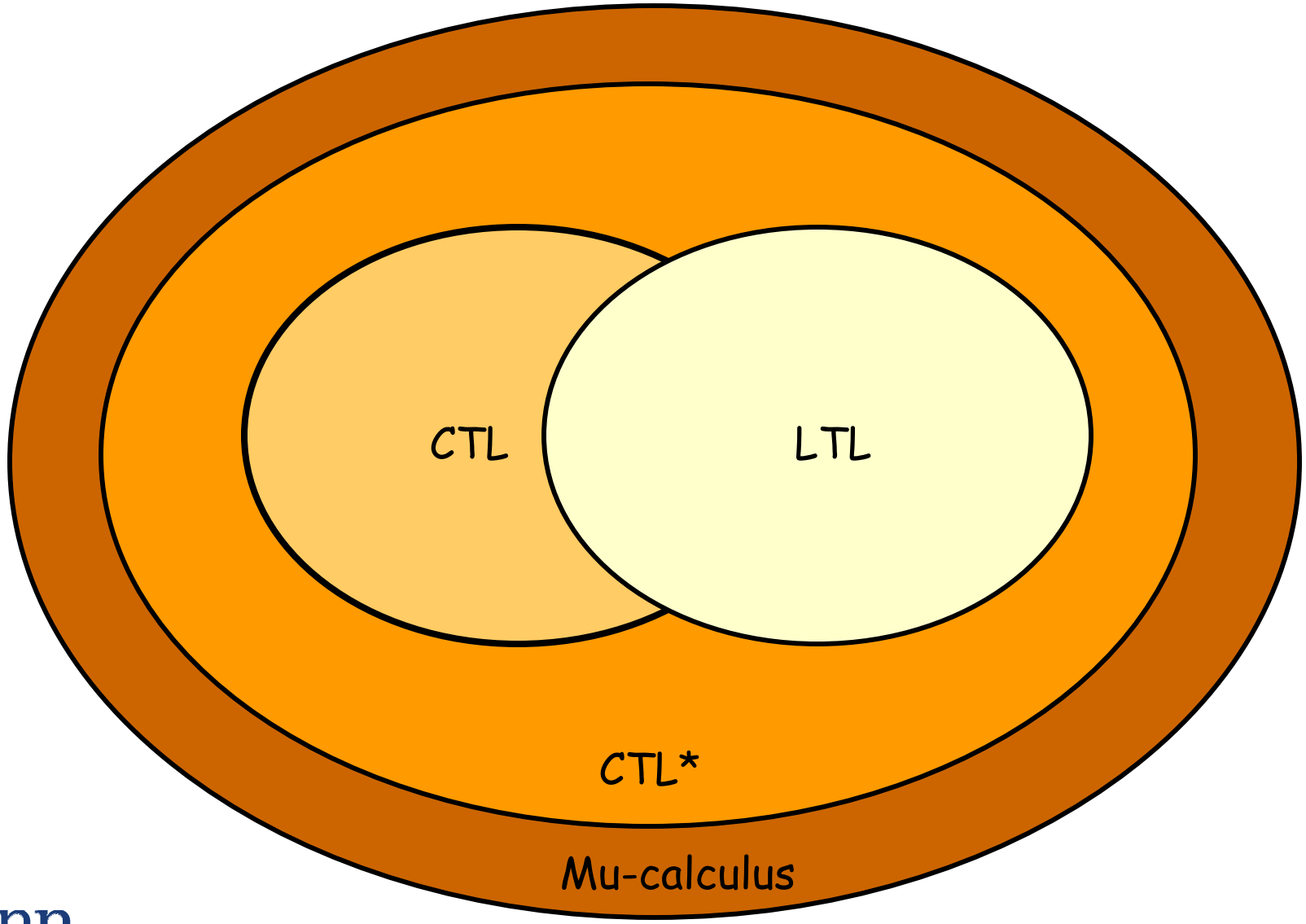
Syntactic boolean abbreviations

Conjunction	$\varphi_1 \wedge \varphi_2 = \neg(\neg\varphi_1 \vee \neg\varphi_2)$
Implication	$\varphi_1 \Rightarrow \varphi_2 = \neg\varphi_1 \vee \varphi_2$
Equivalence	$\varphi_1 \Leftrightarrow \varphi_2 = (\varphi_1 \Rightarrow \varphi_2) \wedge (\varphi_2 \Rightarrow \varphi_1)$

Syntactic temporal abbreviations

Eventually	$\Diamond\, \varphi = \top\, U\, \varphi$
Always	$\Box\, \varphi = \neg \Diamond\, \neg\varphi$
In 3 steps	$\bigcirc_3\ \varphi = \bigcirc\, \bigcirc\, \bigcirc\, \varphi$
Within 3 steps	$\Diamond_3\, \varphi = \bigvee_{i=0}^{3} \bigcirc_i\, \varphi$

Comparing logics



Mu-calculus model checking

Model-checking semi-algorithm

Input $R_S = (P, Pre, And, Diff, Empty), \mu-$ formula φ

Output $[\varphi] :=$

If $\varphi = p$ return $\{p\}$

If $\varphi = \overline{p}$ return $\{Diff(q,p) \mid p \in P\}$

If $\varphi = \varphi_1 \vee \varphi_2$ return $[\varphi_1] \cup [\varphi_2]$

If $\varphi = \varphi_1 \wedge \varphi_2$ return $\{And(s,t) \mid s \in [\varphi_1]\ , t \in [\varphi_2]\}$

If $\varphi = \exists \bigcirc \psi$ return $\{Pre(s) \mid s \in [\psi]\}$

If $\varphi = \forall \bigcirc \psi$ return $P \setminus\setminus \{Pre(s) \mid s \in (P \setminus\setminus [\psi])\}$

If $\varphi = (\mu x : \psi)$ return $\ldots$

If $\varphi = (\nu x : \psi)$ return $\ldots$

State equivalence

Given a state equivalence $\cong$ on the state space we define the

Quotient transition system

$$S/\cong = (Q/\cong, \delta/\cong, R, P, [\cdot]/\cong)$$

where $Q/\cong$ is the set of equivalenc e classes

$t \in \delta/\cong(s)$ if there exist $q' \in t$ and $q \in s$ such that $q' \in \delta(q)$

