Safety of Cyber-Physical Systems

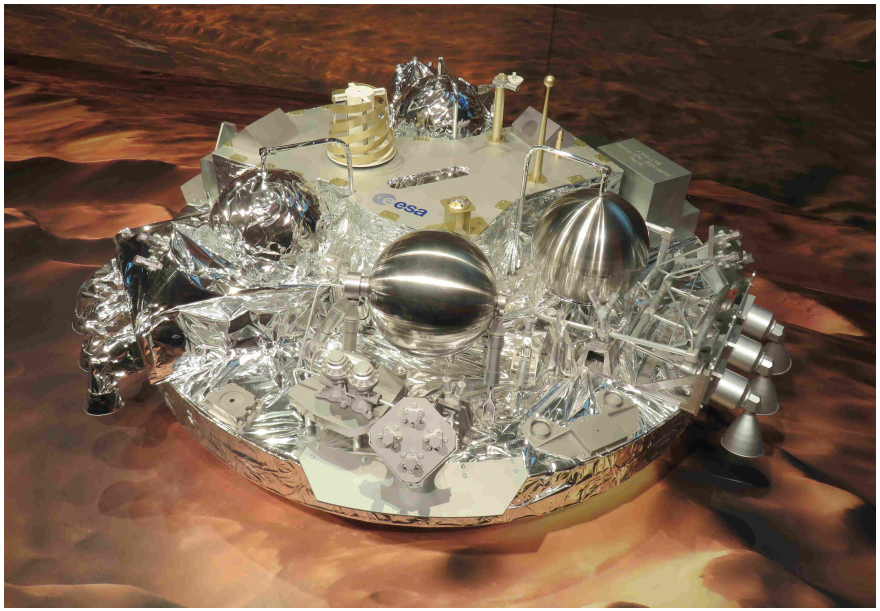
Uli Fahrenberg

École polytechnique, Palaiseau, France

February 27, 2018

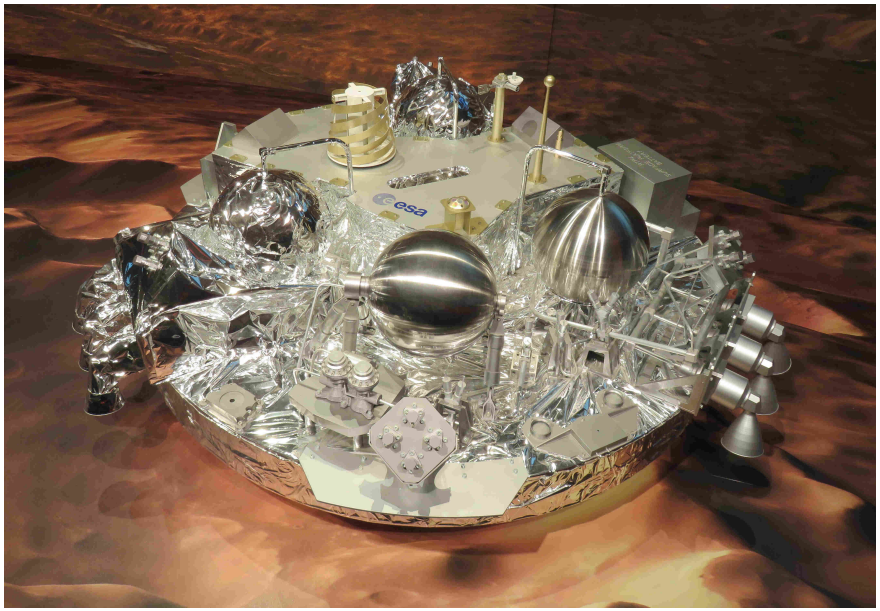
Schiaparelli

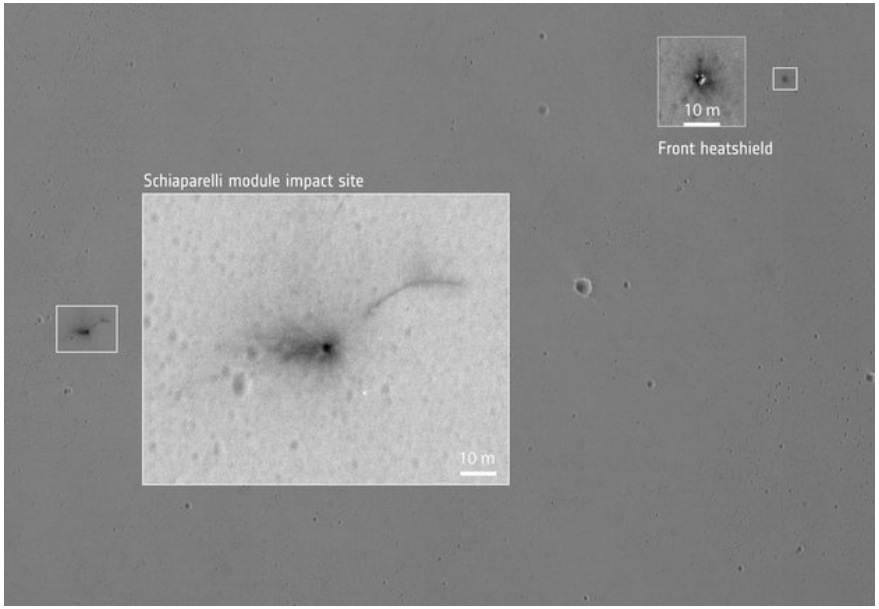
ESA / Roscosmos Experimental Mars Lander



Schiaparelli

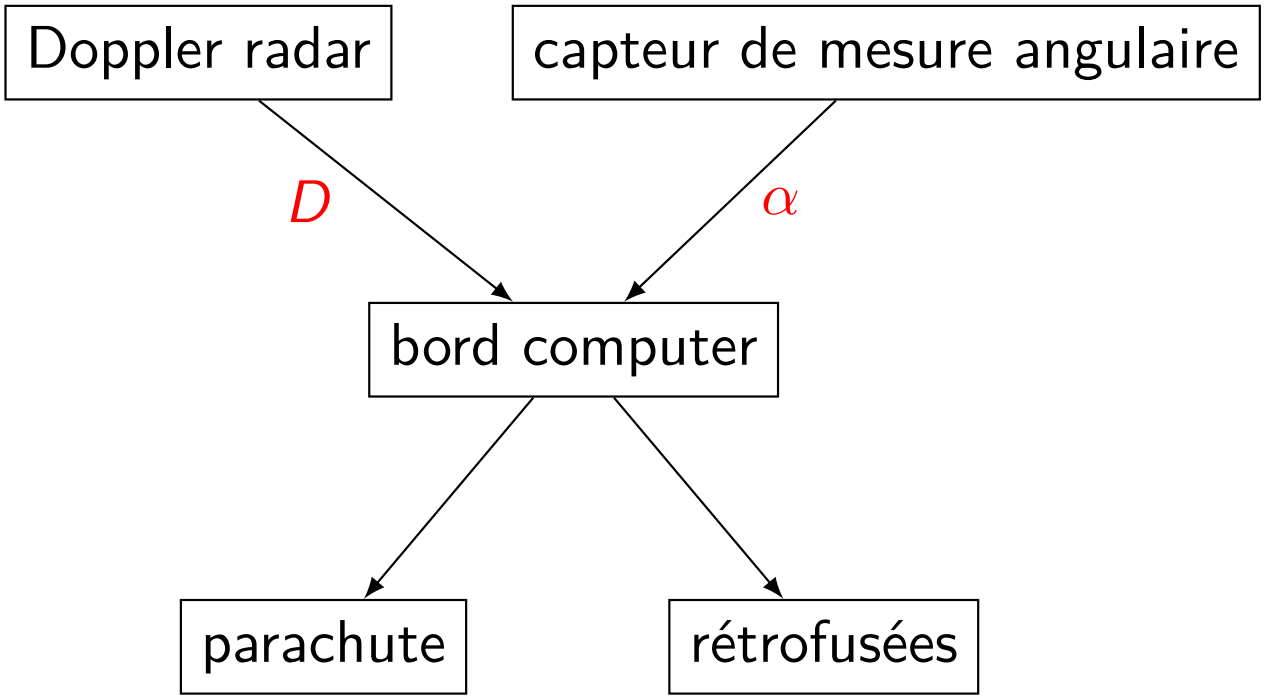
ESA / Roscosmos Experimental Mars Lander

