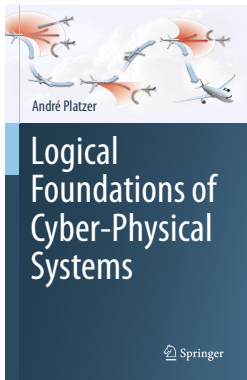


Logical Foundations of Cyber-Physical Systems

01: Cyber-Physical Systems: Overview



André Platzer

Karlsruhe Institute of Technology
Department of Informatics

Computer Science Department
Carnegie Mellon University

- 1 **CPS: Introduction**
 - Hybrid Systems & Cyber-Physical Systems
 - Applications
 - Robot Labs
- 2 **Course: Logical Foundations of Cyber-Physical Systems**
 - Educational Approach
 - Objectives
 - Outline
 - CPS V&V Grand Prix
 - Assessment
 - Resources
- 3 **Summary**

1 CPS: Introduction

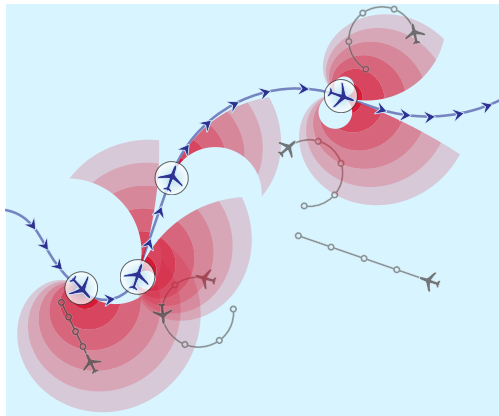
- Hybrid Systems & Cyber-Physical Systems
- Applications
- Robot Labs

2 Course: Logical Foundations of Cyber-Physical Systems

- Educational Approach
- Objectives
- Outline
- CPS V&V Grand Prix
- Assessment
- Resources

3 Summary

Which control decisions are safe for aircraft collision avoidance?



Cyber-Physical Systems

CPSs combine cyber capabilities with physical capabilities to solve problems that neither part could solve alone.

Prospects: Safe & Efficient

Driver assistance
Autonomous cars

Pilot decision support
Autopilots / UAVs

Train protection
Robots near humans



Prerequisite: CPSs need to be safe

How do we make sure CPSs make the world a better place?

Can you trust a computer to control physics?

Can you trust a computer to control physics?

- 1 Depends on how it has been programmed
- 2 And on what will happen if it malfunctions

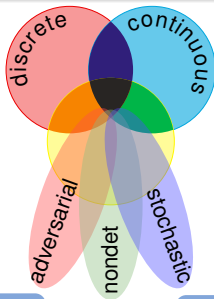
Rationale

- 1 Safety guarantees require analytic foundations.
- 2 A common foundational core helps all application domains.
- 3 Foundations revolutionized digital computer science & our society.
- 4 Need even stronger foundations when software reaches out into our physical world.

CPSs deserve proofs as safety evidence!

CPS Dynamics

CPS are characterized by multiple facets of dynamical systems.



CPS Compositions

CPS combines multiple simple dynamical effects.

Descriptive simplification

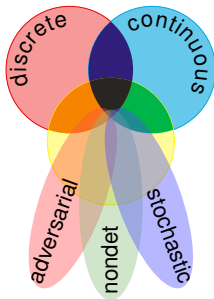
Tame Parts

Exploiting compositionality tames CPS complexity.

Analytic simplification

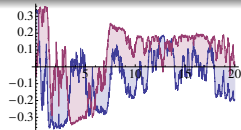
hybrid systems

HS = discrete + ODE



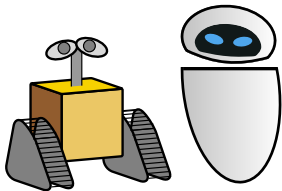
stochastic hybrid sys.

SHS = HS + stochastic



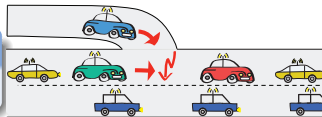
hybrid games

HG = HS + adversary



distributed hybrid sys.

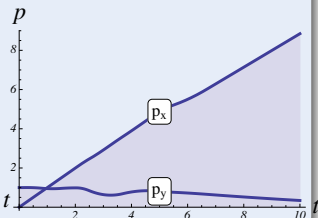
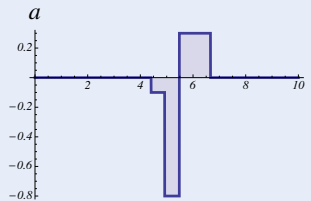
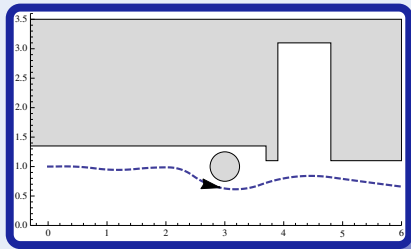
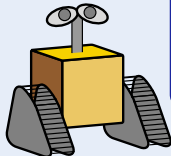
DHS = HS + distributed