$s \in [p]/\cong$ if there exists $q \in s$ such that $q \in [p]$

Bisimulation equivalence

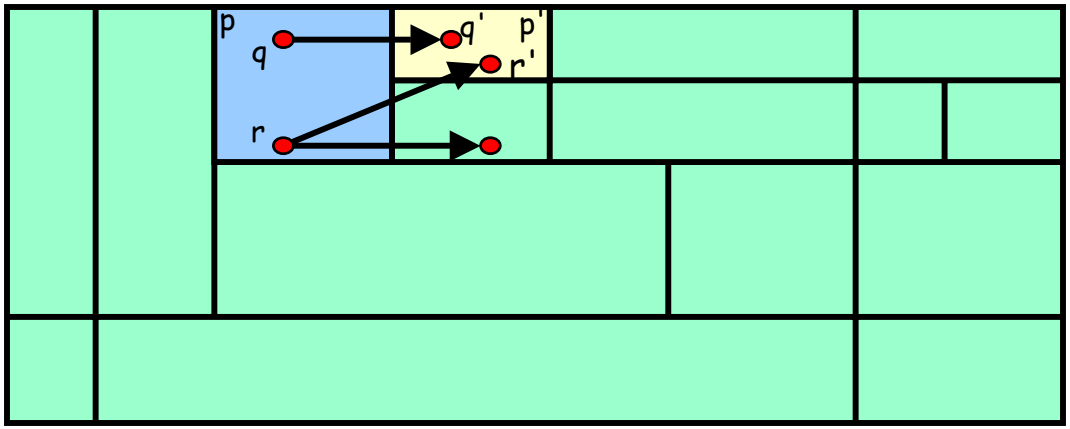
Bisimulation is a special state equivalence

State equivalence $\cong$ is a bisimulation iff the following conditions hold for any two equivalent states $q \cong r$

For every observable $p \in P$, $q \in [p]$ iff $r \in [p]$

For every $q' \in \delta(q)$ there is $r' \in \delta(r)$ and $q' \cong r'$

For every $r' \in \delta(r)$ there is $q' \in \delta(q)$ and $q' \cong r'$



Bisimulation algorithm

Bisimulation semi-algorithm

Input $R_S = (P, Pre, And, Diff, Empty)$

Initialize $T_0 := P$

while $[T_{i+1}] \subseteq [T_i]$

$$T_{i+1} := T_i \cup \{Pre(s) \mid s \in T_i\}$$
$$\cup \{And(s,t) \mid s,t \in T_i\}$$
$$\cup \{Diff(s,t) \mid s,t \in T_i\}$$

end while

Algorithm terminates if no new regions are generated

If S is infinite, there is no guarantee of termination

Preserved properties

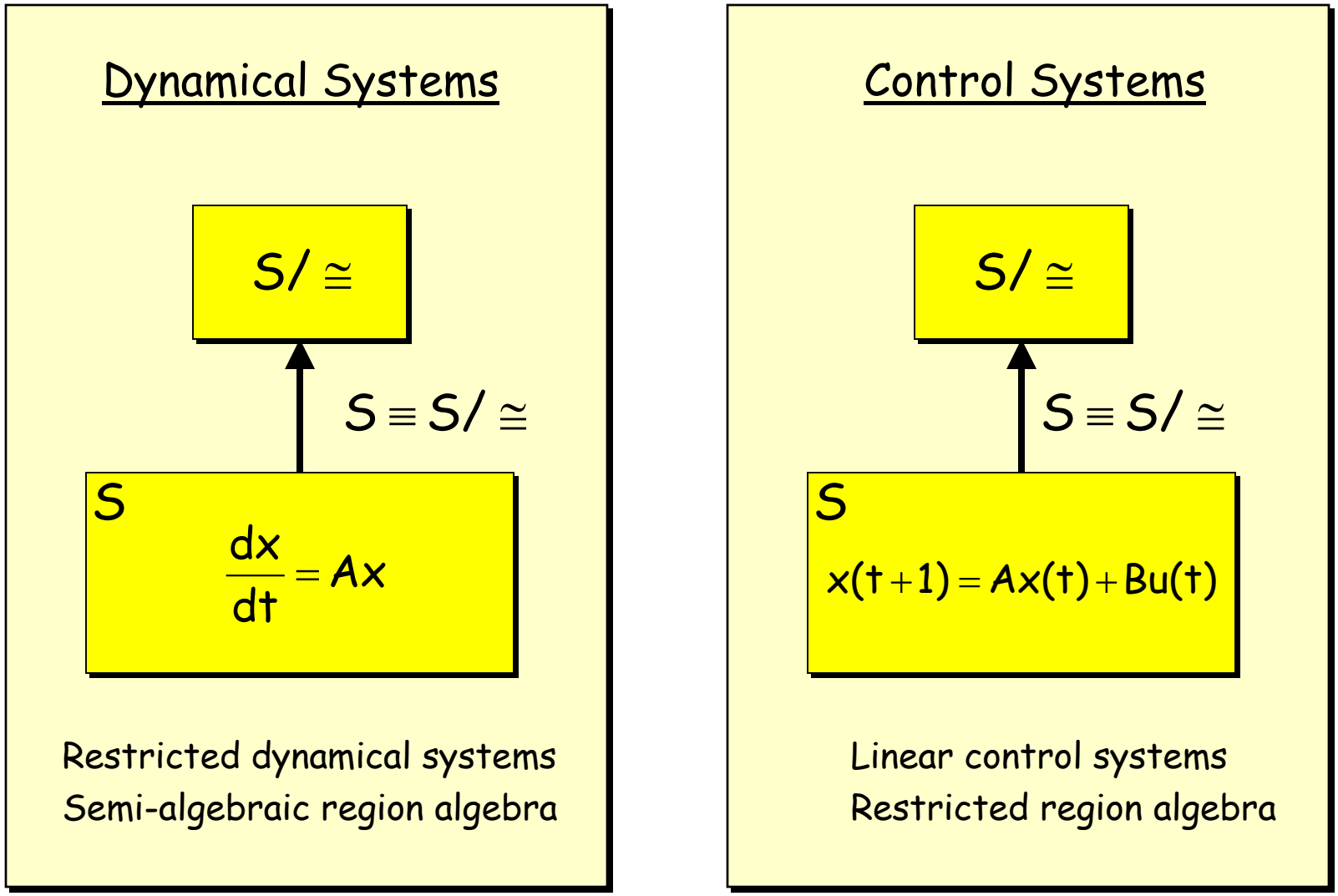
mu-calculus (also CTL and LTL) equivalence