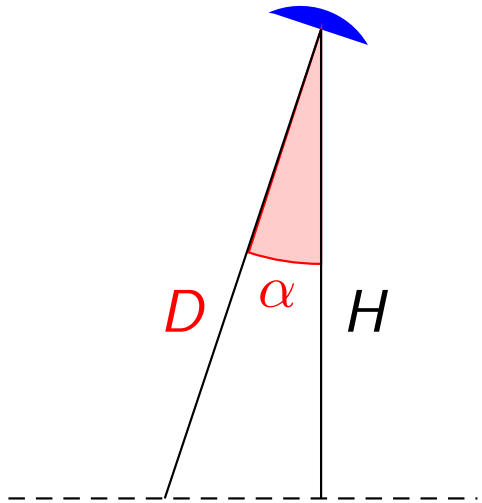




- an example of a cyber-physical system

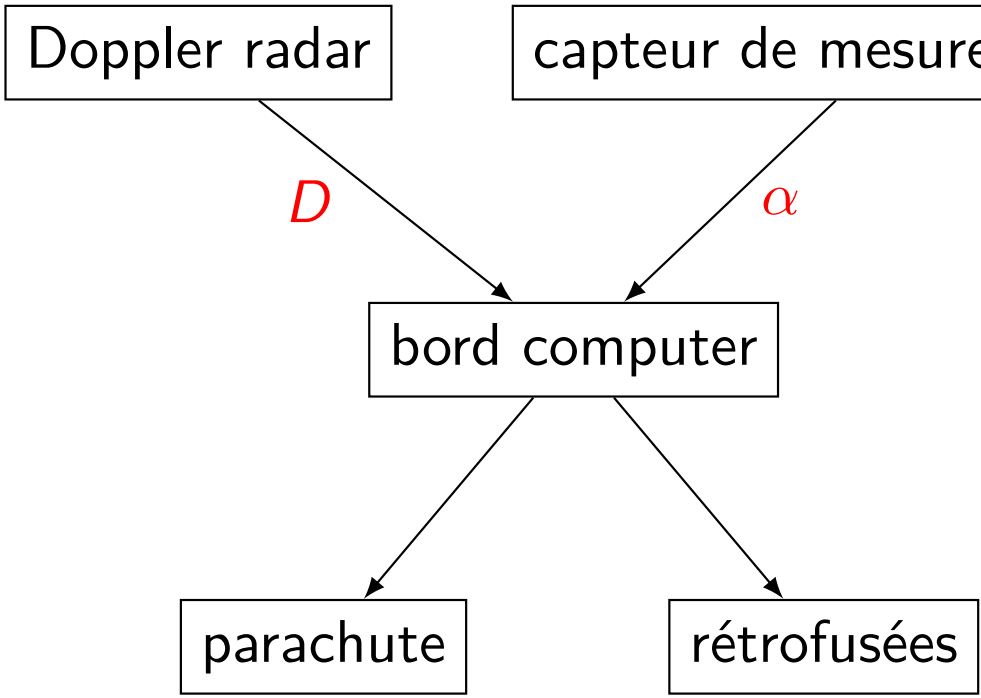
Schiaparelli
Simplified Schematic

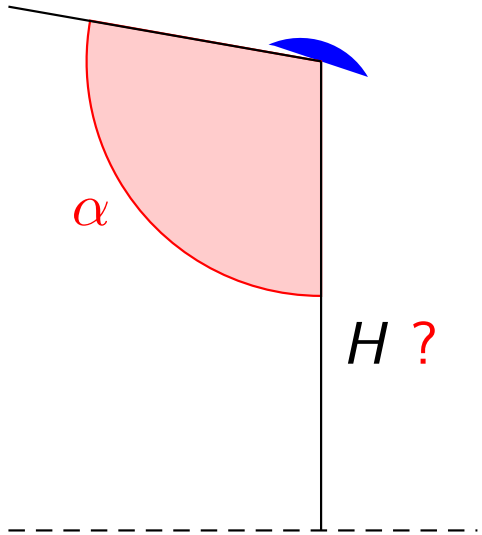




$$H = D \cdot \cos(\alpha)$$

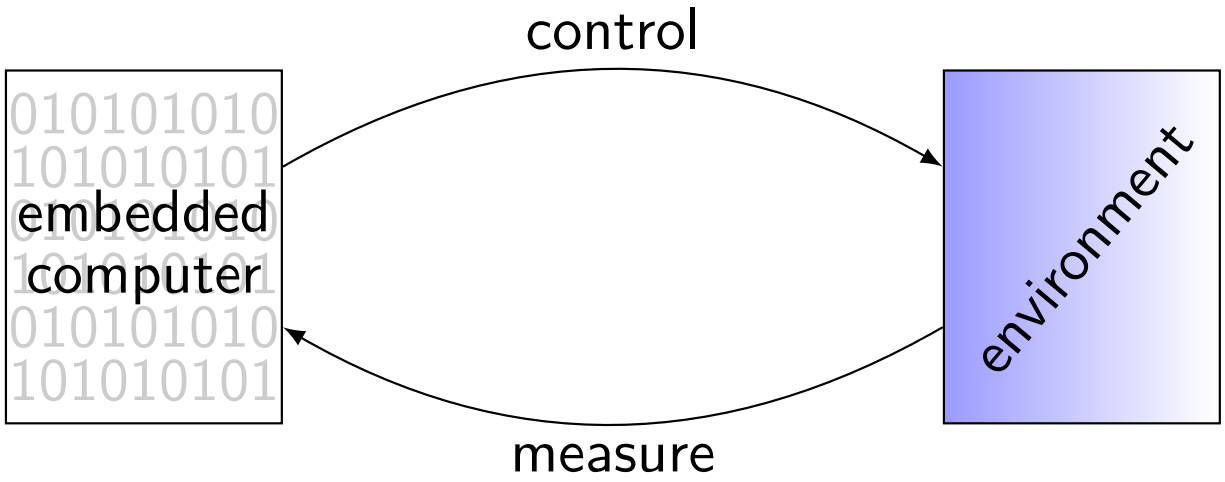
Schiaparelli
Simplified Schematic

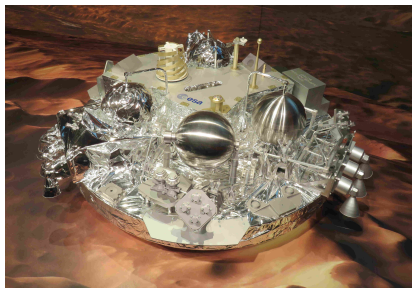
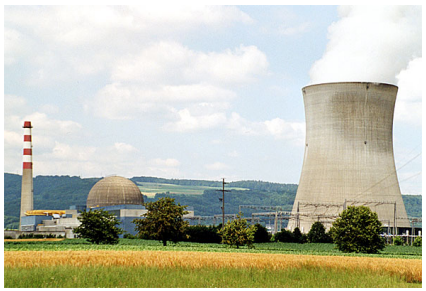