Challenge (CPS)

Fixed rule describing state evolution with both

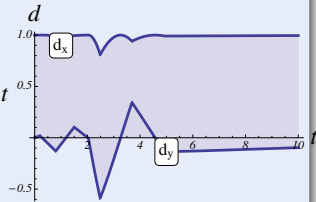
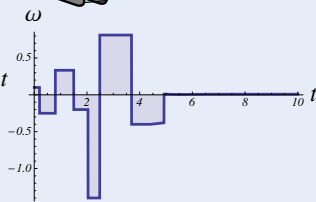
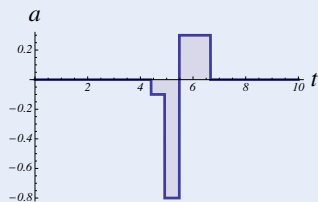
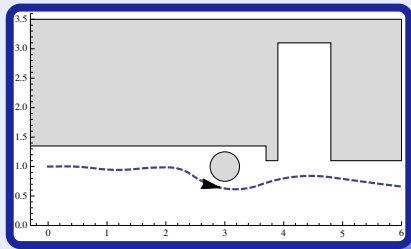
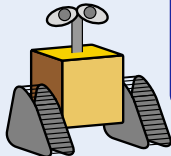
- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)



Challenge (CPS)

Fixed rule describing state evolution with both

- Discrete dynamics (control decisions)
- Continuous dynamics (differential equations)



Mathematical model for complex physical systems:

Definition (Hybrid Systems)

Systems with interacting discrete and continuous dynamics

Technical characteristics:

Definition (Cyber-Physical Systems)

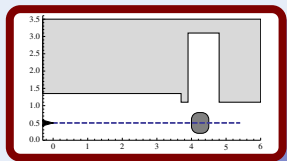
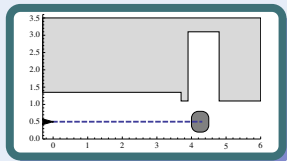
(Distributed networks of) computerized control for physical system
Communication, computation, and control for physics

What CPSs are around us?

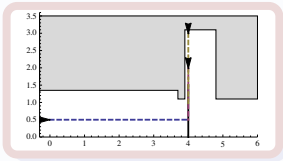
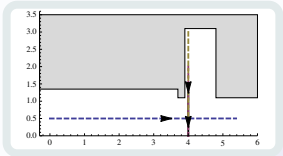
What CPSs will be around us in the future?

Which CPSs do we trust with our lives?

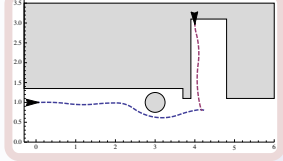
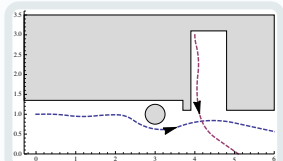
1: Charging Station



2: Follow the Leader

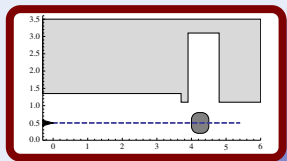
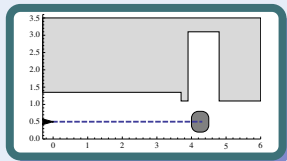


4: Obstacles

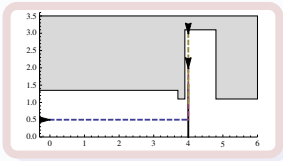
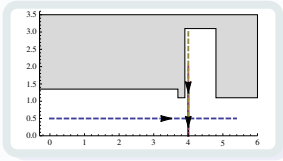


- ✓ Design, model
- ✓ Verify

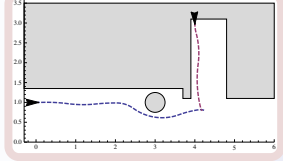
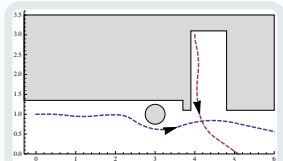
1: Charging Station



2: Follow the Leader

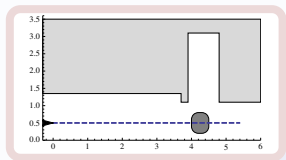
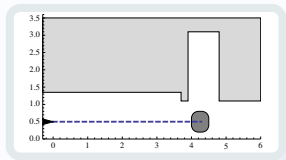


4: Obstacles

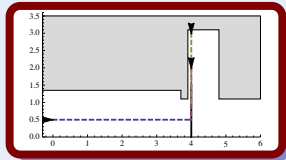
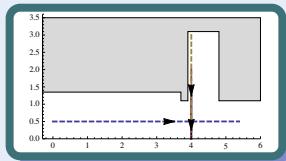


