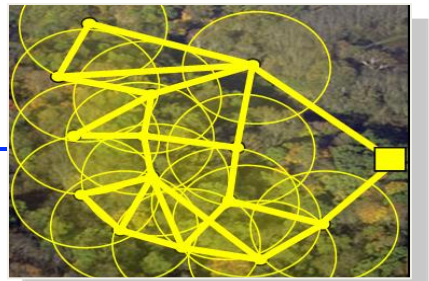
Cyber-Physical Systems Research Challenges

George J. Pappas
Joseph Moore Professor
NSF CPS Academic Executive Board
University of Pennsylvania
pappasg@seas.upenn.edu

NSF CPS AEB Membership

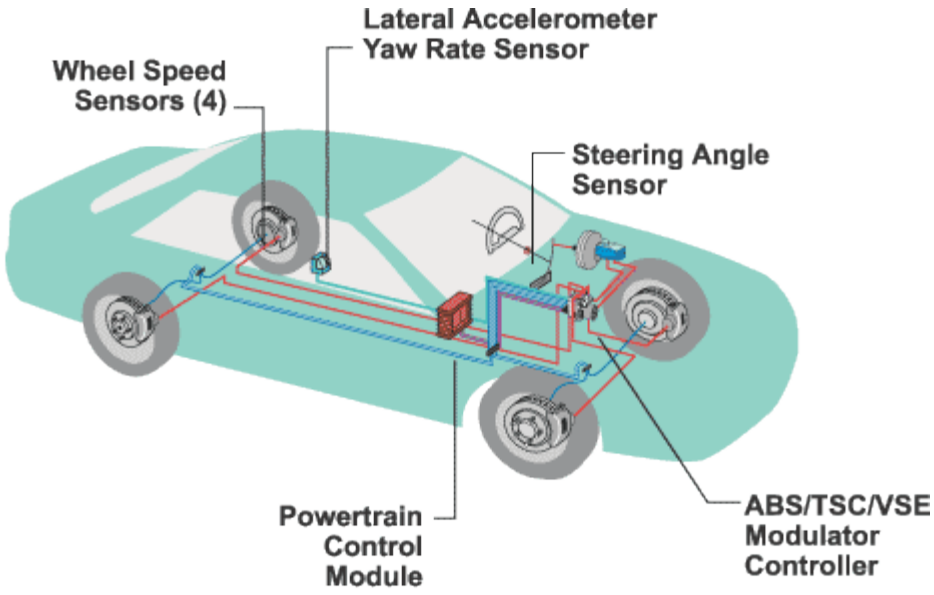
❑ George Pappas, U Penn	control
❑ Janos Sztipanovits, Vanderbilt	embedded software
❑ Edward Lee, UC Berkeley	education
❑ Eric Feron, Georgia Tech	avionics
❑ Jack Stankovic, Virginia	health monitoring/CCC
❑ Karl Hedrick, UC Berkeley	automotive
❑ Wei Zhao, Macau	real-time
❑ PR Kumar, UIUC	control, sensor nets
❑ John Baras, Maryland	systems engineering
❑ Vijay Kumar, U Penn	robotics
❑ John Mitchell, Stanford	security
❑ Anjan Bose, Washington State	smart grid
❑ Dimitri Mavris, Ga Tech	manufacturing
❑ Insup Lee, U Penn	real-time systems
❑ Bruce Krogh, CMU	control
❑ Nancy Lynch, MIT	distributed computing
❑ Claire Tomlin, UC Berkeley	hybrid systems
❑ Raj Rajkumar, CMU	real-time systems
❑ Daniela Rus, MIT	robotics

Cyber-physical systems*



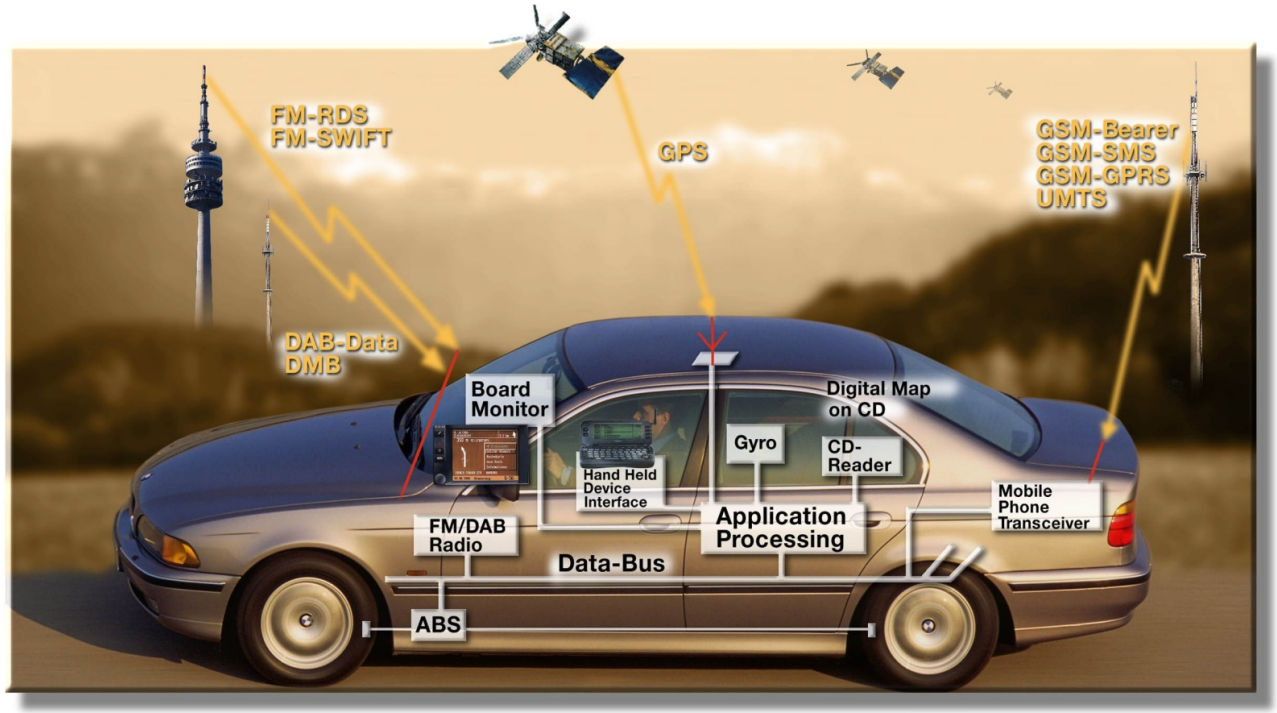
*Information systems interacting with physical systems, broadly

Automotive Systems



- Longitudinal dynamics : ABS (anti-lock brake system) and ASC (automatic stability control)
- Lateral dynamics : EDRC (engine drag reduction control) and CBC (corner braking control)
- DSC (dynamic stability control) is using all the above
- Also:
- Automatic gearboxes
- Anti-theft systems
- Multimedia systems
- Navigation systems
- etc