$$H = D \cdot \cos(\alpha)$$

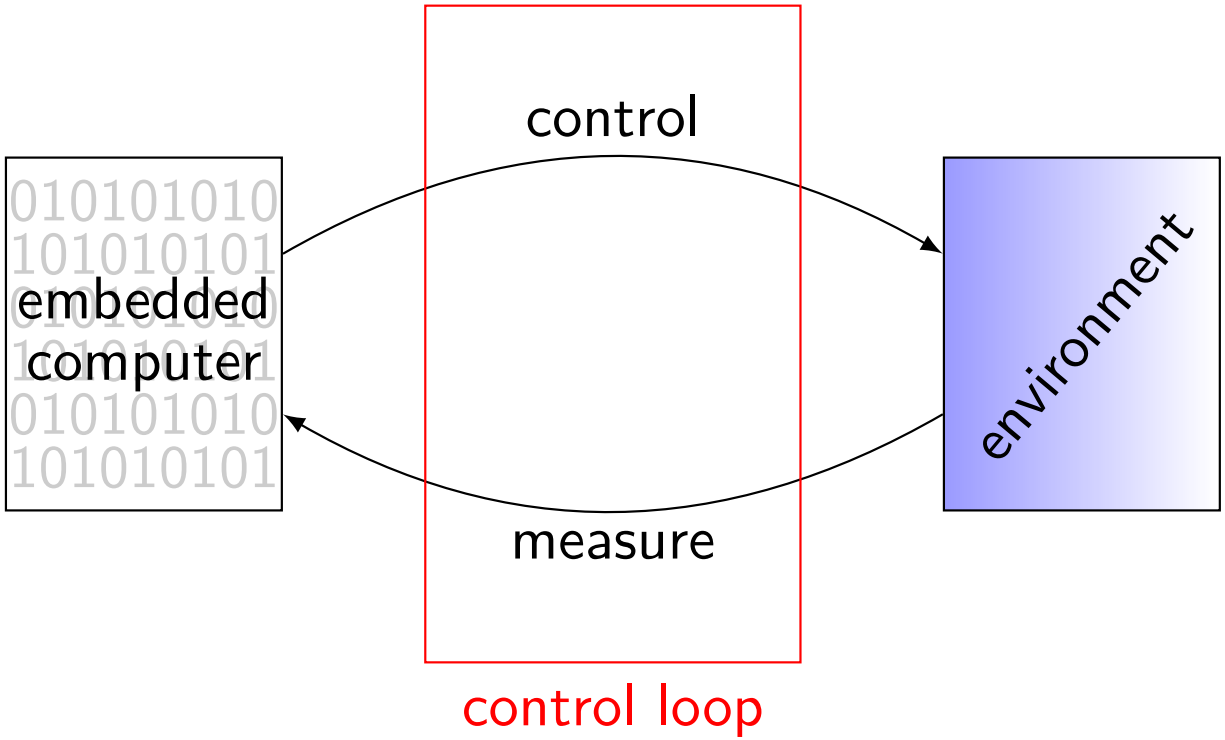
Cyber-physical systems
Examples



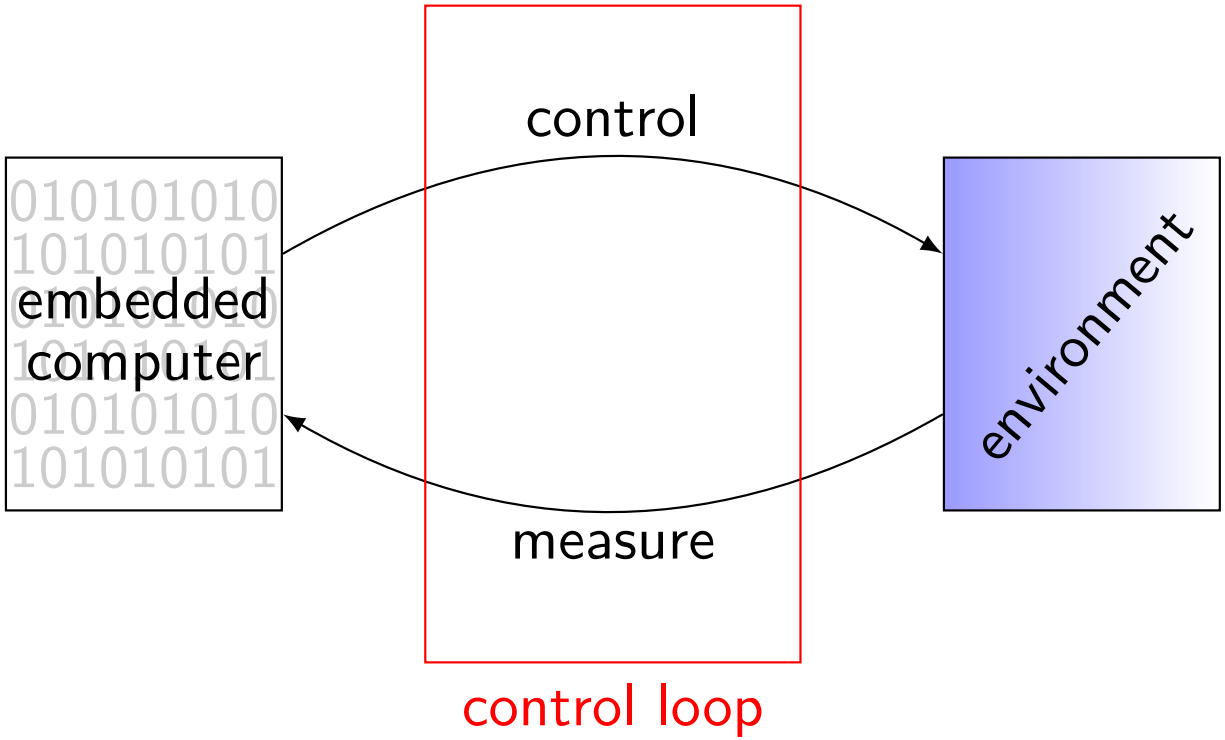
Cyber-physical systems

Schematic



Cyber-physical systems

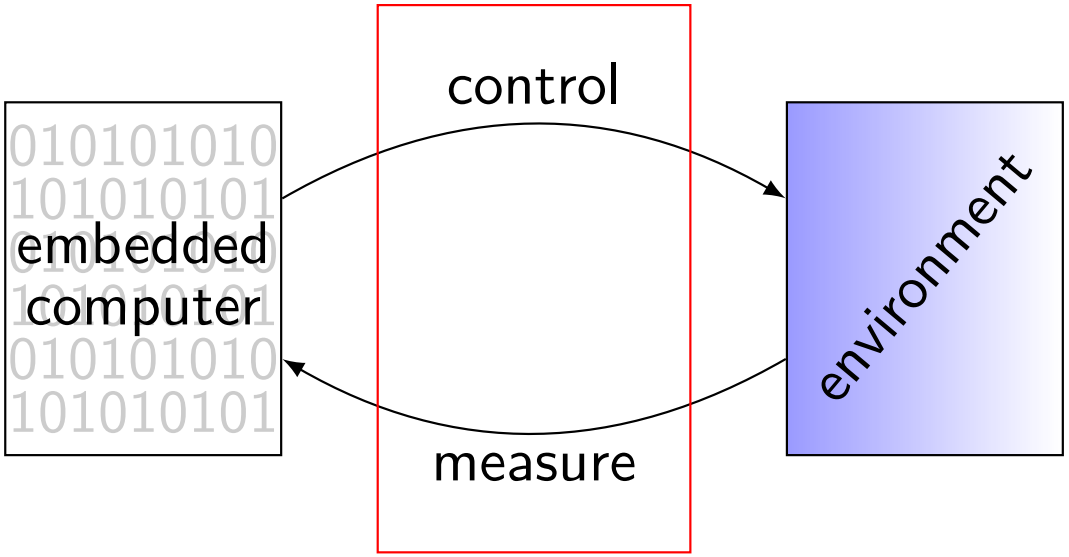
Schematic