If $\cong$ is a bisimulation, then $S \models \varphi \Leftrightarrow S/\cong \models \varphi$

Decidable mu-calculus model checking

If $\cong$ is a **finite** bisimulation, then the model checking algorithm terminates

Infinite to finite



Technical outline

Symbolic transition systems

Emphasis on region algebras and finite bisimulations

Temporal logic verification of linear dynamical systems

Emphasis on finite bisimulations using order-minimality

Temporal logic synthesis of linear control systems

Emphasis on finite bisimulations using Brunovsky canonical forms

Continuous verification

$$S = (Q, \delta, R, P, [\cdot])$$

$$\frac{dx}{dt} = Ax$$

Symbolic Transition System S

$Q = R^n$

$x' \in \delta(x) \Leftrightarrow \exists t \geq 0 \quad \text{with} \quad x' = e^{At}x$

$R = \text{Semi-algebraic sets}$

$P = \{X_0, X_F, R^n \setminus (X_0 \bigcup X_F)\}$

$[Pre(s)] = \{x \in R^n \mid \exists x' \in \delta(x) \wedge x' \in s\}$

Is $R_S = (P, Pre, And, Diff, Empty)$ a region algebra?

Closure under Pre

Consider linear vector fields of the form F(x)=Ax where

- A is rational and nilpotent
- A is rational, diagonalizable, with rational eigenvalues
- A is rational, diagonalizable, with purely imaginary, rational eigenvalues

Let S be any semi-algebraic set. Then Pre(S) is also semi-algebraic*.

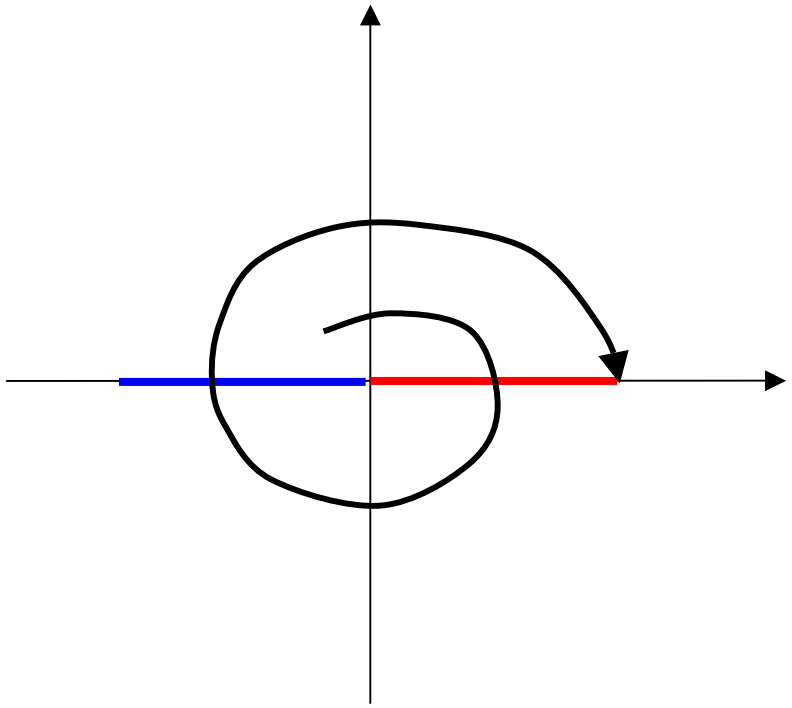
Under such assumptions on A

$$R_S = (P, Pre, And, Diff, Empty)$$

is a region algebra

*G. Lafferriere, G.J. Pappas, S. Yovine, **Symbolic reachability computations for families of linear vector fields**, Journal of Symbolic Computation, 2001.

Finite bisimulation ?



Bisimulation algorithm never terminates !!

Sets

$$P_1 = \{(x,0) \mid 0 \le x \le 4\}$$

$$P_2 = \{(x,0) \mid -4 \le x < 0\}$$

$$P_3 = R^2 \setminus (P_1 \cup P_2)$$

Dynamics

$$\dot{x}_1 = 0.2x_1 + x_2$$

$$\dot{x}_2 = -x_1 + 0.2x_2$$

Penn
University of Pennsylvania

First-order logic

Every theory of the reals has an associated language

$$(\Re, <, +, -, 0, 1)$$

Universe Relation Functions Constants

Variables : $x_1, x_2, x_3, \ldots$

TERMS : Variables, constants, or functions of them

$$x_1 - x_2 + 1, 1 + 1, -x_3$$

ATOMIC FORMULAS : Apply the relation and equality to the terms

$$x_1 + x_2 < -1, 2x_1 = 1, x_1 = x_3$$

(FIRST ORDER) FORMULAS : Atomic formulas are formulas

If φ_1, φ_2 are formulas, then $\varphi_1 \vee \varphi_2, \neg\varphi_1, \forall x.\varphi_1, \exists x.\varphi_1$

First-order logic

$(\Re, <, +, -, 0, 1)$	$\forall x \forall y (x + 2y \geq 0)$
$(\Re, <, +, -, \times, 0, 1)$	$\exists x . ax^2 + bx + c = 0$
$(\Re, <, +, -, \times, e^x, 0, 1)$	$\exists t . (t \geq 0) \wedge (y = e^t x)$

A theory of the reals is **decidable** if there is an algorithm which in a finite number of steps will decide whether a formula is true or not

A theory of the reals admits **quantifier elimination** if there is an algorithm which will eliminate all quantified variables.

$$\exists x . ax^2 + bx + c = 0 \equiv b^2 - 4ac \geq 0$$

First-order logic

Theory	Decidable ?	Quant. Elim. ?
$(\Re, <, +, -, 0, 1)$	YES	YES
$(\Re, <, +, -, \times, 0, 1)$	YES	YES
$(\Re, <, +, -, \times, e^x, 0, 1)$	?	NO