- ✓ Design, model
- ✓ Verify

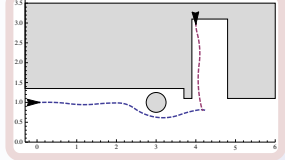
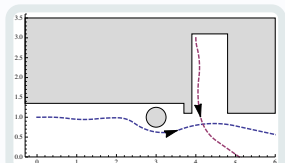
1: Charging Station



2: Follow the Leader

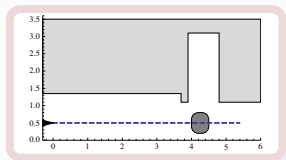
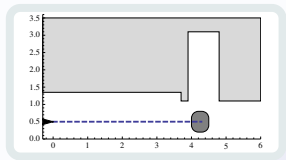


4: Obstacles

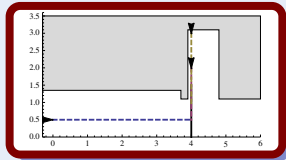
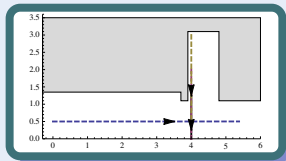


- ✓ Design, model
- ✓ Verify

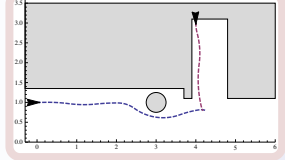
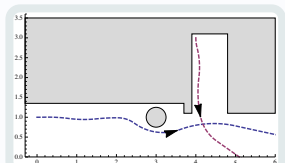
1: Charging Station



2: Follow the Leader

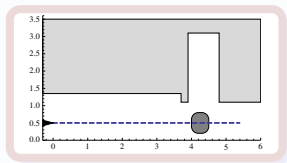
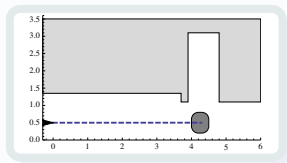


4: Obstacles

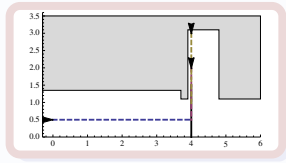
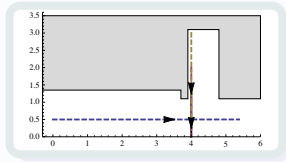


- ✓ Design, model
- ✓ Verify

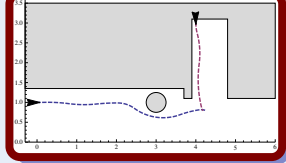
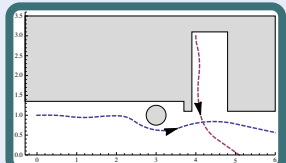
1: Charging Station



2: Follow the Leader

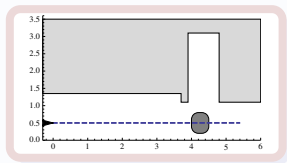
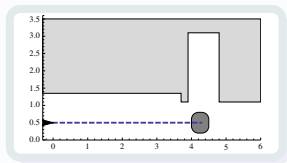


4: Obstacles

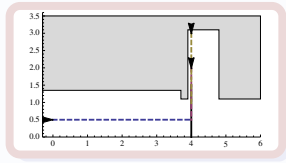
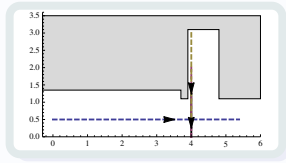


- ✓ Design, model
- ✓ Verify

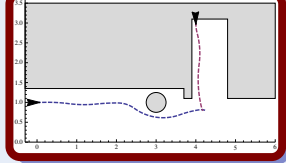
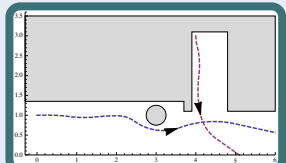
1: Charging Station



2: Follow the Leader

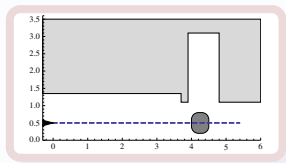
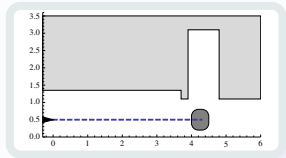


4: Obstacles

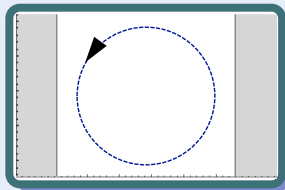


- ✓ Design, model
- ✓ Verify

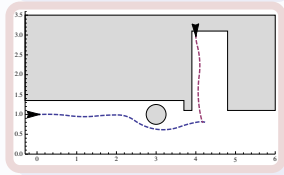
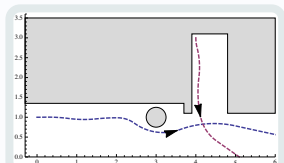
1: Charging Station