Informatics Control theory Mathematics

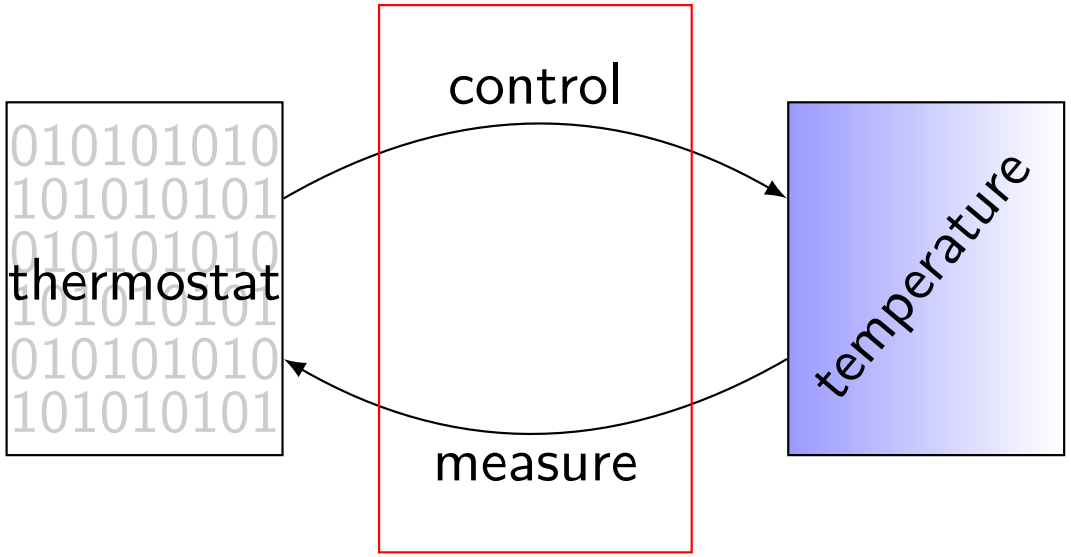
Hybrid Automata

Model of a thermostat

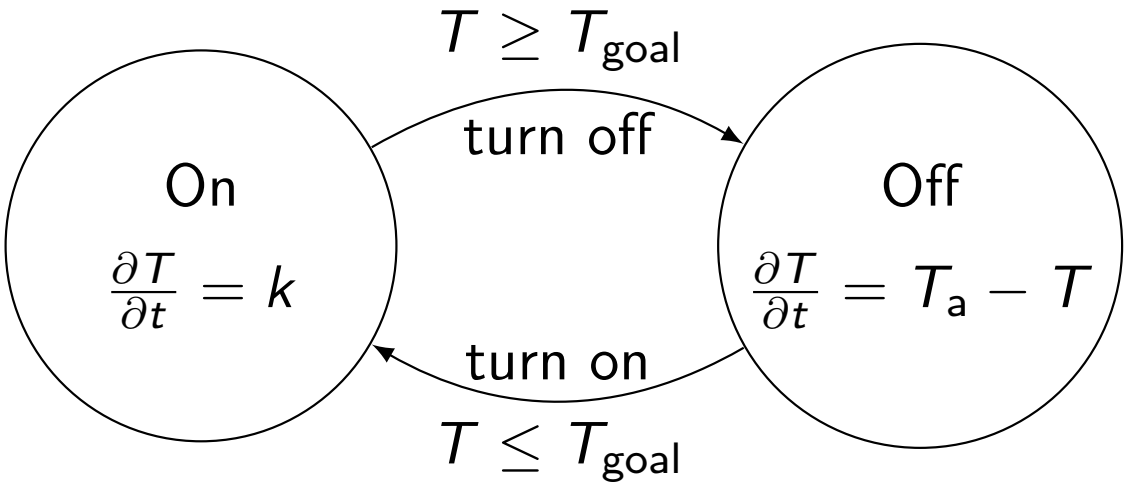


Hybrid Automata

Model of a thermostat



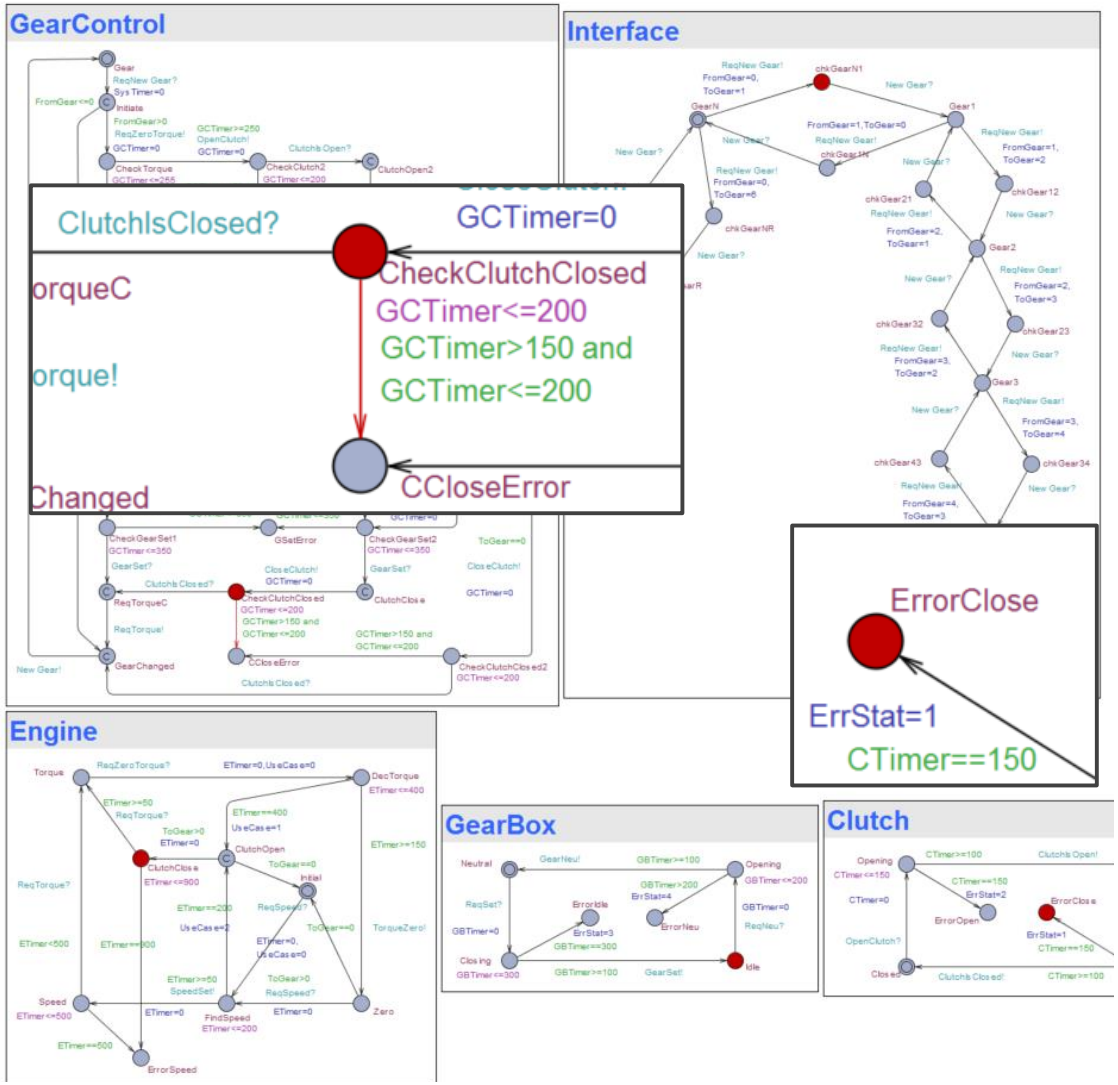
as a hybrid automaton:



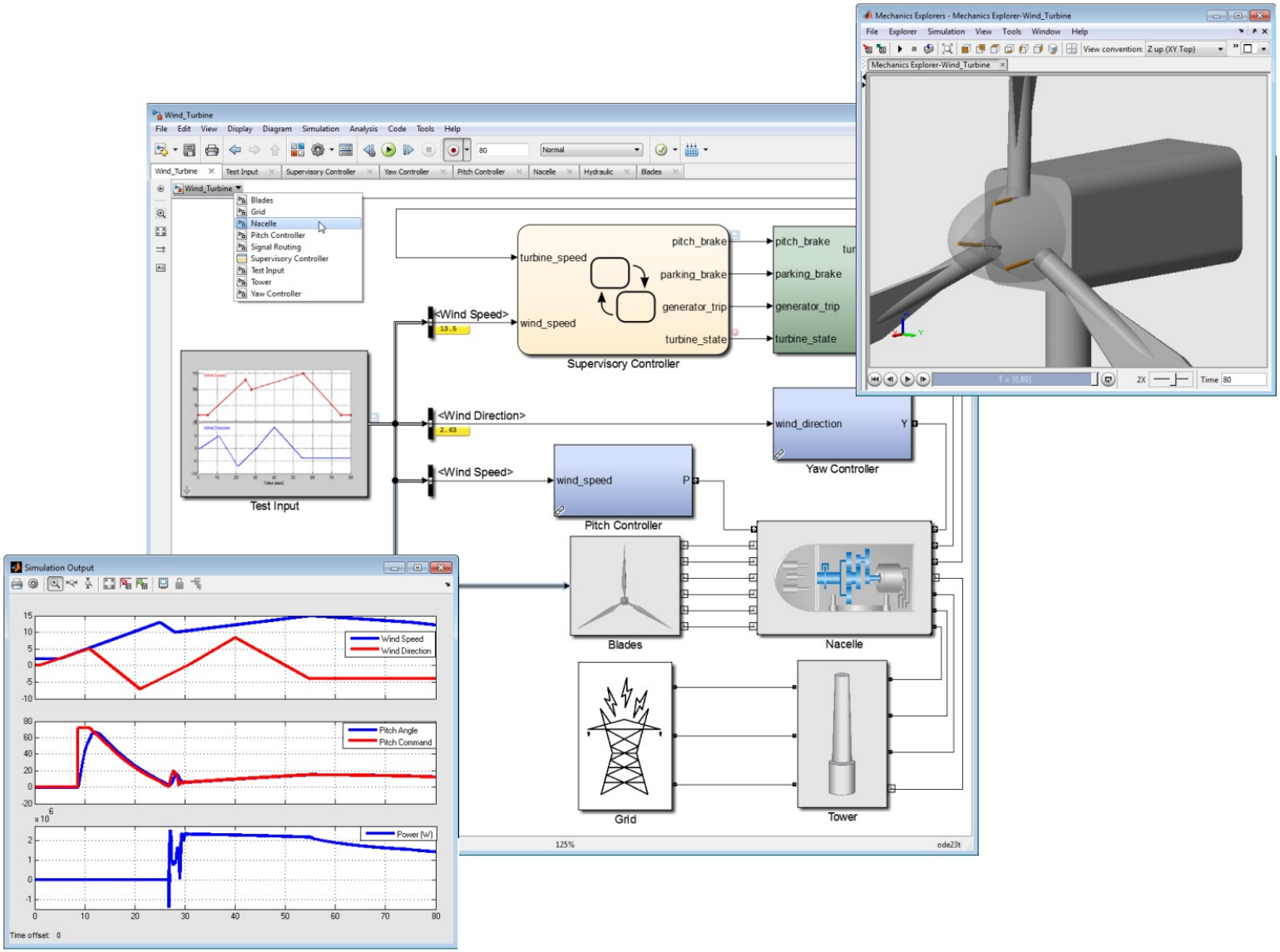
Timed Automata
UPPAAL

- <Global variables>
  - UseCase = 0
  - FromGear = 0
  - ToGear = 0
  - ErrStat = 0
- <Constraints>
  - CTimer ≥ 0
  - ETimer ≥ 0
  - GBTimer ≥ 0
  - GCTimer ≥ 0
  - SysTimer ≥ 0
  - GearControl.GCTimer ≥ 0
  - CTimer = ETimer
  - ETimer = GBTimer
  - GBTimer = GCTimer
  - GCTimer = SysTimer
  - SysTimer = GearControl.GCTimer
  - GearControl.GCTimer = CTimer