Tarski's result : Every formula in $(\Re, <, +, -, \times, 0, 1)$ can be decided

1. Eliminate quantified variables
2. Quantifier free formulas can be decided

O-Minimal Theories

A definable set is $Y = \{(x_1, x_2, \ldots, x_n) \in \Re^n \mid \varphi(x_1, \ldots, x_n)\}$

A theory of the reals is called **o-minimal** if every definable subset of the reals is a **finite** union of points and intervals

Example: $Y = \{(x) \in \Re \mid p(x) \geq 0\}$ for polynomial p(x)

Recent o-minimal theories

$(\Re, <, +, -, 0, 1)$

$(\Re, <, +, -, \times, 0, 1)$

$(\Re, <, +, -, \times, e^x, 0, 1)$ ⟶ Related to Hilbert's 16th problem

$(\Re, <, +, -, \times, \hat{f}, 0, 1)$

$(\Re, <, +, -, \times, \hat{f}, e^x, 0, 1)$

Penn
University of Pennsylvania

Finite bisimulations

Finite bisimulations of dynamical systems*

Consider a vector field X and a finite partition of R^n where

1. The flow of the vector field is definable in an o-minimal theory
2. The finite partition is definable in the same o-minimal theory

Then a finite bisimulation always exists.

*G. Lafferriere, G.J. Pappas, and S. Sastry, **O-minimal hybrid systems**, Mathematics of Control, Signals and Systems, March 2000.

Corollaries

$(\Re, <, +, -, 0, 1)$

Consider continuous systems where

- Finite partition is polyhedral (semi-linear)
- Vector fields have linear flows (timed, multi-rate)

Then a finite bisimulation exists.

$(\Re, <, +, -, \times, 0, 1)$

Consider continuous systems where

- Finite partition is semialgebraic
- Vector fields have polynomial flows

Then a finite bisimulation exists.

Corollaries

$(\Re, <, +, -, \times, e^x, 0, 1)$

Consider continuous systems where

- Finite partition is semi-algebraic
- Vector fields are linear with real eigenvalues

Then a finite bisimulation exists.

$(\Re, <, +, -, \times, \hat{f}, 0, 1)$

Consider continuous systems where

- Finite partition is sub-analytic
- Vector fields are linear with purely imaginary eigenvalues

Then a finite bisimulation exists.

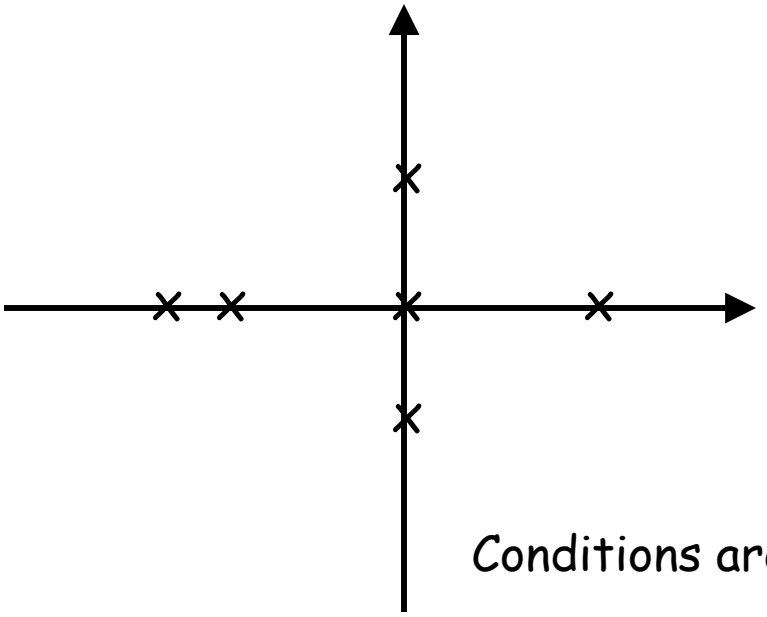
Corollaries

$(\Re, <, +, -, \times, \hat{f}, e^x, 0, 1)$

Consider continuous systems where

- Finite partition is semi-algebraic
- Vector fields are linear with real or imginary eigenvalues

Then a finite bisimulation exists.



Conditions are sufficient but tight

Model checking continuous systems

Consider linear vector fields of the form F(x)=Ax where

A is rational and nilpotent
A is rational, diagonalizable, with rational eigenvalues
A is rational, diagonalizable, with purely imaginary, rational eigenvalues

Then

1. Consider a finite semi-algebraic partition of the state space.
   Then a finite bisimulation always, exists and can be computed.

2. Consider an LTL formula where atomic propositions denote
   semi-algebraic sets. Then LTL model checking is decidable.

3. The reachability problem between semi-algebraic sets is decidable.

Technical outline

Symbolic transition systems

Emphasis on region algebras and finite bisimulations

Temporal logic verification of linear dynamical systems

Emphasis on finite bisimulations using order-minimality

Temporal logic synthesis of linear control systems

Emphasis on finite bisimulations using Brunovsky canonical forms

Control abstract transitions

$S = (Q, \delta, R, P, [\cdot])$

Δ

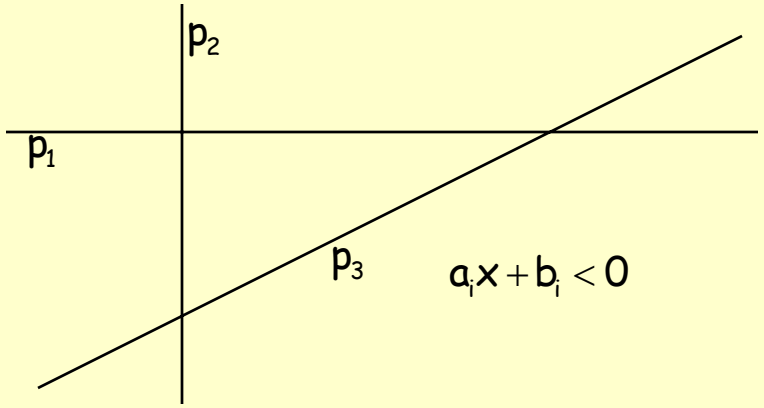
$x(t+1) = Ax(t) + Bu(t)$

Symbolic Transition System S

State set $Q = R^n$

$x' \in \delta(x) \Leftrightarrow \exists u$ with $x' = Ax + Bu$

R = Semi-linear sets



Observables are atomic propositions of temporal logic formulas

$R_S = (P, Pre, And, Diff, Empty)$

Termination?