Automotive Systems

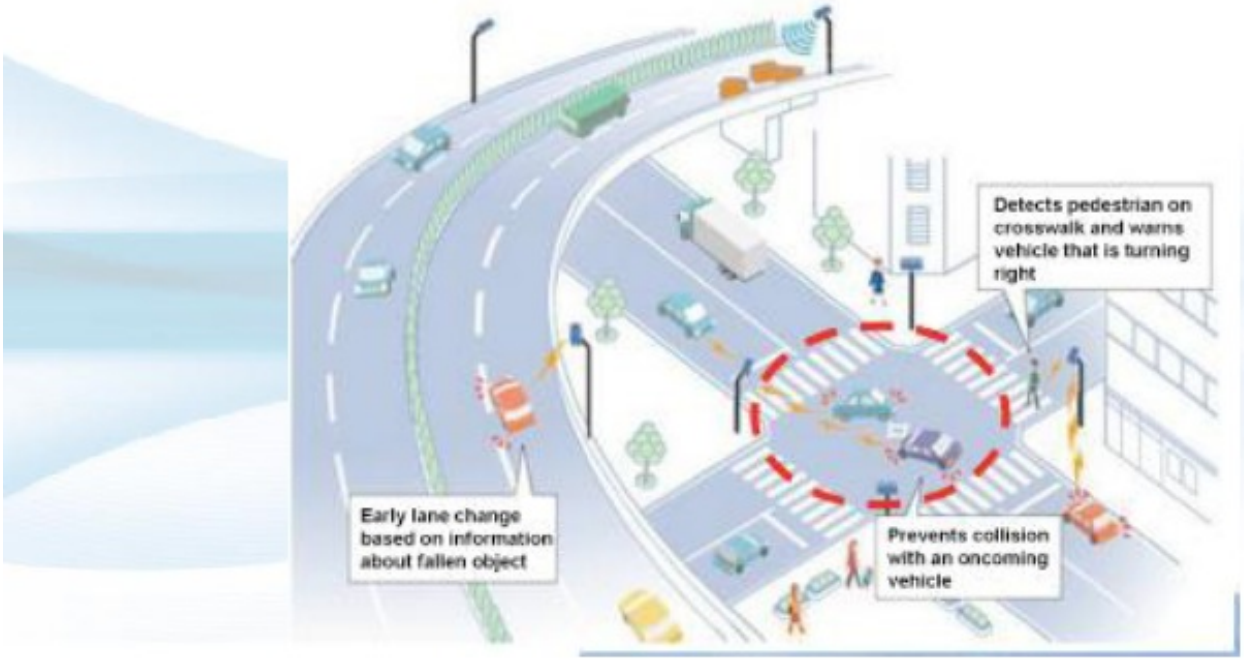


- 270 user interactive functions
- 67 embedded platforms / (5 data buses?)
- 65 MB of binary code.
- Next generation (2010):
  - ~ 1 GB of software, IP is being studied

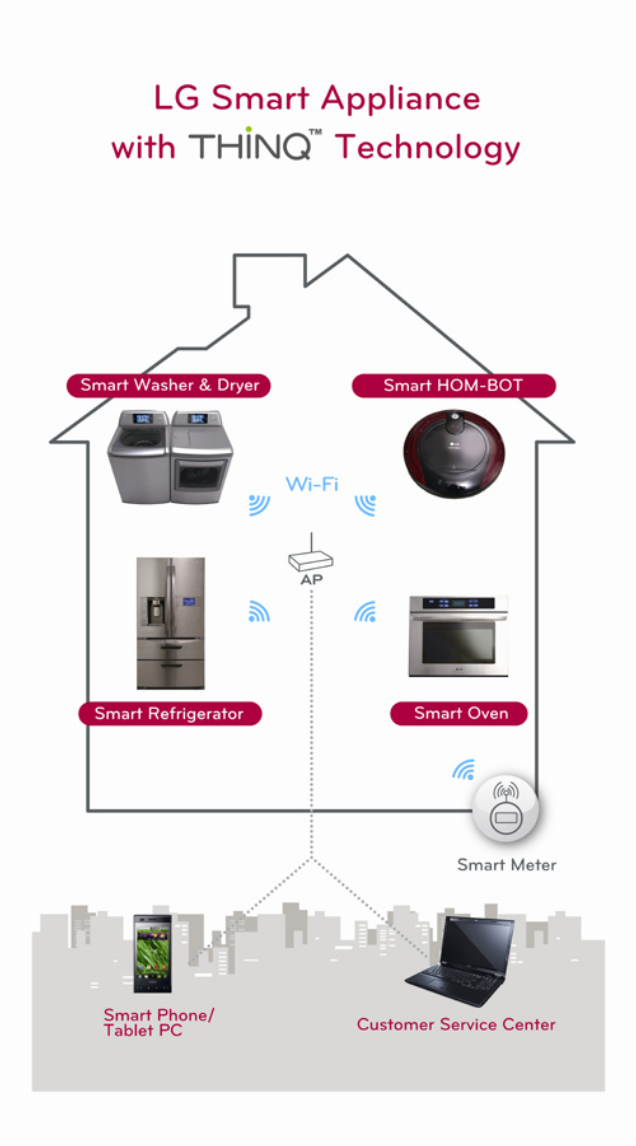
Smart Road Infrastructure

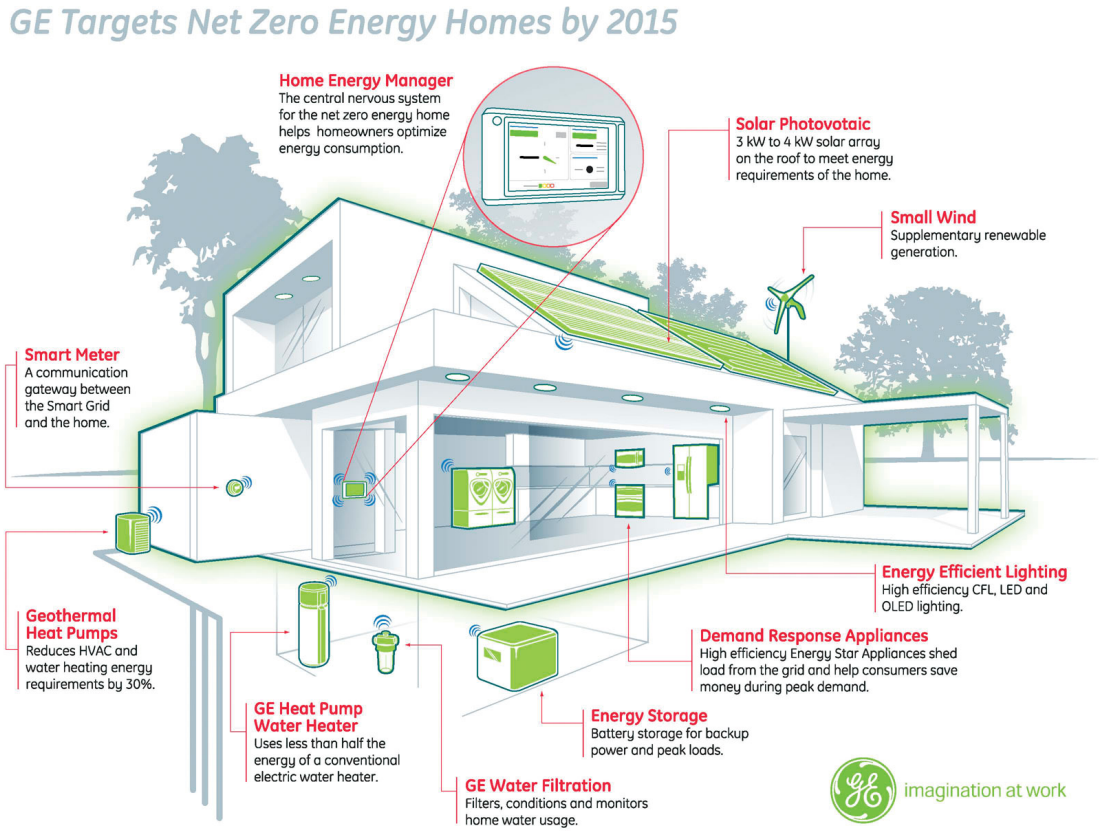
Methods Gap - Cyber-physical systems (e.g. collision avoidance)

- Heterogeneous modeling – hybrid dynamics, wireless networking, dynamic agent scenarios
- Abstractions and refinements for synthesis and analysis – hierarchical systems structures
- Component composability and consistency – systems integration, rapid development
- Verification (within and across) design and implementation



Embedded in : Home appliances





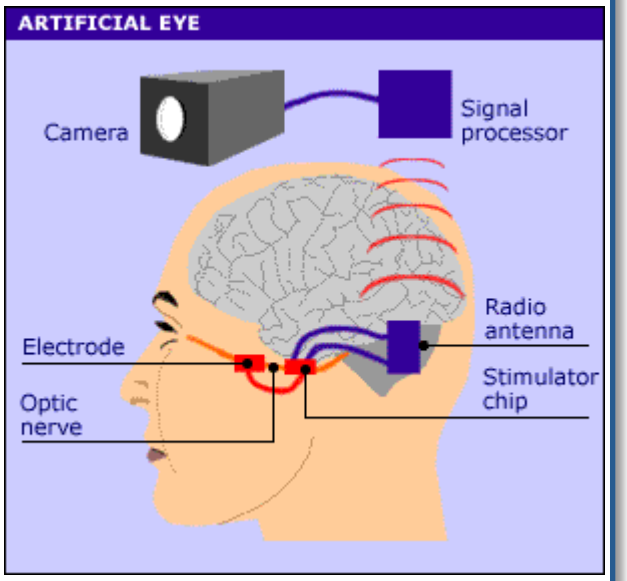
Medical devices

Vision : Doctor-on-a-chip, artificial eye

Operating room of the future & Digital hospital

Remote monitoring of elderly (body area networks)

Medical implants (artificial eyes, ears etc)



BBC news

NSF CPS AEB: What are we doing well?