3: Racetrack



4: Obstacles

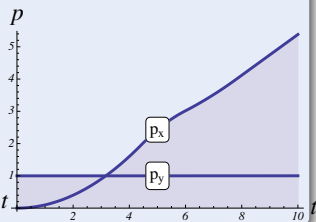
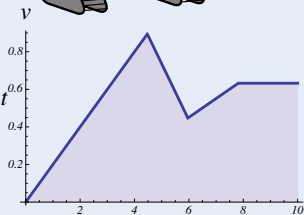
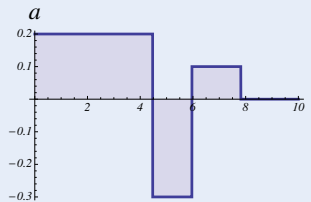
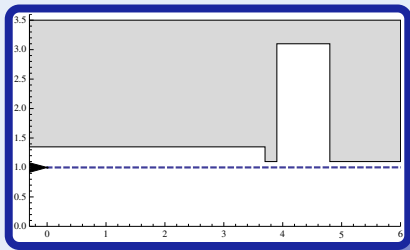
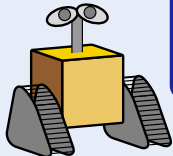


- ✓ Design, model
- ✓ Verify

Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

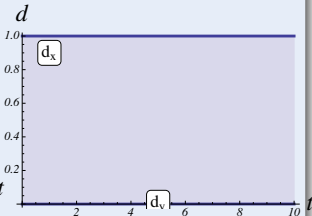
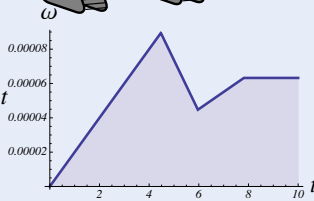
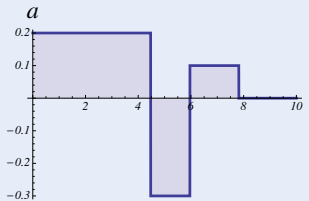
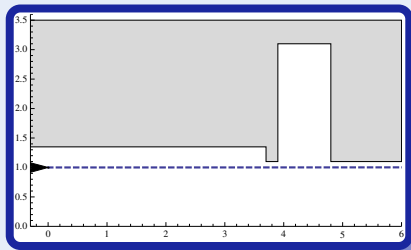
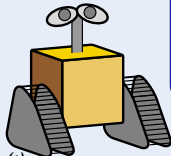
- Accelerate / brake (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

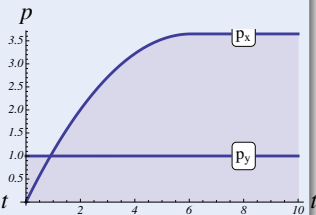
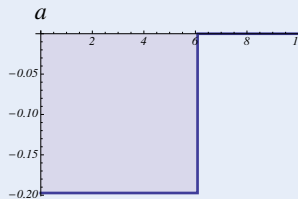
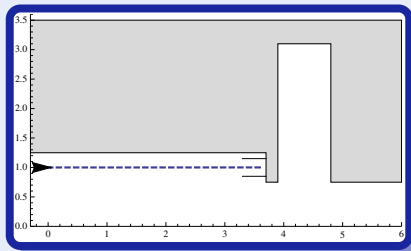
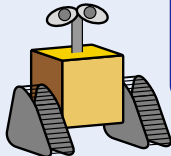
- Accelerate / brake (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

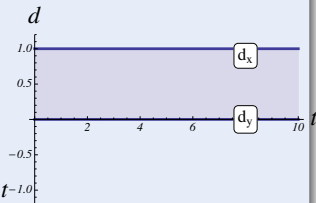
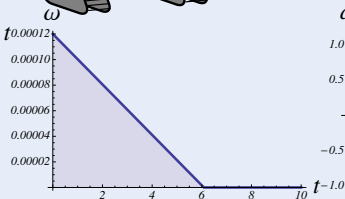
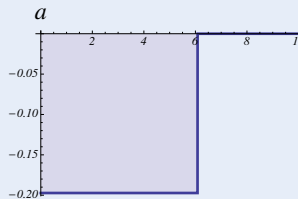
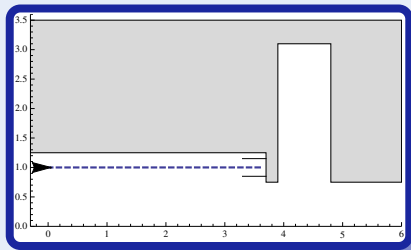
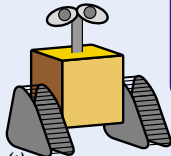
- Accelerate / brake / stop (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