Termination escapes us…probably undecidable

If the region algebra in addition satisfies

$$Pre(A \cap B) = Pre(A) \cap Pre(B)$$

$$Pre(\overline{A}) = \overline{Pre(A)}$$

$$\exists k \geq 0 \qquad Pre^k(A) = \Re^n$$

then a finite bisimulation exists and can be computed

We will search for a sub-algebra of semi-linear sets

Controllability

Assume the linear system is completely controllable

$$x(t+1) = Ax(t) + Bu(t)$$

Then by definition

$$Pre(Y) = \{x \in \Re^n \mid \exists y \in Y \, \exists u \quad y = Ax + Bu\}$$

and since the system is controllable

$$\exists k \leq n \qquad Pre^k(Y) = \Re^n$$

Controllability of linear systems can be decided using rank conditions

$$rank[B \;\; AB \;\; A^2B \;\; \ldots \;\; A^{n-1}B] = n$$

Searching for a sub-algebra

Another attempt : Rectangular sets but in Brunovsky coordinates

Boolean algebra generated by sets of the form

$$y_i \sim c_i \quad c_i \in Q, \quad \sim \in \{>, =, <\}$$

For any completely controllable linear system, there exist invertible linear transformations F and H, and a feedback G such that the resulting system is in Brunovsky normal form.

Original coordinates

$$x(t+1) = Ax(t) + Bu(t)$$

$$\begin{bmatrix} y \\ v \end{bmatrix} = U \begin{bmatrix} x \\ u \end{bmatrix}$$

$$U = \begin{bmatrix} F & 0_{n\times m} \\ G & H \end{bmatrix}$$

Brunovsky coordinates

$$\left.\begin{aligned} y_1(t+1) &= y_2(t) \\ y_2(t+1) &= y_3(t) \\ y_3(t+1) &= v_1(t) \end{aligned}\right\} = k_1$$

$$\left.\begin{aligned} y_4(t+1) &= y_5(t) \\ y_5(t+1) &= v_2(t) \end{aligned}\right\} = k_2$$

Brunovsky boolean algebra

Original coordinates

$$x(t+1) = Ax(t) + Bu(t)$$

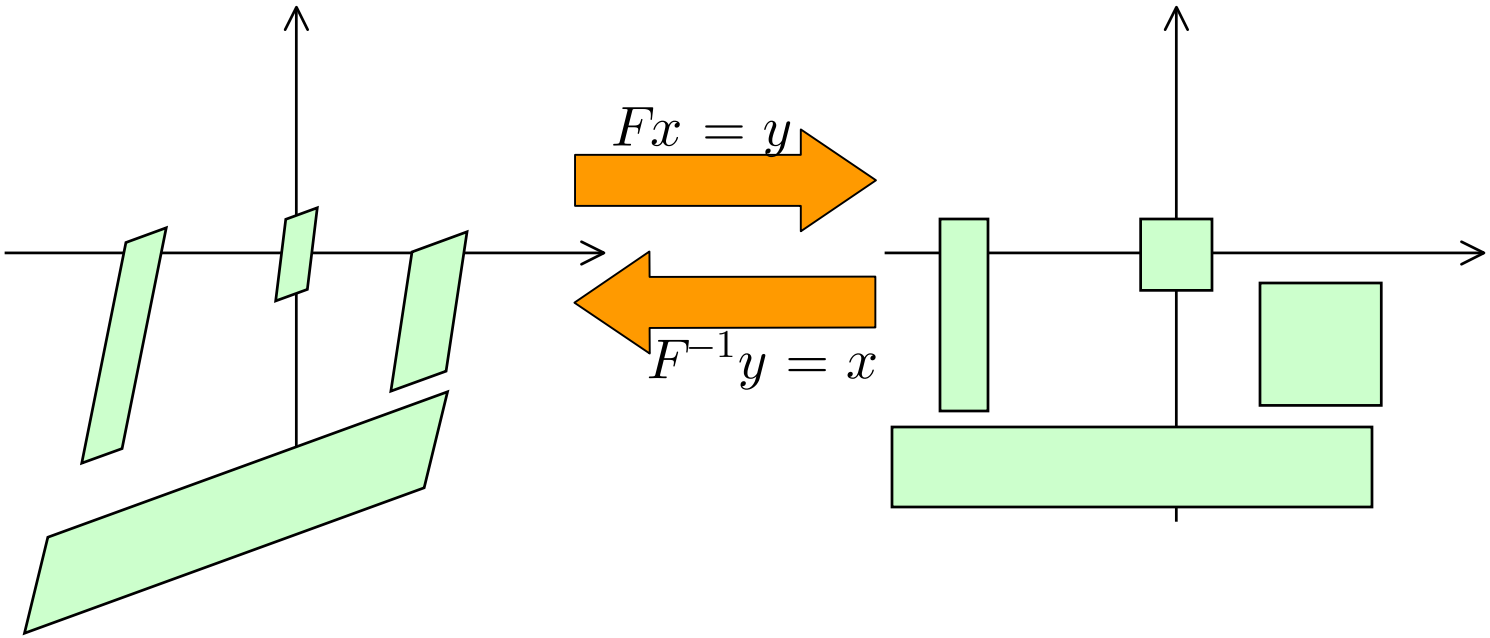
$$\begin{bmatrix} y \\ v \end{bmatrix} = U \begin{bmatrix} x \\ u \end{bmatrix}$$

$$U = \begin{bmatrix} F & 0_{n\times m} \\ G & H \end{bmatrix}$$

Brunovsky coordinates

$$\left.\begin{aligned} y_1(t+1) &= y_2(t) \\ y_2(t+1) &= y_3(t) \\ y_3(t+1) &= v_1(t) \end{aligned}\right\} = k_1$$

$$\left.\begin{aligned} y_4(t+1) &= y_5(t) \\ y_5(t+1) &= v_2(t) \end{aligned}\right\} = k_2$$