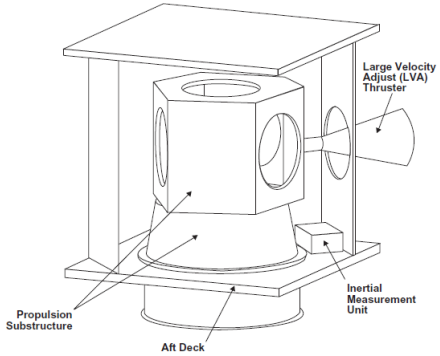
CPS is impacting numerous critical sectors of society

- ❑ Energy (SmartGrid, Microgrids, Green Buildings)
- ❑ Health (Medical devices)
- ❑ Automotive/Transportation (Mobile Millenium)
- ❑ Aerospace/Air traffic control (NextGen)
- ❑ Infrastructure monitoring (bridges, lakes)
- ❑ DoD (META)
- ❑ Robotics (medical robotics, mobile sensor networks)
- ❑ Next generation flexible manufacturing?

Tremendous potential for broad impact

CPS problems due to software-system interactions

1998: Near Earth Asteroid Rendezvous (NEAR)



2005: Demonstration of Autonomous Rendezvous Technology (DART)

1996: Ariane 5

1991: Patriot Missile Software Failure

Caltech's 2005 DARPA Grand Challenge Entry, Alice

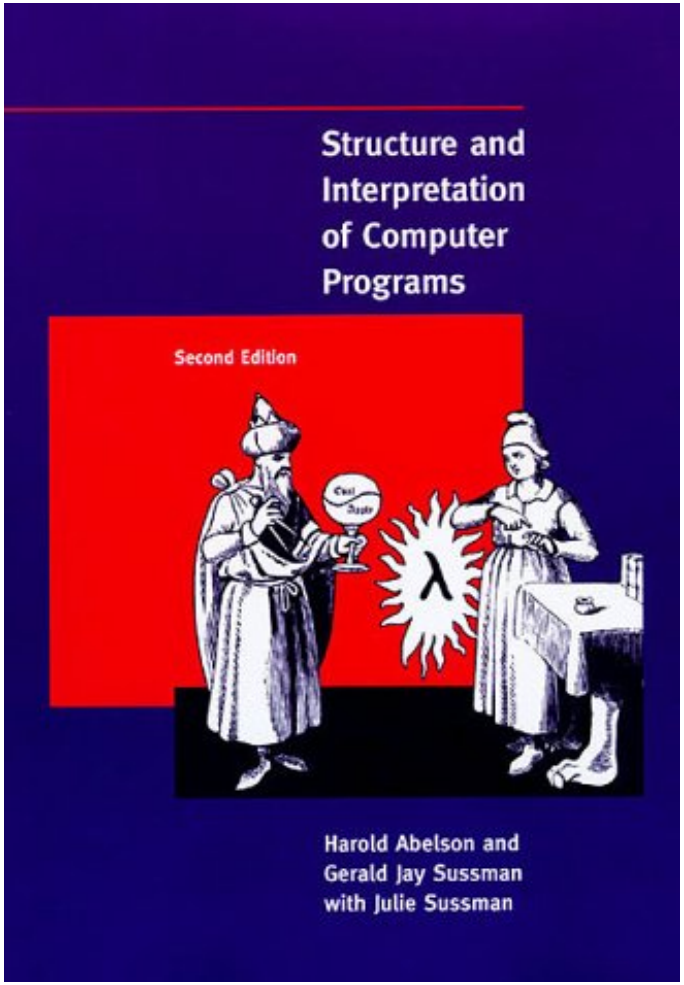
1997: Korean Air 747 in Guam

Northeast Blackout

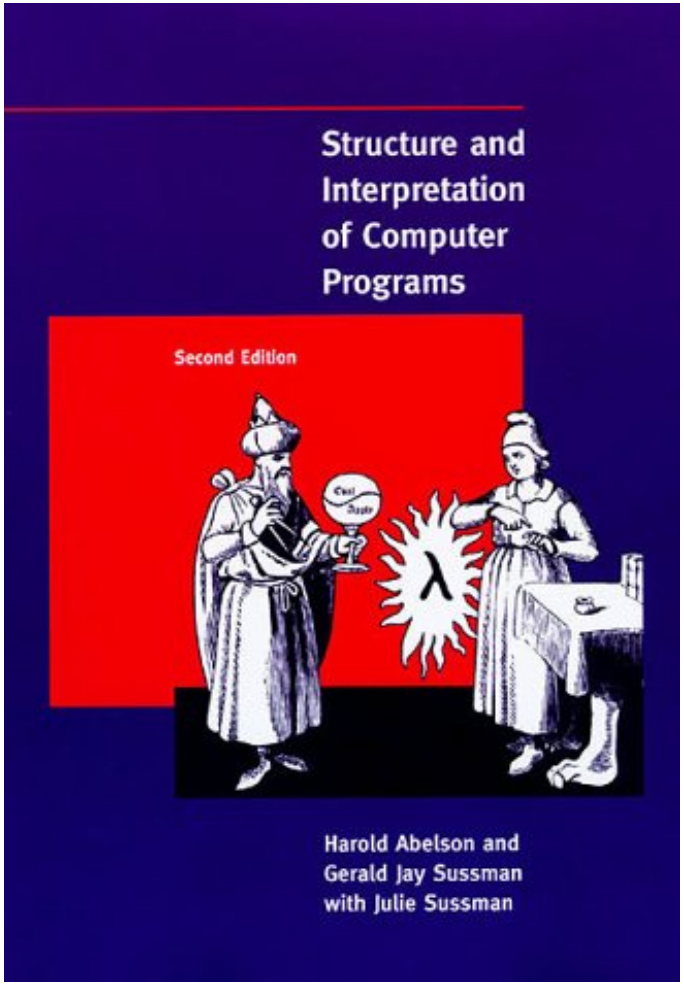
Penn
University of Pennsylvania

Research Challenges

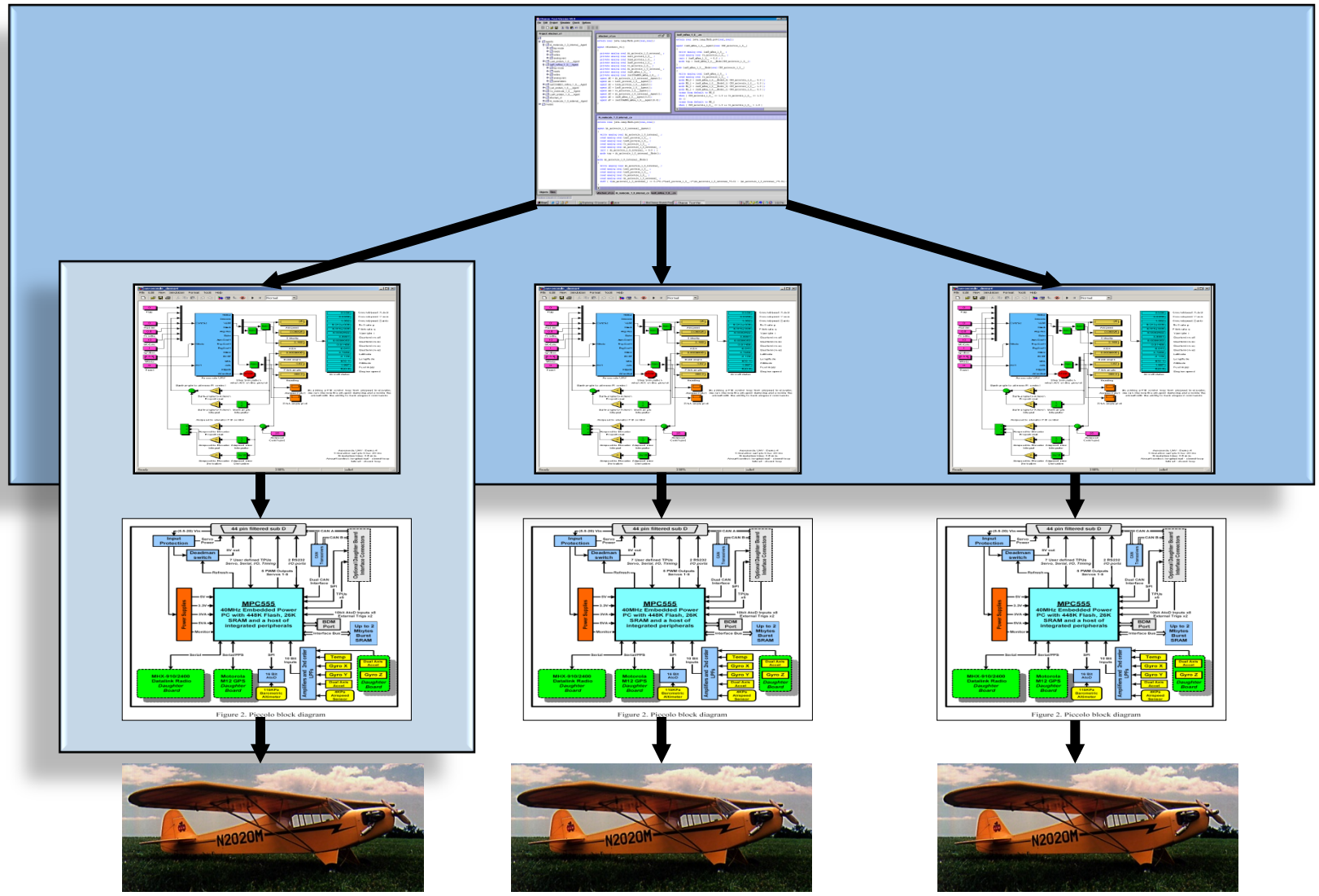
Part I : Modeling
Part II: Verification
Part III: Robustness
Part IV: Security