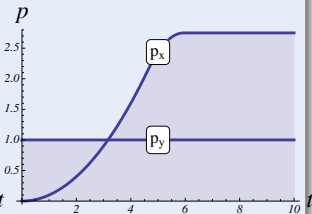
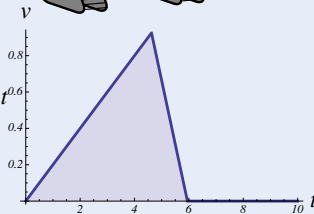
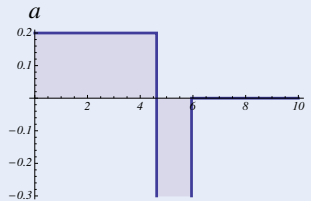
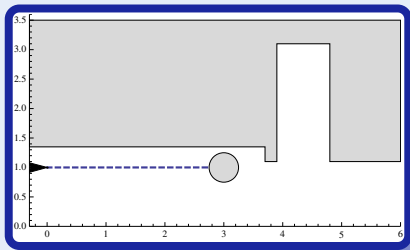
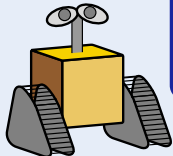
- Accelerate / brake / stop (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

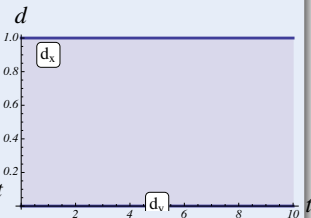
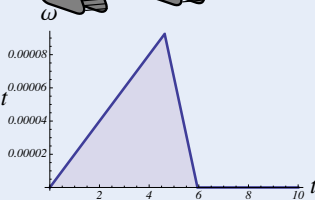
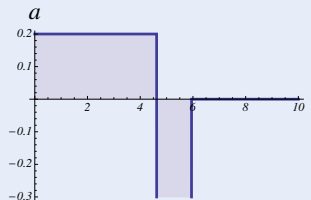
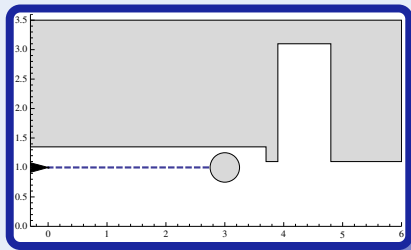
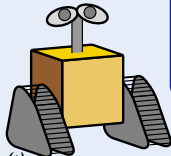
- Accelerate / brake (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

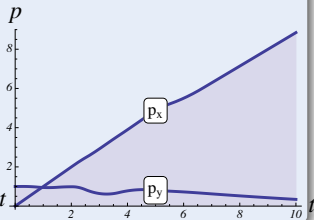
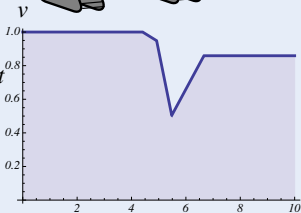
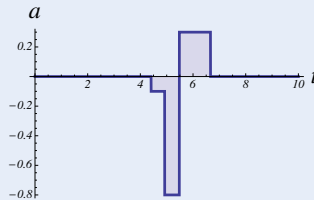
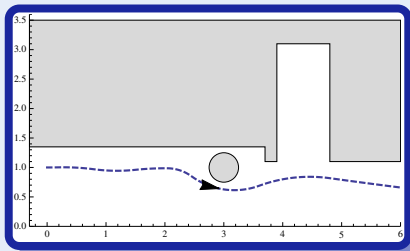
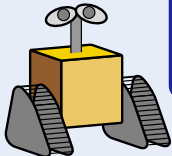
- Accelerate / brake (discrete dynamics)
- 1D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

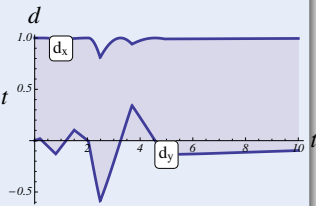
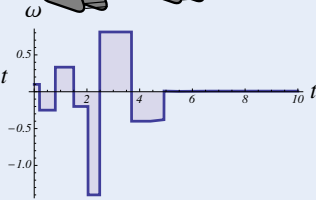
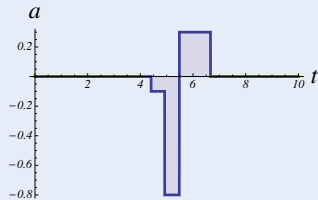
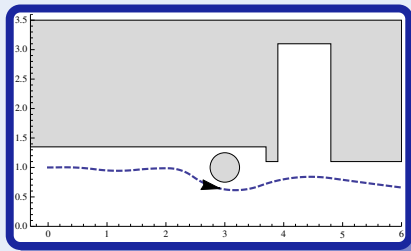
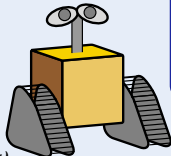
- Accel / brake / steer (discrete dynamics)
- 2D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