Brunovsky boolean algebra

Brunovksy Boolean Algebra

$f^T x \sim c_i \quad c_i \in Q, \quad \sim \in \{>, =, <\}$

$f^T \in rows(F)$

$y = Fx$

$x = F^{-1}y$

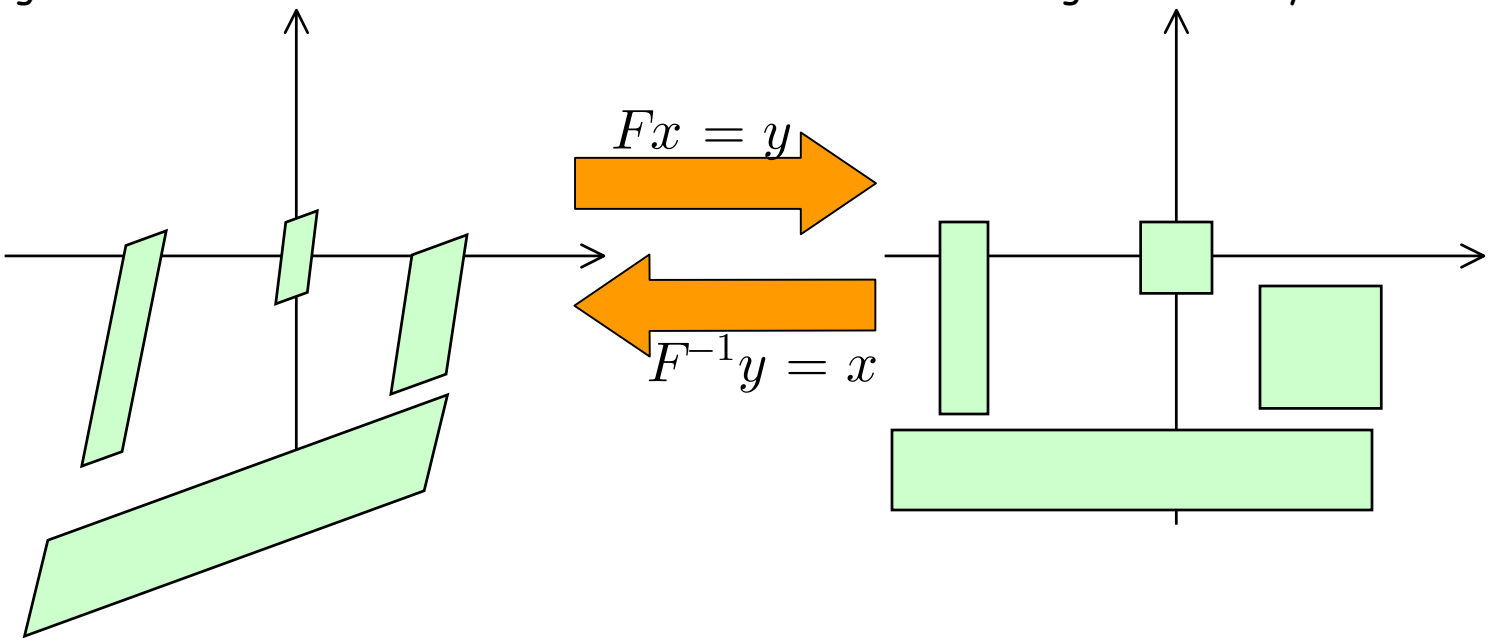
Algebra in Brunovsky coordinates

$w^T y_i \sim c_i \quad c_i \in Q, \quad \sim \in \{>, =, <\}$

$w \in \{ \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \pm \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, ..., \pm \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \}$

Subalgebra of semi-linear sets

Rectangular sets in y coordinates



Properties

Consider a discrete-time controllable linear system in Brunovsky normal form. Let A,B be any sets in the Brunovksy boolean algebra. Then Pre(A) belongs in the Brunovsky boolean algebra*. Furthermore

$$Pre(A \cap B) = Pre(A) \cap Pre(B)$$
$$Pre(\overline{A}) = \overline{Pre(A)}$$

Therefore the Brunovsky boolean algebra

$$R_S = (P, Pre, And, Diff, Empty)$$

is a region algebra.

*P. Tabuada, and G.J. Pappas, **Finite bisimulations of controllable linear systems**, IEEE Conference on Decision and Control, 2003.

Temporal logic synthesis

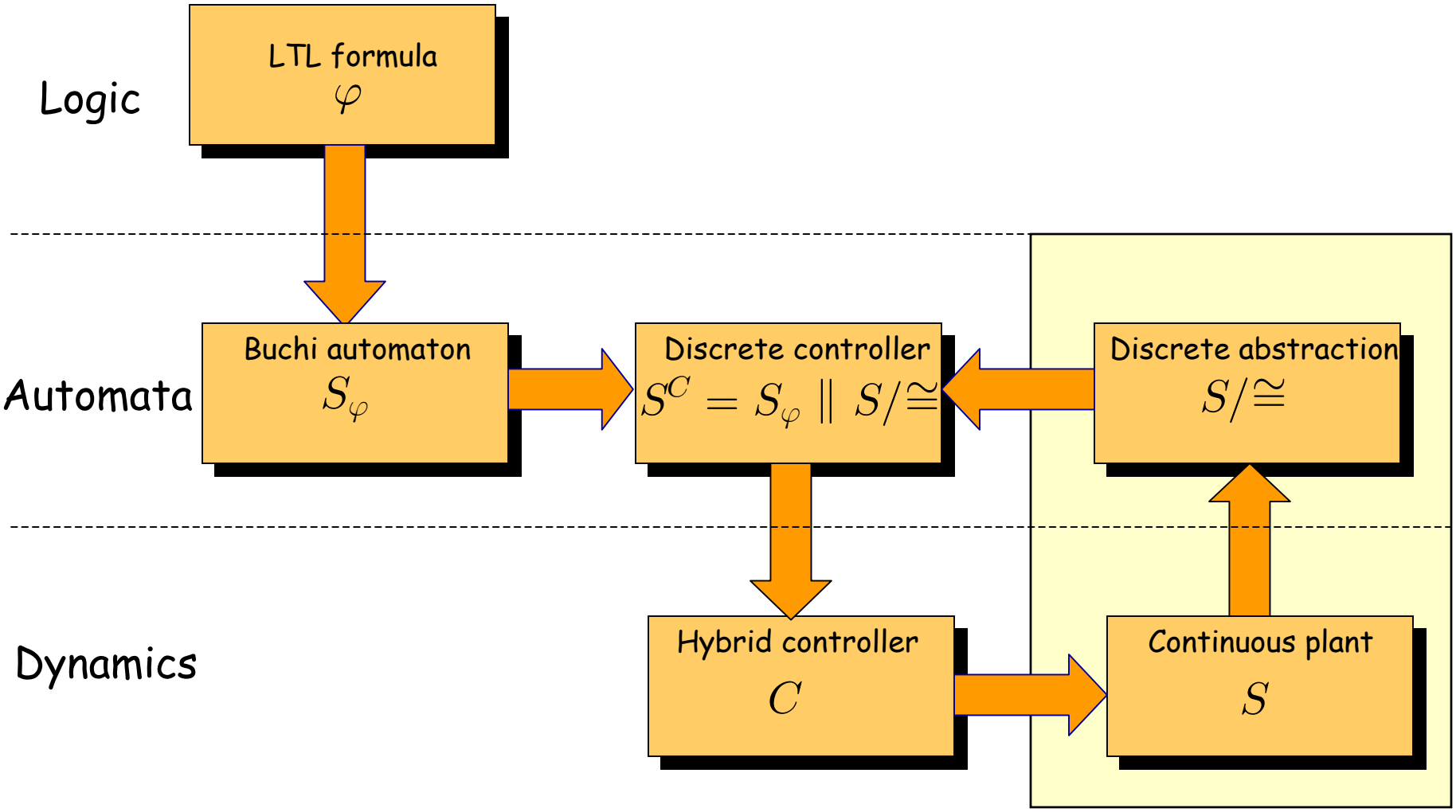
Consider discrete-time controllable systems $x(t+1)=Ax(t)+Bu(t)$
Then*