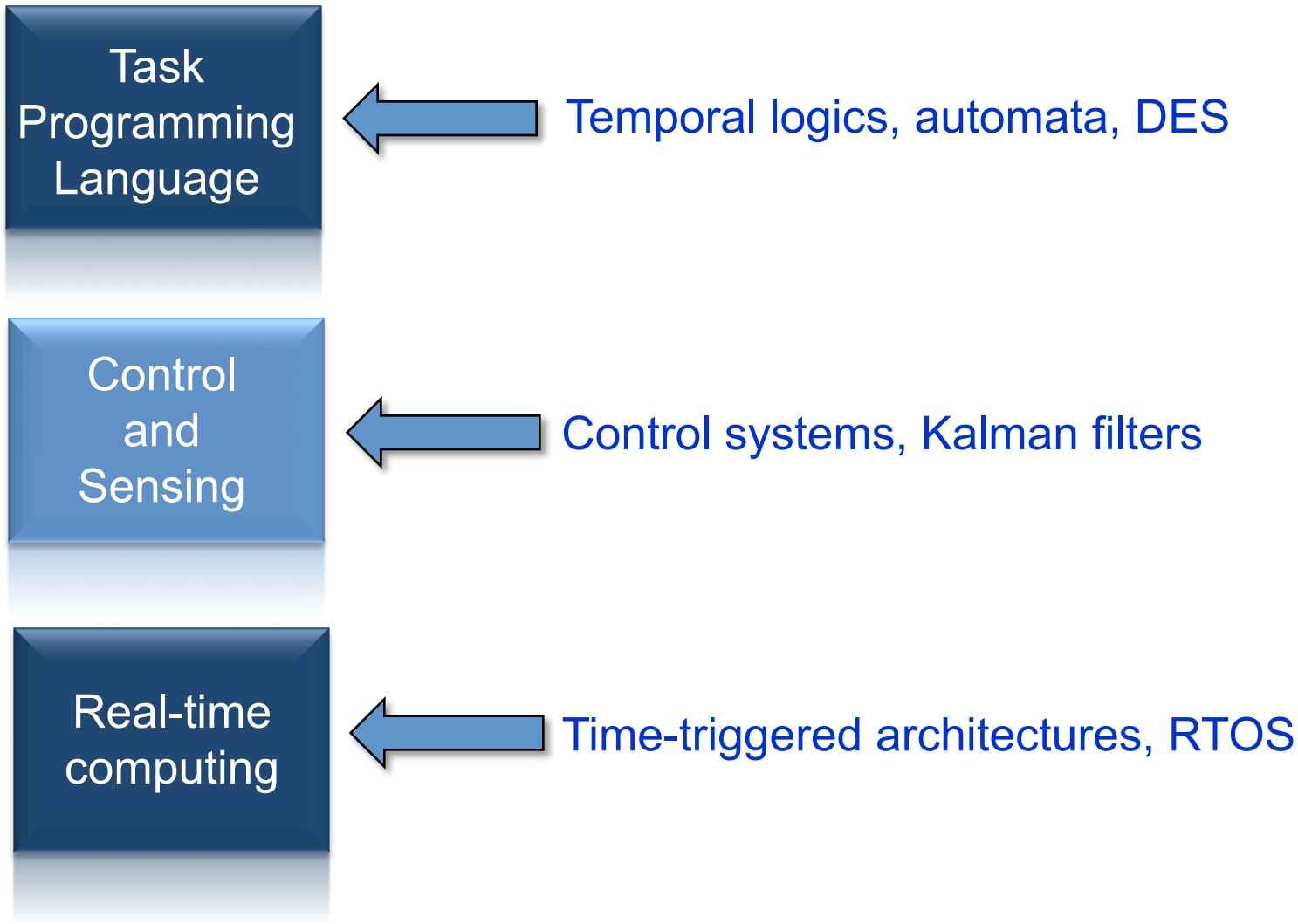
Part I : Modeling challenges



Control and computing hierarchies



Control and computing hierarchy



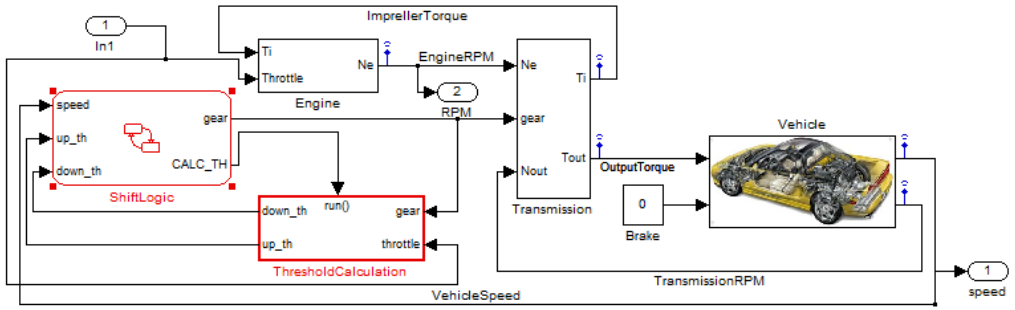
A CS view of the same hierarchy

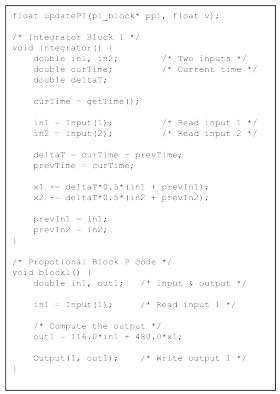
Task Programming Language

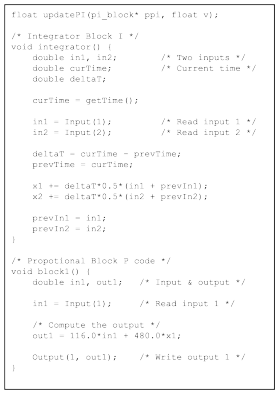
Control and Sensing

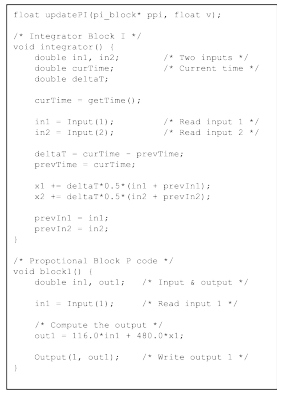
Real-time computing

$$\varphi = \left(\begin{array}{c} \neg s_{\text{Waldo}} \\ \wedge\Box\left(s_{\text{Waldo}} \rightarrow \bigcirc s_{\text{Waldo}}\right) \\ \wedge\Box\left(\left(\bigwedge_{i\in\{1,3,5,8\}} \neg r_i\right) \rightarrow \left(s_{\text{Waldo}} \leftrightarrow \bigcirc s_{\text{Waldo}}\right)\right) \\ \wedge\Box\Diamond(True) \end{array} \right) \rightarrow \left(\begin{array}{c} r_{12} \bigwedge_{i\neq 12} \neg r_i \\ \wedge\Box\left(r_1 \rightarrow \bigcirc r_1 \vee \bigcirc r_9\right) \\ \vdots \\ \wedge\Box\left(\bigvee_i \left(\bigcirc r_i \bigwedge_{j\neq i} \neg \bigcirc r_j\right)\right) \\ \bigwedge_{i\in\{1,3,5,8\}} \Box\left(\left(r_i \wedge \bigcirc s_{\text{Waldo}}\right) \rightarrow \bigcirc r_i\right) \\ \bigwedge_{i\in\{1,3,5,8\}} \Box\Diamond\left(r_i \vee s_{\text{Waldo}}\right) \end{array} \right)$$