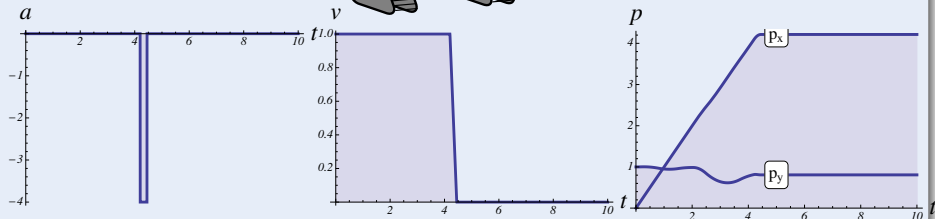
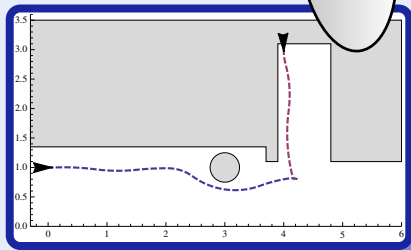
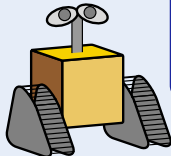
- Accel / brake / steer (discrete dynamics)
- 2D motion (continuous dynamics)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

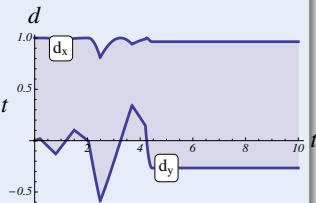
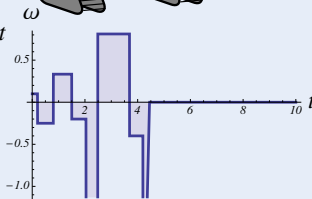
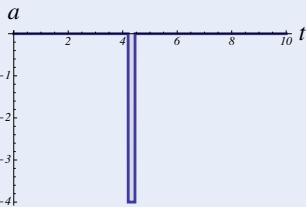
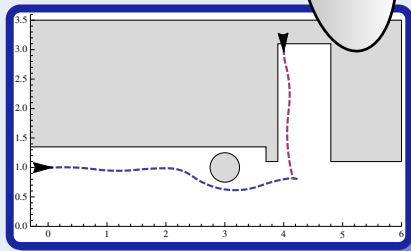
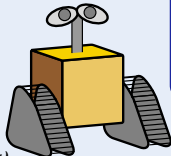
- Dynamic obstacles (other agents)
- Avoid collisions (define safety)



Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

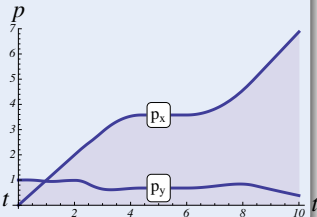
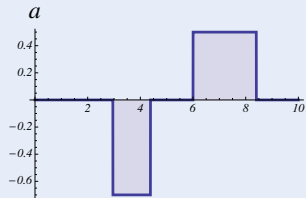
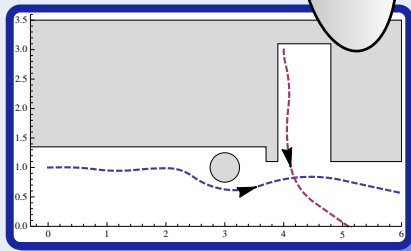
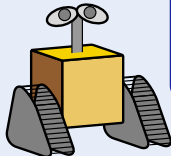
- Dynamic obstacles (other agents)
- Avoid collisions (define safety)



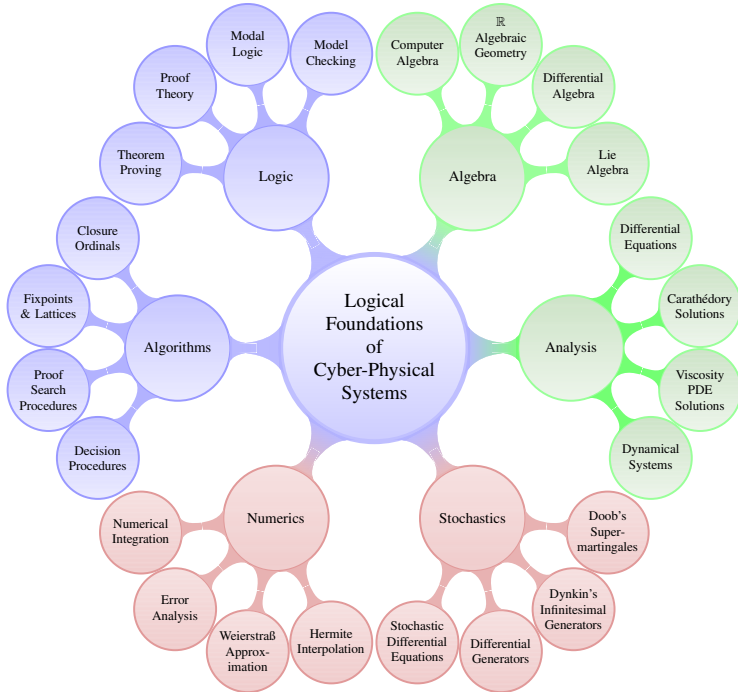
Challenge (Hybrid Systems)

Design & verify controller for a robot avoiding obstacles

- Control robot (respect delays)
- Environment interaction (obstacles, agents, uncertainty)



- 1 CPS: Introduction
 - Hybrid Systems & Cyber-Physical Systems
 - Applications
 - Robot Labs
- 2 **Course: Logical Foundations of Cyber-Physical Systems**
 - Educational Approach
 - Objectives
 - Outline
 - CPS V&V Grand Prix
 - Assessment
 - Resources
- 3 Summary



How to Teach Cyber-Physical Systems?