1. Consider a finite partition of the state space where definable in the Brunovsky boolean algebra. Then a finite bisimulation always exists and can be computed.

2. Consider an LTL formula where atomic propositions denote sets in the Brunovsky boolean algebra. Then LTL controller synthesis is decidable.

*P. Tabuada, and G.J. Pappas, **Finite bisimulations of controllable linear systems**, IEEE Conference on Decision and Control, 2003.

LTL controller synthesis



Set representation

Set Representation in DNF

Each clause is represented with "Interval" matrices (IM)

i.e. $S = \{x \text{ in } R^2: (-1 \le x_1 < 1/2 \wedge x_2 < 2) \quad (x_2 \ge 2)\}$

x_1 x_2

$$\begin{bmatrix} -1 & \le & 2 & < \\ 1/2 & < & -\inf & < \end{bmatrix}$$

$$\begin{bmatrix} \inf & < & \inf & < \\ -\inf & < & 2 & \le \end{bmatrix}$$

"Cheap" operations (linear in the number of variables):

- Intersection of 2 clauses
- Check emptiness of a clause
- Computing Pre
- Checking inclusion of 2 clauses

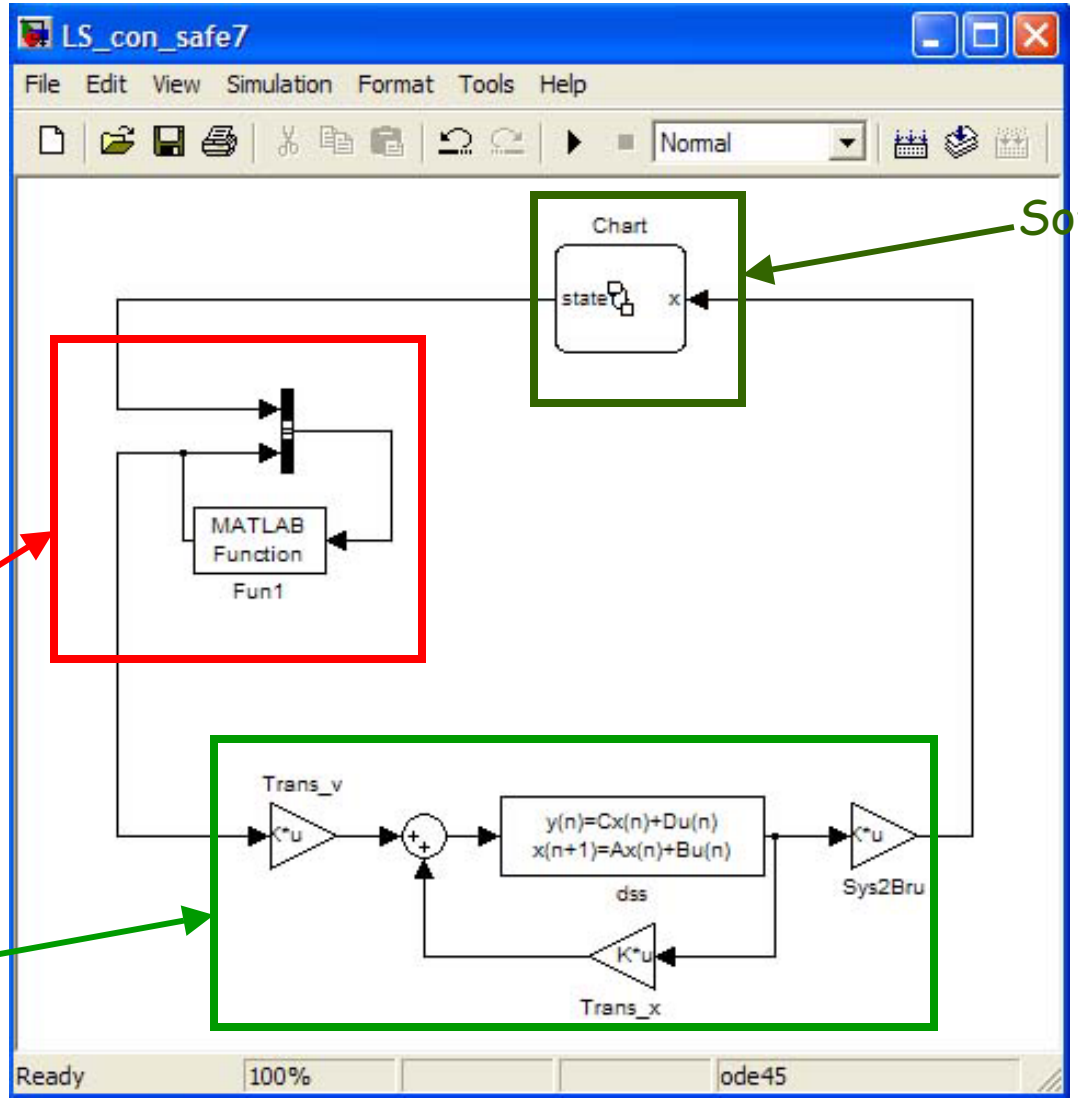
"Expensive" operations:

- Complementation of a formula in DNF
- Intersection of DNF formulas

⇩

Try to avoid non-convex sets in the specifications

Closed loop system in Simulink



Software - A/D

D/A

Continuous System

Penn
University of Pennsylvania

Stateflow logic

D/A

Input Constraints:

State = 1

$u \in R$

State = 2

$|u| < 1/2$

State = 3

$|u| < 1/2$

State = 4

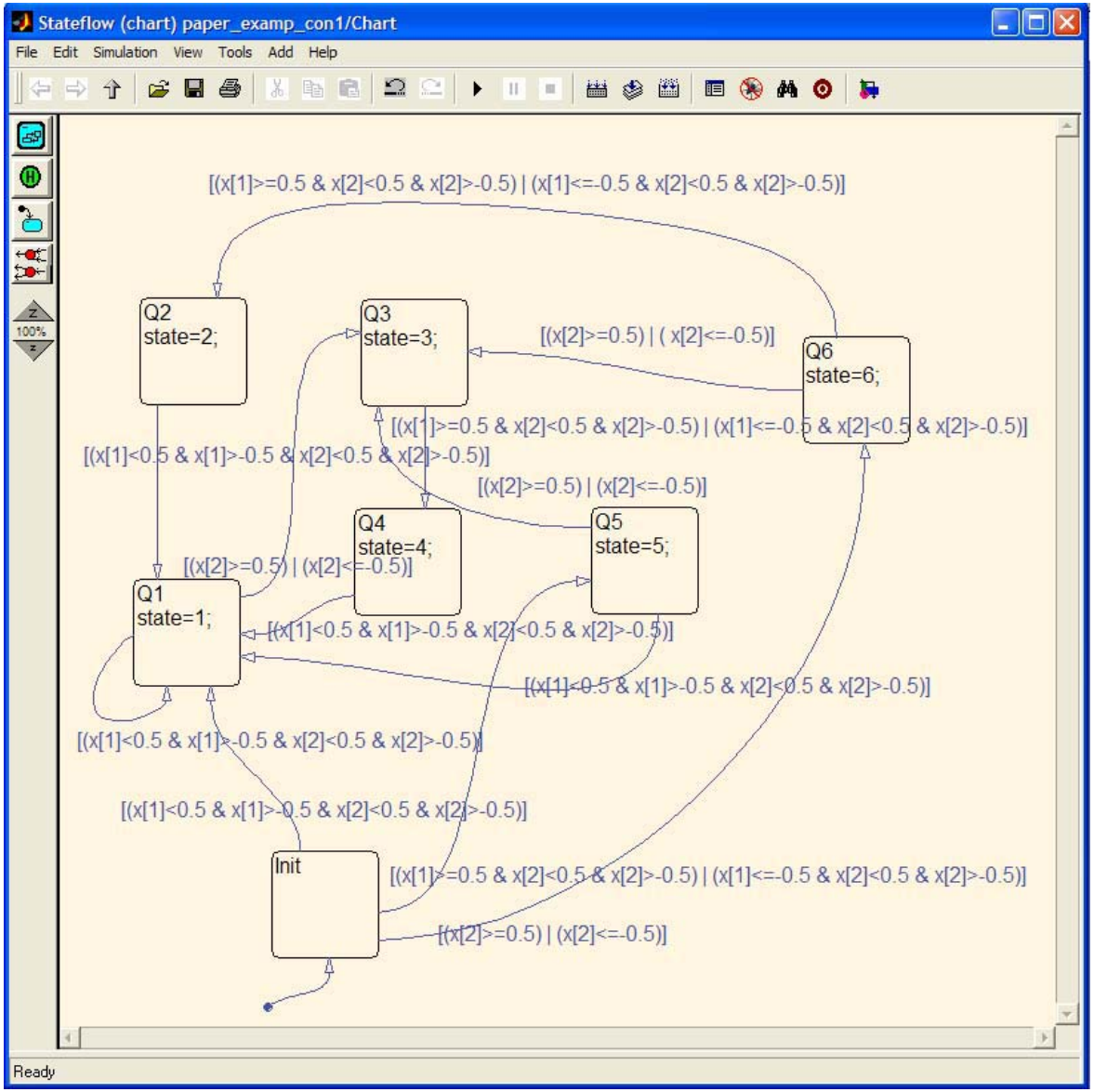
$|u| < 1/2$

State = 5

$u \in R$

State = 6

$u \in R$



Penn
University of Pennsylvania

Future challenges

Control on the fly…

Exploit parallel predicates

Get as close to the semi-linear sets as possible

Include input constraints and environmental disturbances

Include hybrid and switching dynamics

Discretize in time while preserving temporal logic formulas

Compositional controller synthesis

Acknowledgments

Postdocs

- Paulo Tabuada (off to UND)
- Herbert Tanner (off to UNM)

Ph.D Students

- Ali Ahmazadeh
- George Fainekos
- Hadas Kress Gazit
- Hakan Yazarel
- Michael Zavlanos

M.S. students

- Selcuk Bayraktar
- Pranav Srivastava

Collaborators

Rajeev Alur, Datta Godbole, Tom Henzinger, Ali Jadbabaie, John Koo, Vijay Kumar, Gerardo Lafferierre, Insup Lee, John Lygeros, Shankar Sastry, Omid Shakernia, Claire Tomlin, Sergio Yovine

Support

- NSF Career
- NSF ITR (2)
- NSF EHS
- ARO MURI
- DARPA MoBIES
- Honeywell, Boeing

Penn
University of Pennsylvania