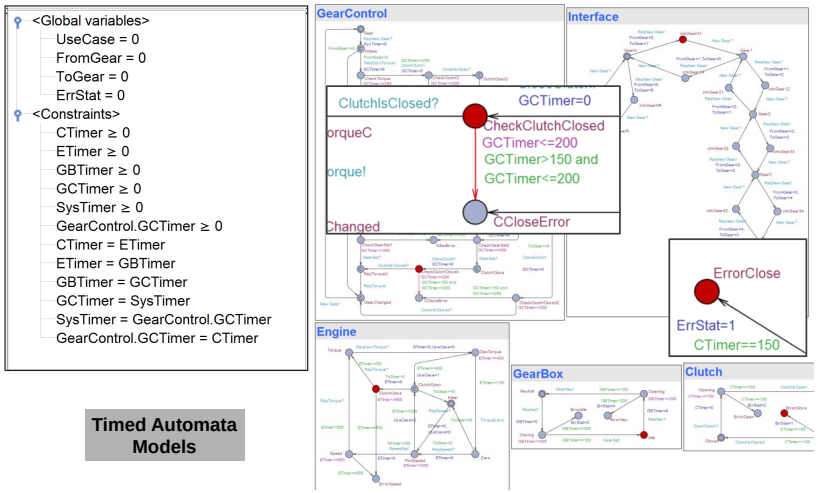
Timed Automata Models



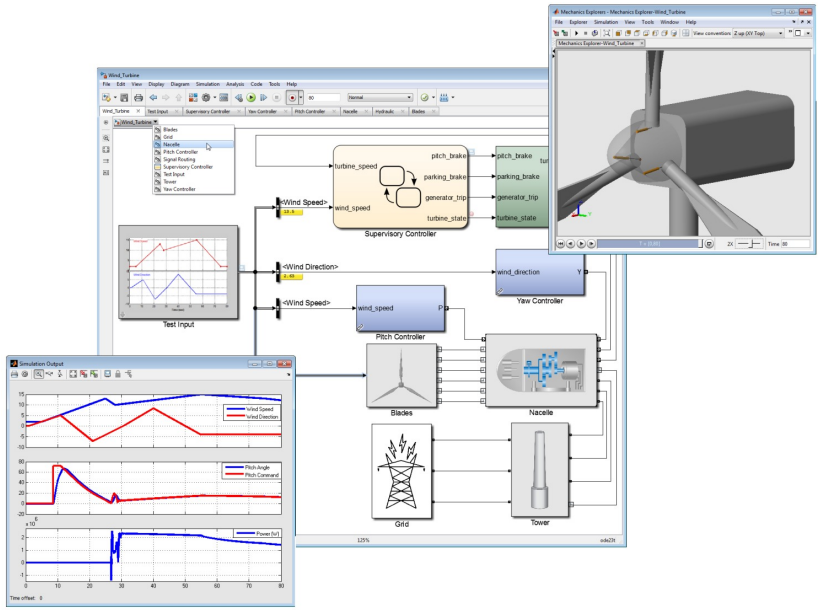
Simulink Model
of a wind turbine



Mathematical Models
in industry



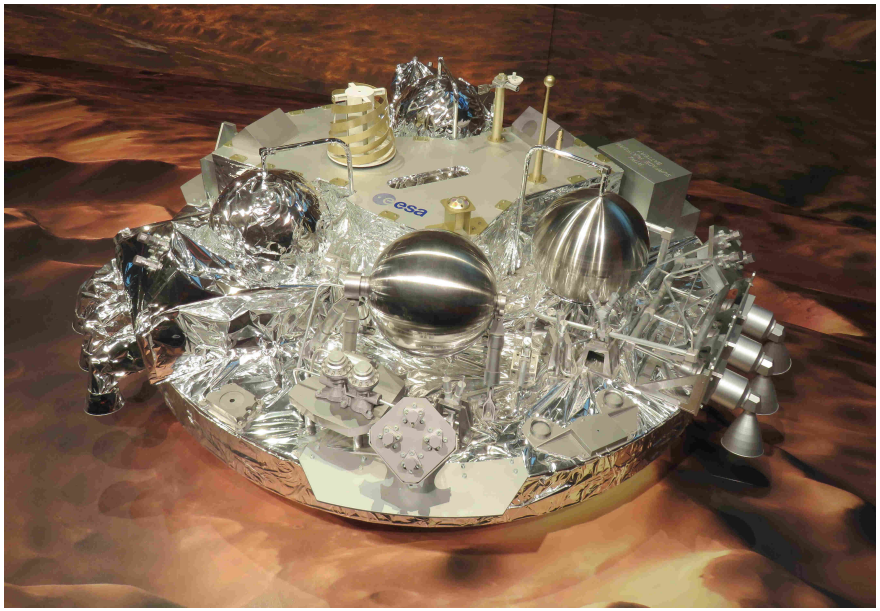
Timed Automata Models

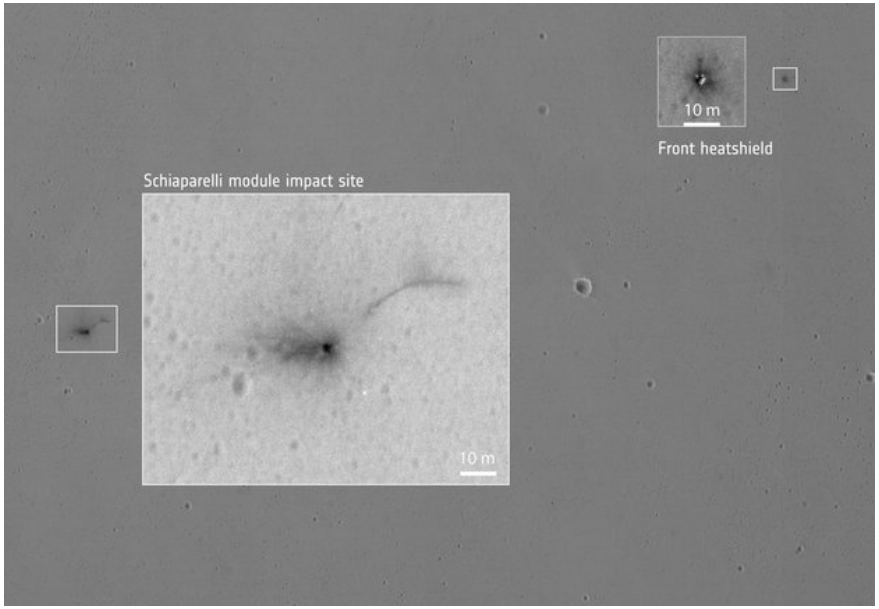


- Mathematical modeling is an industry standard
- Especially in avionics / space flight
- Mostly Statechart models like with Simulink
- Used for testing design by simulation

Schiaparelli

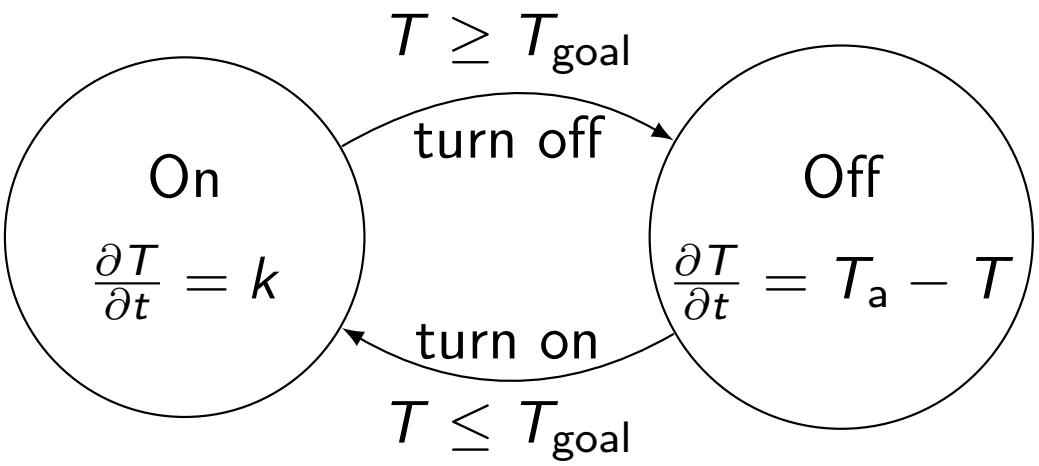
Or, Sometimes Simulation Does Not Suffice





Formal Verification

Ensuring properties beyond simulation



$$\mathbf{F}\ \mathbf{G}\ \mathbf{F}\ \phi$$

$$\text{Mod} \models \text{Prop}$$

formal models — formal properties

Formal Verification of CPS

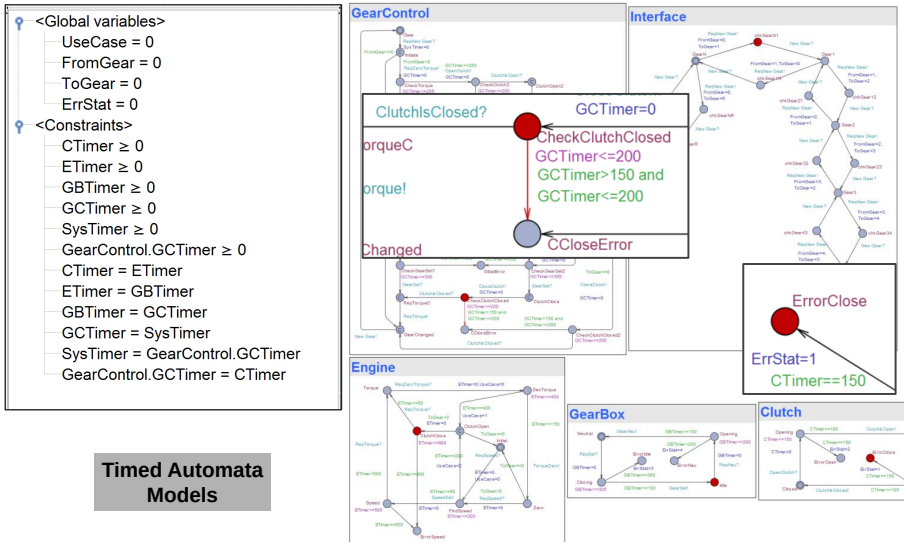
State of the art

Timed automata:

- formally decidable
- fast algorithms
- UPPAAL
- lack expressivity
- extensions to weights and games

Hybrid automata:

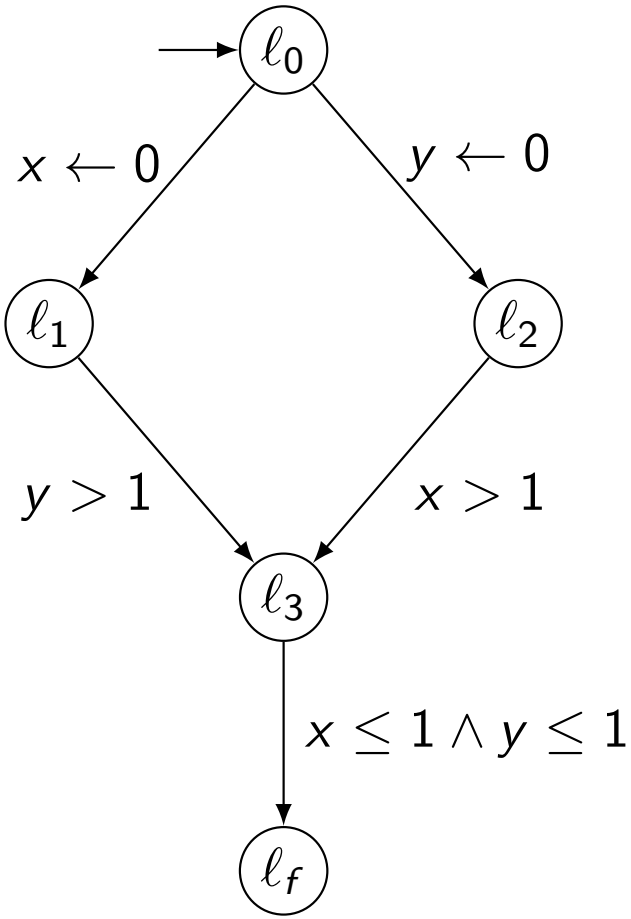
- formally undecidable
- over- and underapproximations
- SpaceEx, PHAVer, iSAT3, C2E2, ...
- curse of dimension
- sweet spot: linear hybrid automata

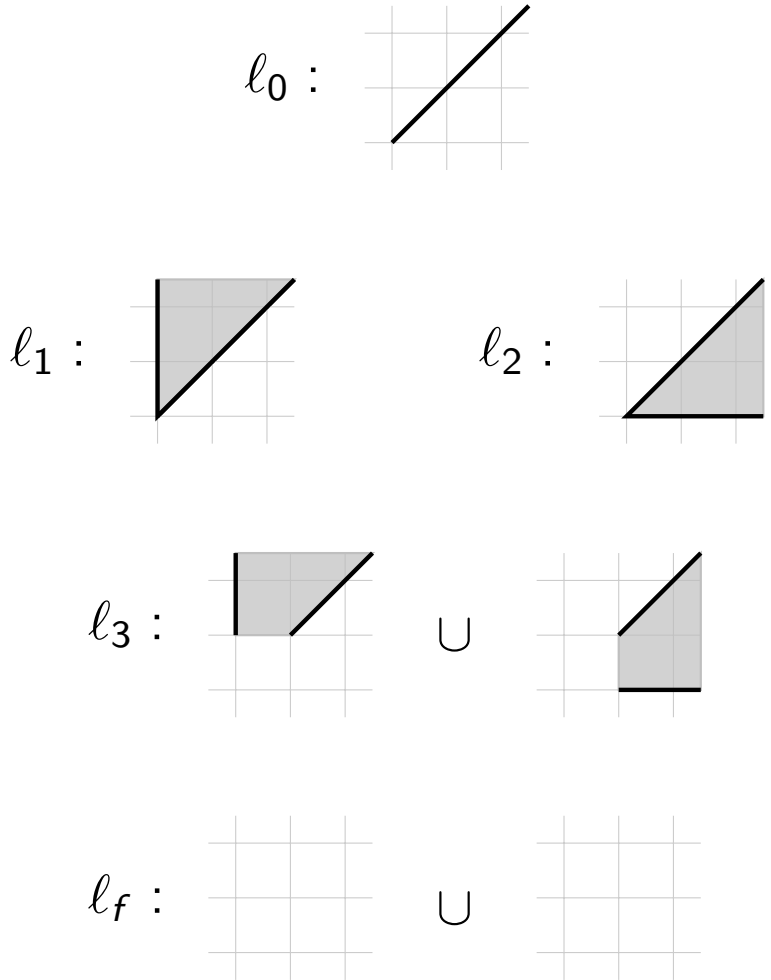


Timed Automata Models

- combine simulation and verification
- statistical methods
- learning
- compositionality
- very active research area!