Onion Model

- 1 Going outside in
- 2 Unpeel layer by layer
- 3 Progress when all prereqs are covered
- 4 First study $CS \wedge math \wedge engineering$
- 5 Talk about CPS in the big finale

Scenic Tour Model

- 1 Start at the heart: CPS
- 2 Go on scenic expeditions into various directions
- 3 Explore the world around us as we find the need
- 4 Stay on CPS the whole time
- 5 Leverage CPS as the guiding motivation for understanding more about connected areas



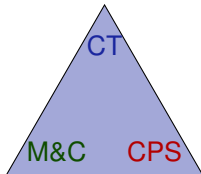
Logical scrutiny, formalization, and correctness proofs are critical for CPS!

- 1 CPSs are so easy to get wrong.
- 2 Retrofitting CPSs for safety is not possible.
- 3 These logical aspects are an integral part of CPS design.
- 4 Critical to your understanding of the intricate complexities of CPS.
- 5 Tame complexity by a simple programming language for core aspects.

- Foundations!
- Modeling & Control
 - ① Understand the core principles behind CPSs.
 - ② Develop models and controls.
 - ③ Identify the relevant dynamical aspects.
- Computational Thinking
 - ① Identify safety specifications and critical properties of CPSs.
 - ② Understand abstraction in system design.
 - ③ Express pre- and postconditions for CPS models.
 - ④ Use design-by-invariant.
 - ⑤ Reason rigorously about CPS models.
 - ⑥ Verify CPS models of appropriate scale.
- CPS Skills
 - ① Understand the semantics of a CPS model.
 - ② Develop an intuition for operational effects.
 - ③ Identify control constraints.
 - ④ Understand opportunities and challenges in CPS and verification.
- Byproducts
 - ① Well-motivated exposure to numerous math and science areas in action.



identify safety specifications for CPS
rigorous reasoning about CPS
understand abstraction & architectures
programming languages for CPS
verify CPS models at scale



cyber+physics models
core principles of CPS
relate discrete+continuous

semantics of CPS models
operational effects
identify control constraints
opportunities and challenges

I Part: Elementary Cyber-Physical Systems

2. Differential Equations & Domains
3. Choice & Control
4. Safety & Contracts
5. Dynamical Systems & Dynamic Axioms
6. Truth & Proof
7. Control Loops & Invariants
8. Events & Responses
9. Reactions & Delays

II Part: Differential Equations Analysis

10. Differential Equations & Differential Invariants
11. Differential Equations & Proofs
12. Ghosts & Differential Ghosts
13. Differential Invariants & Proof Theory

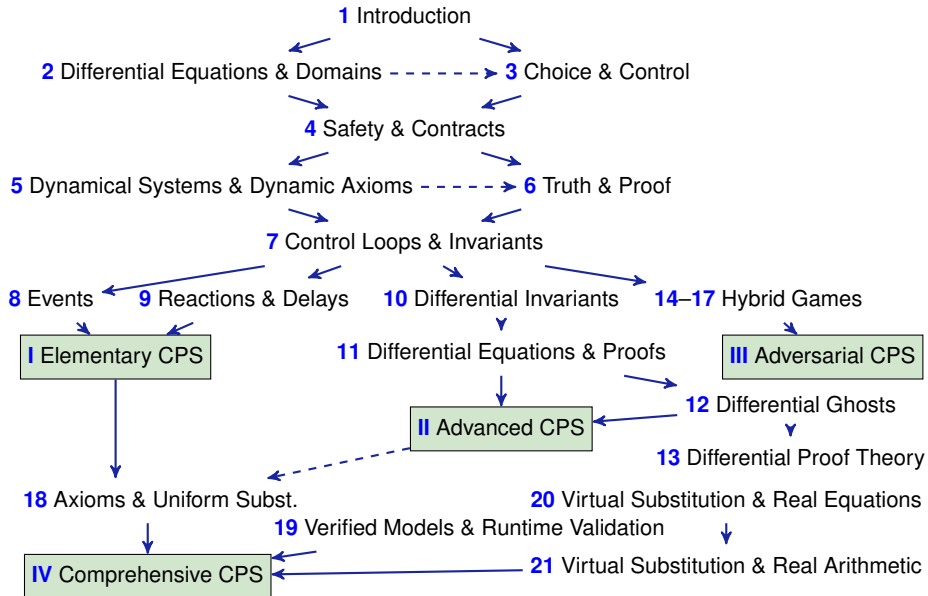
III Part: Adversarial Cyber-Physical Systems