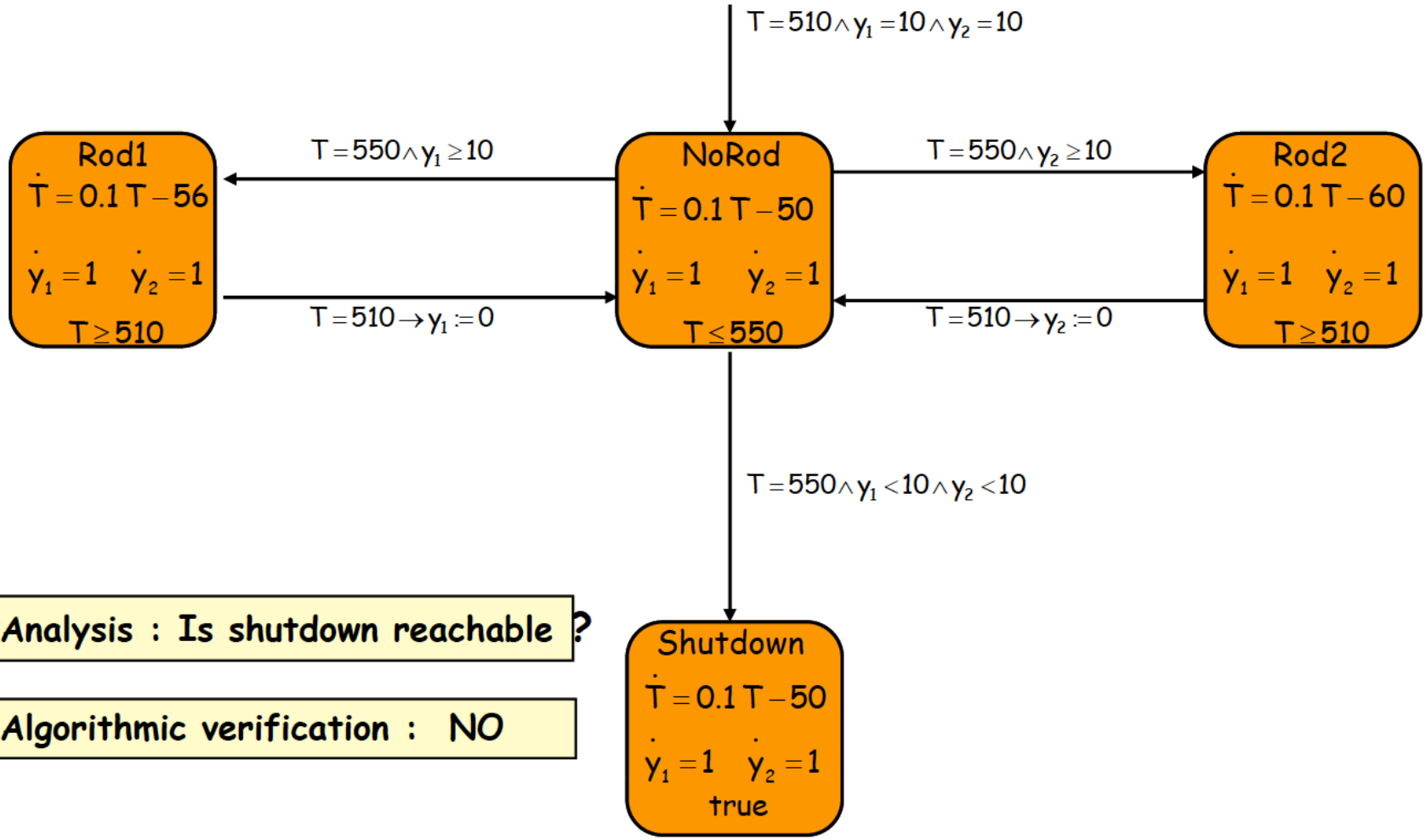








Hybrid model of nuclear reactor

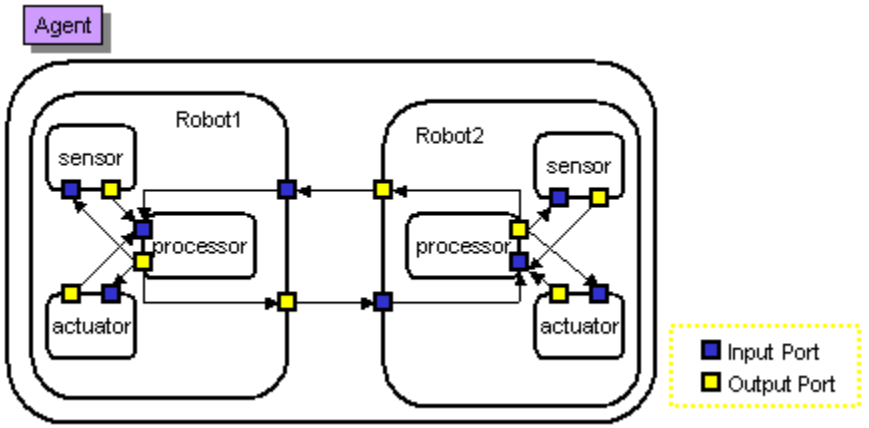


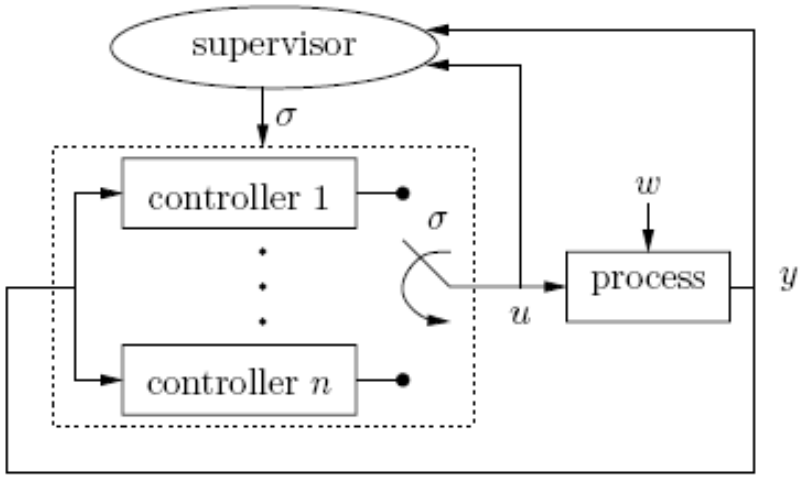
Analysis : Is shutdown reachable ?