- 17. Hybrid Systems & Hybrid Games

IV Part: Comprehensive CPS Correctness



Logical Foundations of Cyber-Physical Systems



Carnegie Mellon University May 5th, 2016



Prerequisites

Principles of Programming

if-then-else

Differential and Integral Calculus

x'

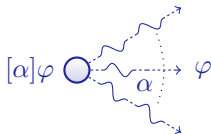
- You are expected to follow extra material in the textbook.
- Further reading and background material on the course web page
- Check Moodle and course web page periodically
<https://lfcps.org/course/lfcps.html>
- KeYmaera X: aXiomatic Tactical Theorem Prover for Hybrid Systems
<https://keymaeraX.org/>
- Read Course Policies [▶ Syllabus](#)
- Active learning quiz, 50% for bonus After each lecture
- Final exam [▶ Schedule](#)

- 1 CPS: Introduction
 - Hybrid Systems & Cyber-Physical Systems
 - Applications
 - Robot Labs
- 2 Course: Logical Foundations of Cyber-Physical Systems
 - Educational Approach
 - Objectives
 - Outline
 - CPS V&V Grand Prix
 - Assessment
 - Resources
- 3 Summary

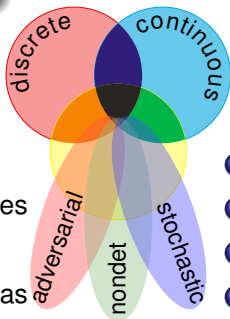
Logical foundations make a big difference for CPS, and vice versa

differential dynamic logic

$$dL = DL + HP$$



- Strong analytic foundations
- Practical reasoning advances
- Significant applications
- Catalyze many science areas



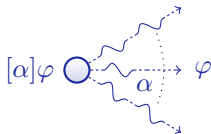
- 1 Multi-dynamical systems
- 2 Combine simple dynamics
- 3 Tame complexity
- 4 V&V cool challenges

Numerous wonders remain to be discovered

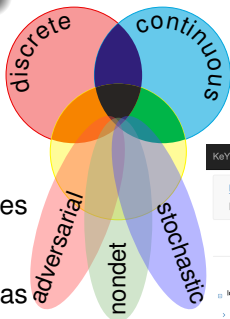
Logical foundations make a big difference for CPS, and vice versa

differential dynamic logic

$$dL = DL + HP$$



- Strong analytic foundations
- Practical reasoning advances
- Significant applications
- Catalyze many science areas



KeYmaera X

KeYmaera X Models Proofs Theme Help Power Refresh

Proof Auto Normalize Step back

Propositional Hybrid Programs Differential Equations

Base case 4 Use case 5 Induction step 6

$x \geq 0 \vdash [x := x + 1; u \{x' = v\}] x \geq 0$

$v \geq 0$

$x \geq 0, v \geq 0 \vdash [(x := x + 1; u \{x' = v\})^*] x \geq 0$

$x \geq 0 \wedge v \geq 0 \rightarrow [(x := x + 1; u \{x' = v \wedge true\})^*] x \geq 0$

$[u] [a \cup b] P \rightarrow [a] P \wedge [b] P$

Numerous wonders remain to be discovered



André Platzer.

Logical Foundations of Cyber-Physical Systems.

Springer, Cham, 2018.

[doi:10.1007/978-3-319-63588-0](https://doi.org/10.1007/978-3-319-63588-0).



André Platzer.

Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics.

Springer, Heidelberg, 2010.

[doi:10.1007/978-3-642-14509-4](https://doi.org/10.1007/978-3-642-14509-4).



André Platzer.

Logics of dynamical systems.

In *LICS*, pages 13–24, Los Alamitos, 2012. IEEE.

[doi:10.1109/LICS.2012.13](https://doi.org/10.1109/LICS.2012.13).



André Platzer.

Differential dynamic logic for hybrid systems.

J. Autom. Reas., 41(2):143–189, 2008.

[doi:10.1007/s10817-008-9103-8](https://doi.org/10.1007/s10817-008-9103-8).



André Platzer.

A complete uniform substitution calculus for differential dynamic logic.

J. Autom. Reas., 59(2):219–265, 2017.

doi:10.1007/s10817-016-9385-1.



André Platzer.

Logic & proofs for cyber-physical systems.

In Nicola Olivetti and Ashish Tiwari, editors, *IJCAR*, volume 9706 of *LNCS*, pages 15–21, Cham, 2016. Springer.

doi:10.1007/978-3-319-40229-1_3.



André Platzer.

Differential game logic.

ACM Trans. Comput. Log., 17(1):1:1–1:51, 2015.

doi:10.1145/2817824.



André Platzer.

Differential hybrid games.

ACM Trans. Comput. Log., 18(3):19:1–19:44, 2017.

doi:10.1145/3091123.