Algorithmic verification : NO

A zoo of hybrid systems

- Hybrid Automata
- Hybrid Input-Output Automata
- Hybrid Petri Nets
- Simulink/Stateflow MATLAB models
- Supervisory control systems
- Switched systems
- Nonsmooth systems
- Piece-wise affine systems (PWA)
- Mixed Logical Dynamical
- Linear complementarity models





Major switching types

Time-triggered: Switching depends on time only

Switching and dynamics are decoupled

Switching times are known a priori

Switched systems more appropriate

Event-triggered: Switching also depends on state

Switching and dynamics are coupled

Switching times not known a priori

Hybrid automata more appropriate

Similarly there is a large variety of concurrency/synchronization types

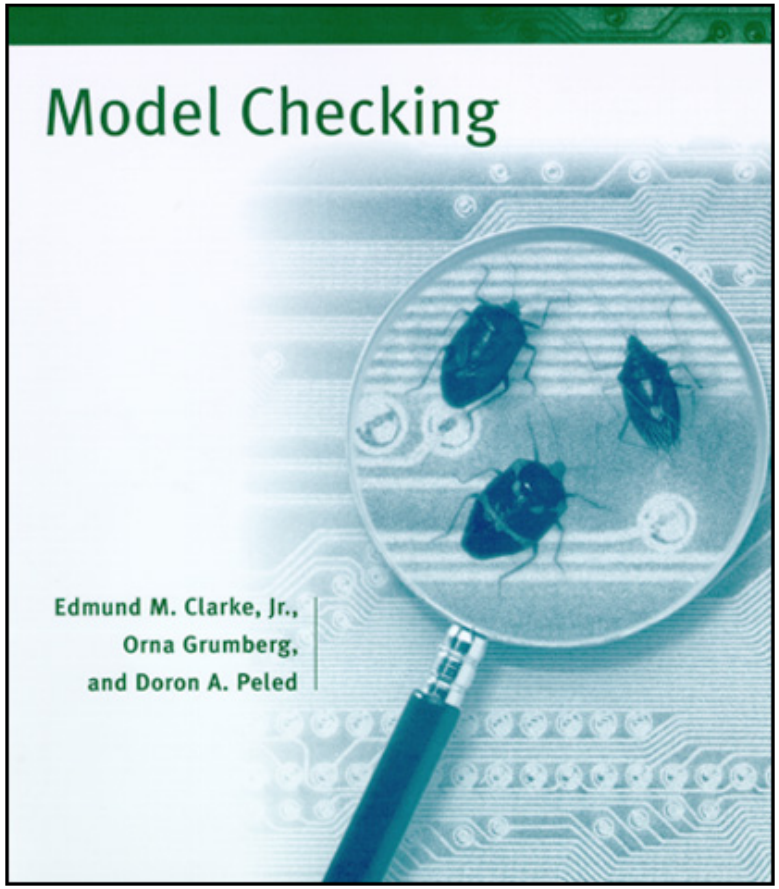
CPS AEB: Modeling challenges for CPS

Penn
University of Pennsylvania

A richer, systems view of computer science is needed. Ingredients include:

- ❑ Enriching CS models with relevant physical/resource properties
  - Physical, model-based computing
  - Resource aware (time/energy) computing
- ❑ Formal composition of **multiple** physics, models of computing, languages
  - Composition of heterogeneous components
- ❑ Impact of cyber components on physical components and vice versa
  - Physically-aware computing

Part II : Composable verification for CPS



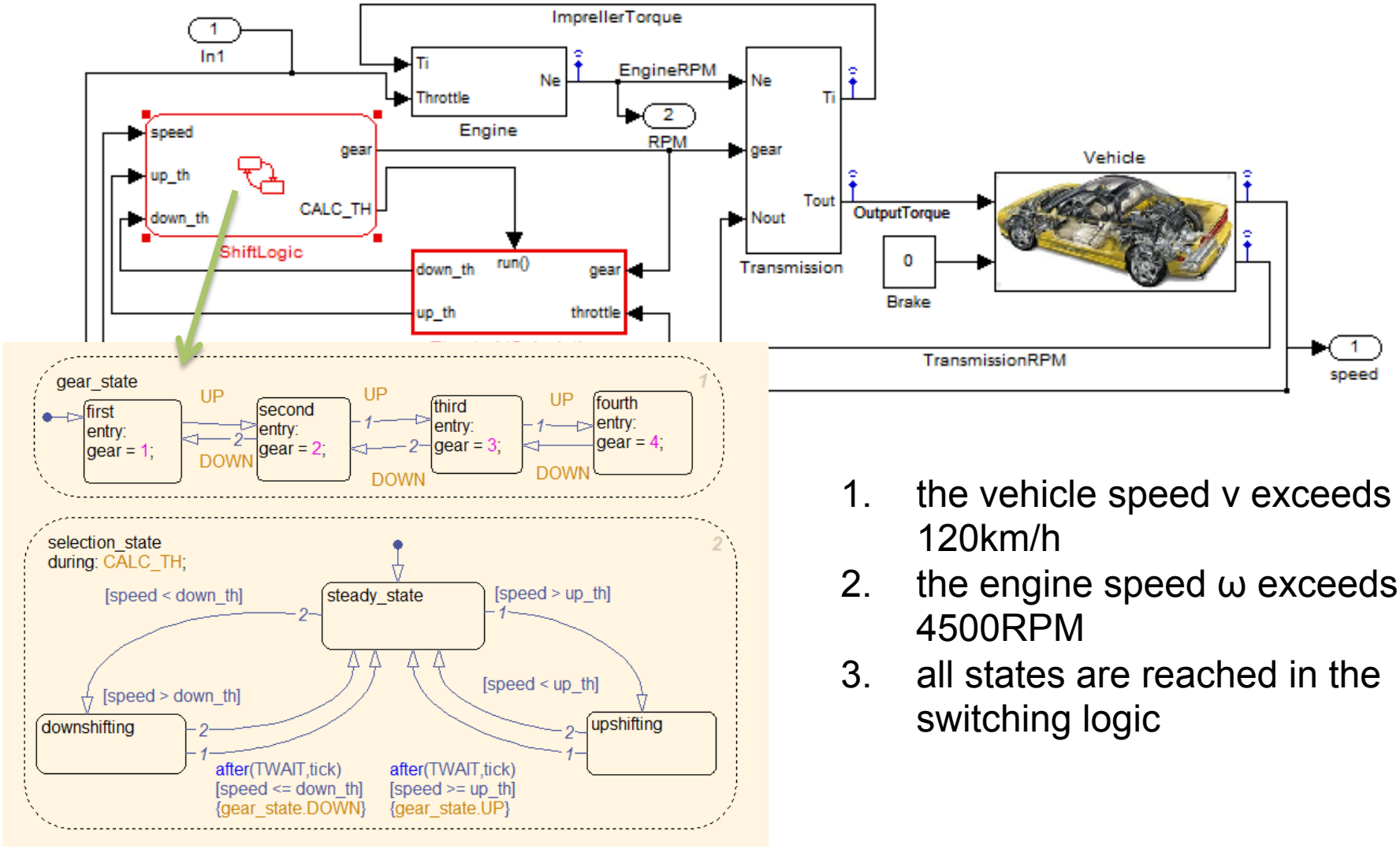
Traditional verification view

Safety verification: Is Reach(S) $\cap$ S_F empty?

Model checking: Does S satisfy temporal logic formula ϕ ?

Controller Synthesis: Does S||C satisfy temporal logic formula ?

Automatic trasmission verification



1. the vehicle speed v exceeds 120km/h
2. the engine speed ω exceeds 4500RPM
3. all states are reached in the switching logic

Assume-guarantee reasoning

- Consider component specifications of the form

$$\Phi : \Phi_E \Rightarrow \Phi_S$$

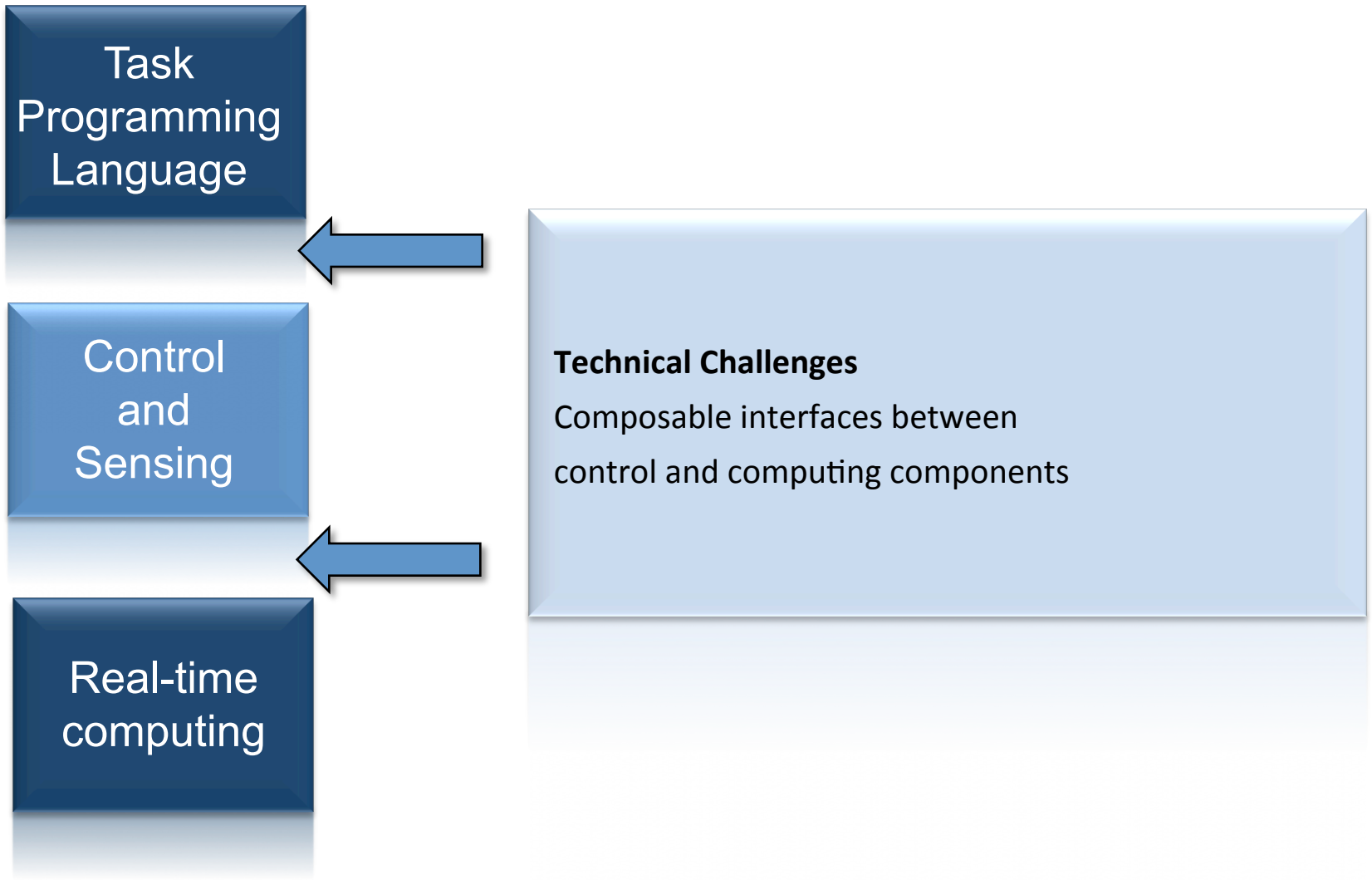
Φ_E: Environment assumptions

Φ_S: Desired behavior

If the component assumptions are met, then desired behavior should be guaranteed

Assumptions can model other components, physical or computational

How to analyze the overall design?

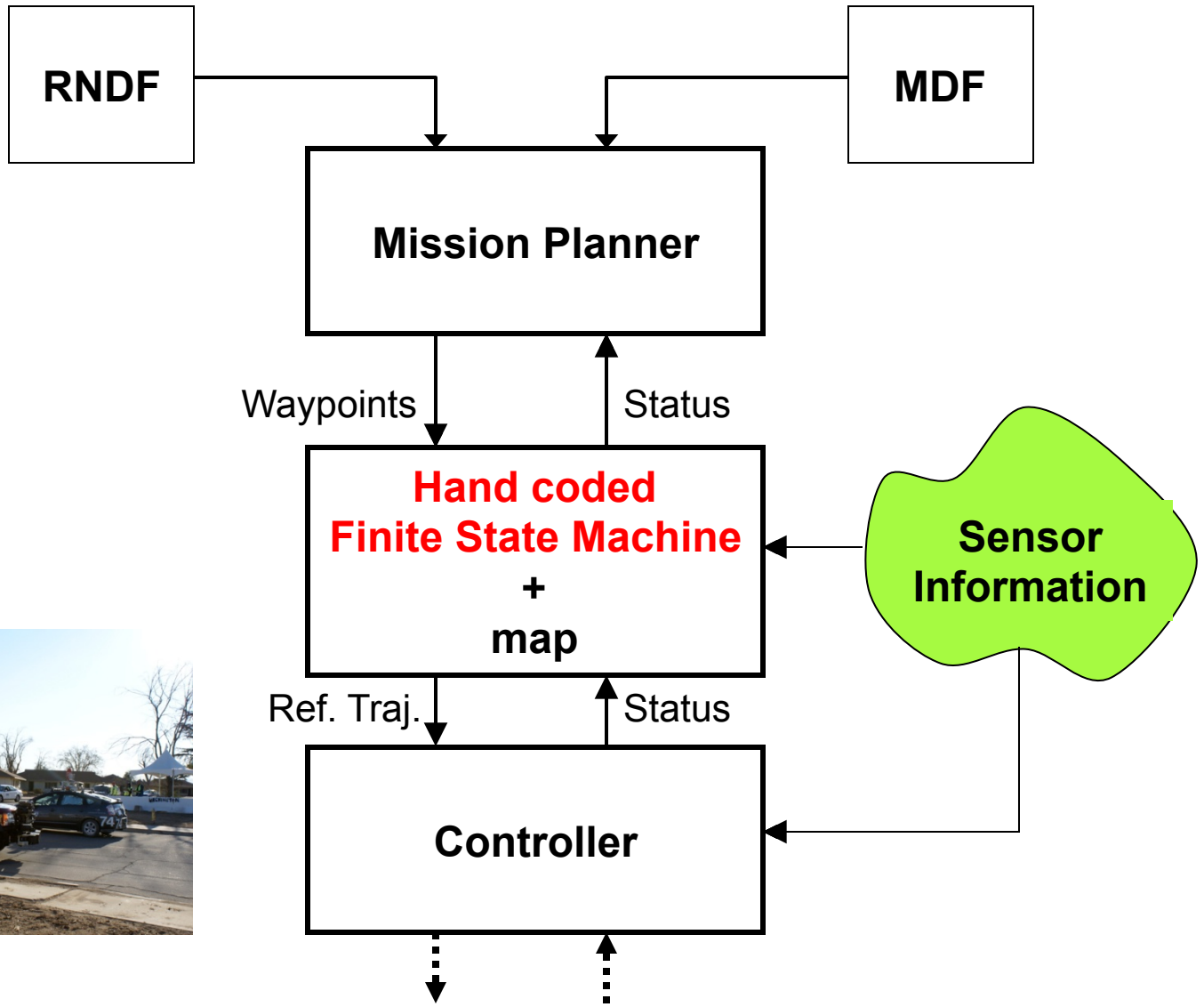


CPS AEB: Methods & Tools Challenges

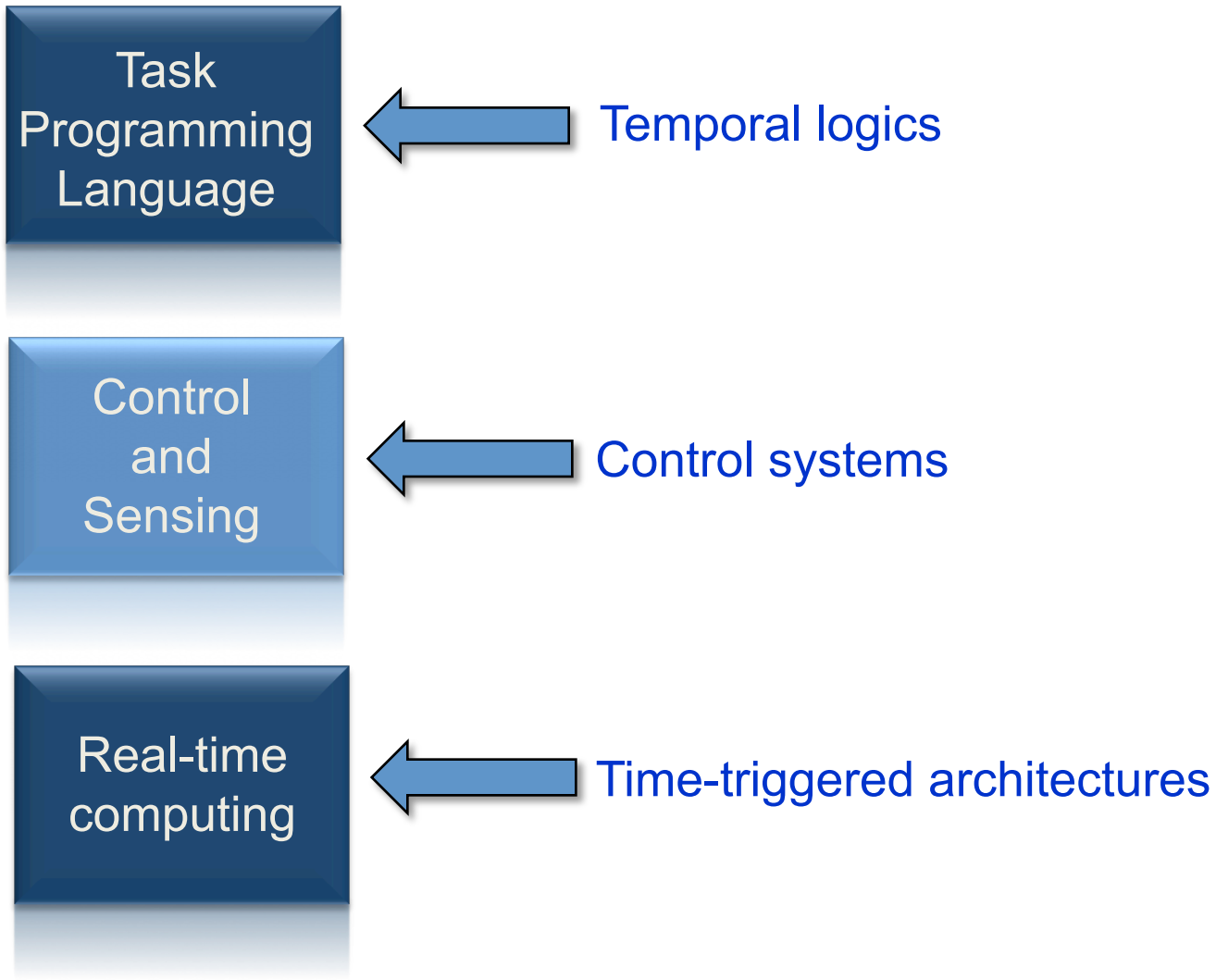
- From verification of C and P systems to verification of CPS systems
- Compositional verification and design
  - Requires composable interfaces
  - Assume-guarantee reasoning (contracts)
- Tradeoffs between different system views
  - Control performance versus scheduling flexibility
  - Understanding the price of compositionality
- From methods for flat CPS to methods for distributed CPS systems
- From verification methods to synthesis and code-generation

Part III : Robustness

DARPA Urban Challenge



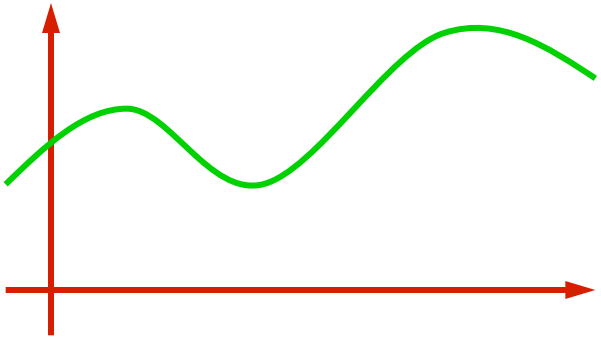
Control and computing hierarchy



From exact to approximate

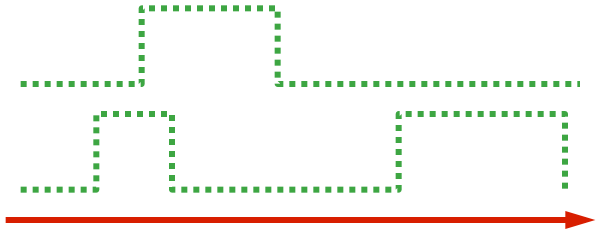
- Exact relationships useful for binary answers/relations
- Exact results are fragile with respect to uncertainty
- When interacting with the physical world, we need approximations
  - Labeled Markov processes (Desharnais et. al., TCS 2004)
  - Quantitative transition systems (de Alfaro et. al., ICALP 2004)
  - Quantitative generalizations of languages (Henzinger, DLT 2007)
- Approximate system relationships
  - Enable larger system "compression"
  - Quantify error/complexity tradeoffs
  - Provide measures of robustness
  - Potentially introduce different algorithms

Boolean testing & verification of CPS



A/D

Boolean abstraction



LTL / MTL

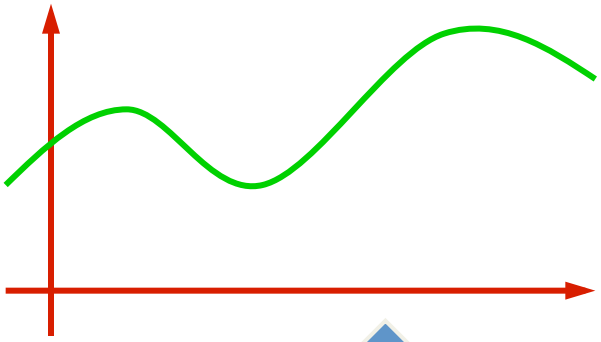
$\Phi = G(p_1 \rightarrow F_{\leq 2}\, p_2)$

Testing/Verification Algorithm

Truth Value
{true,false}

[Maler and Nickovic '04]
[Thati and Rosu '04]
[Rosu and Havelund '05]
[Geilen '01]
others …

Robust verification for CPS



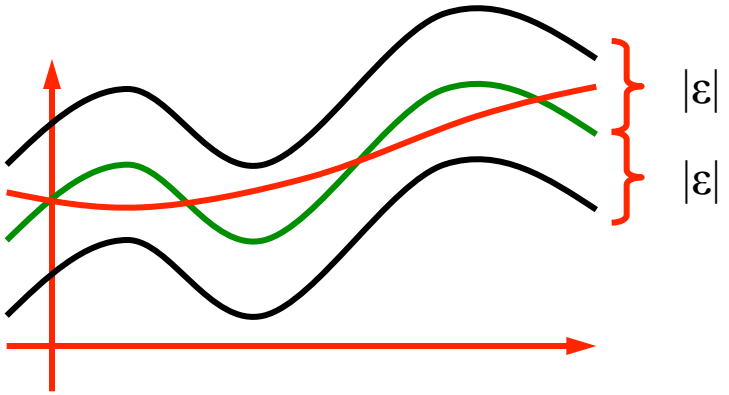
LTL / MTL

$\Phi = G(p_1 \rightarrow F_{\leq 2}\, p_2)$

Robust Testing/ Verification Algorithm

Robustness Estimate

ε



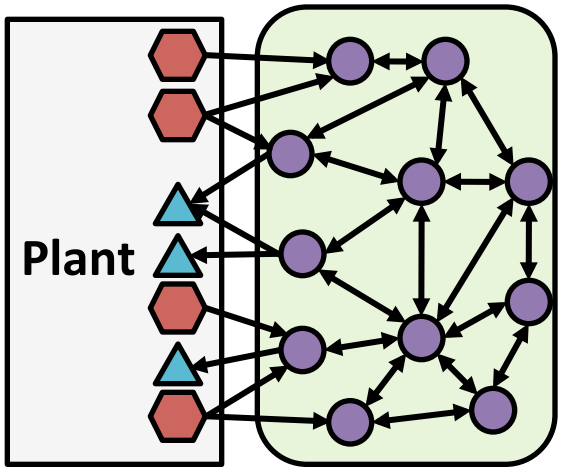
CPS AEB: Robust methods for CPS

- Computing needs to transform from exact to approximate.
  - From boolean to robust computing
  - From qualitative to quantitatve
- Robust computing interacting with uncertain world (stochastic)
  - Uncertainty due to P or C components, networking
  - From deterministic to stochastic verification
- Robustness verification of CPS systems
  - Closer to sensitivity analysis
- Verification of numerical software

Part IV : CPS Security
Beyond Cybersecurity

Monitoring for faulty and malicious behavior

- Many CS efforts focus on building security walls
- What happens after a fault or after an attack?
- What if certain wireless nodes become faulty or malicious?



Monitoring for faulty and malicious behavior

- Security of control networks in industrial control systems is a major issue [NIST Technical Report, 2008]
  - Data Historian: Maintain and analyze logs of plant and network behavior
  - Intrusion Detection System: Detect and identify any abnormal activities

- Is it possible to design an Intrusion Detection System (IDS) to determine if any nodes are not following protocol?
- Can IDS scheme avoid listening all nodes? Under what conditions? Which nodes?

CPS AEB: Secure CPS

- CPS Beyond cybersecurity
  - From pre-attack defense to post-attack methods
  - Detection, identification, reconfiguration, graceful degradation
- Theory of resilient and trustworthy CPS
  - Security/trust metrics
- Privacy for CPS
  - Crucial in many contexts (i.e. medical CPS)
  - Interesting ideas in database theory (differential privacy)