A Bit of UPPAAL: Zones

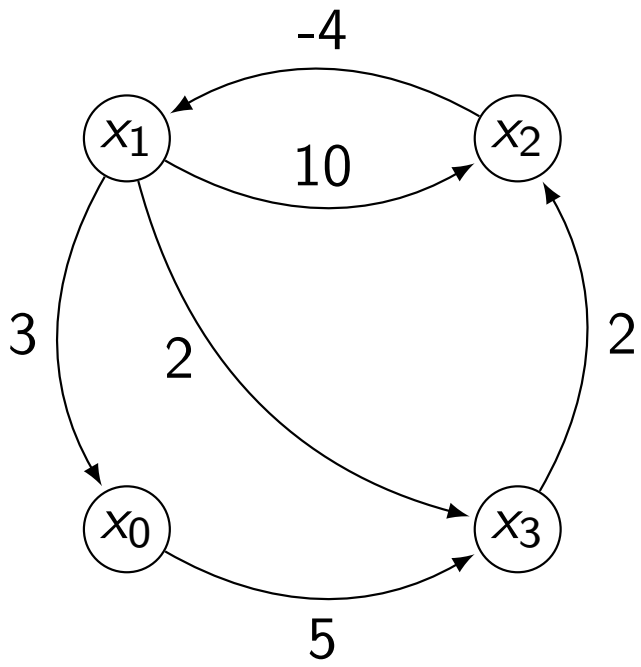




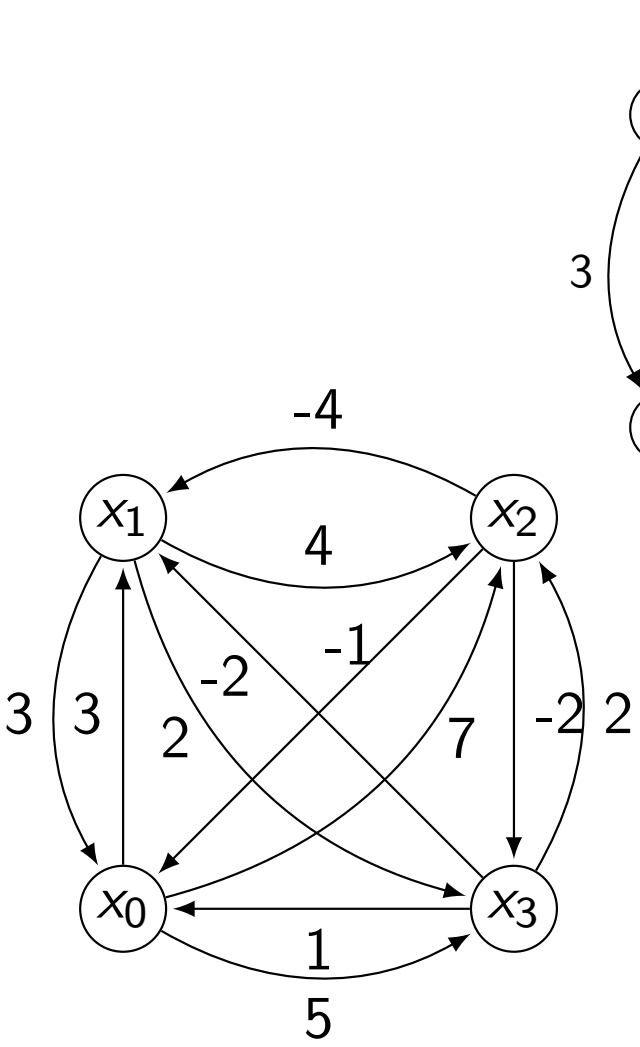
Zones: Representation

Zone $\rightsquigarrow$ digraph $\cong$ difference-bound matrix

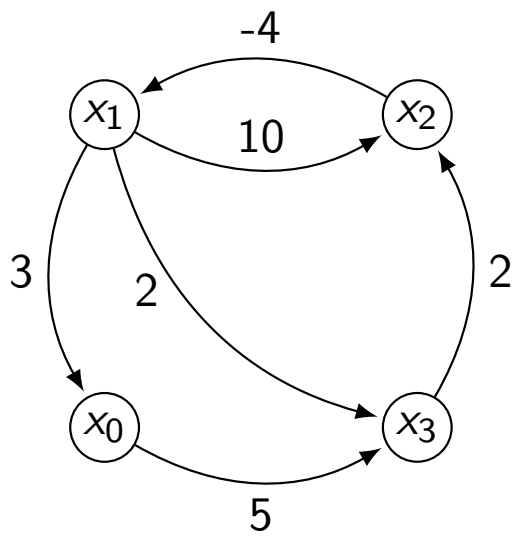
$$Z = \begin{cases} x_1 \leq 3 \\ x_1 - x_2 \leq 10 \\ x_1 - x_2 \geq 4 \\ x_1 - x_3 \leq 2 \\ x_3 - x_2 \leq 2 \\ x_3 \geq -5 \end{cases}$$

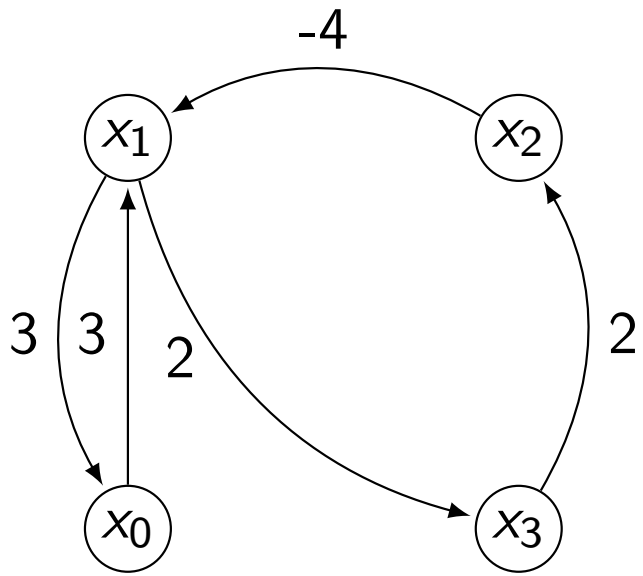


Zones: Representation



shortest-path closure

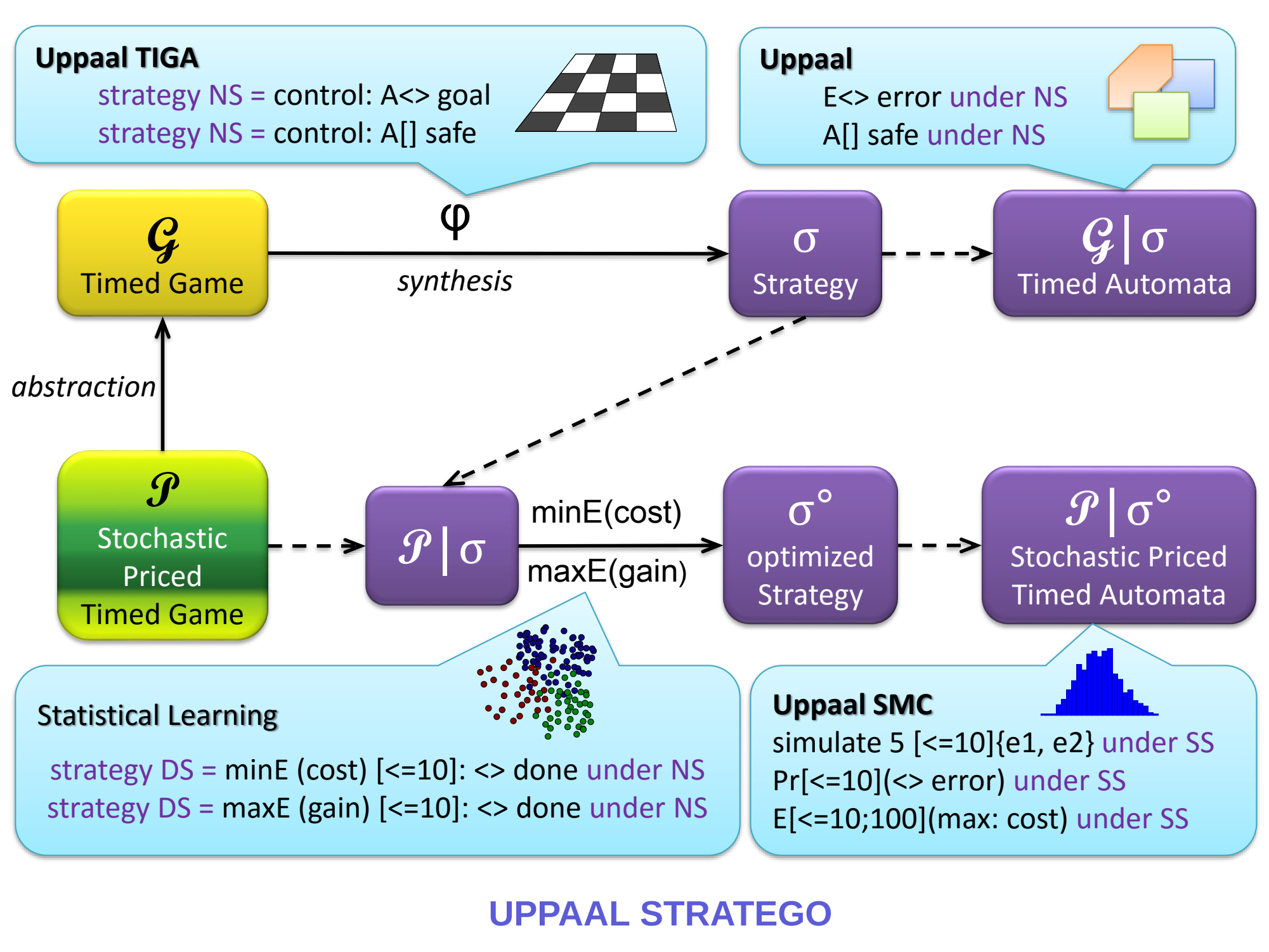




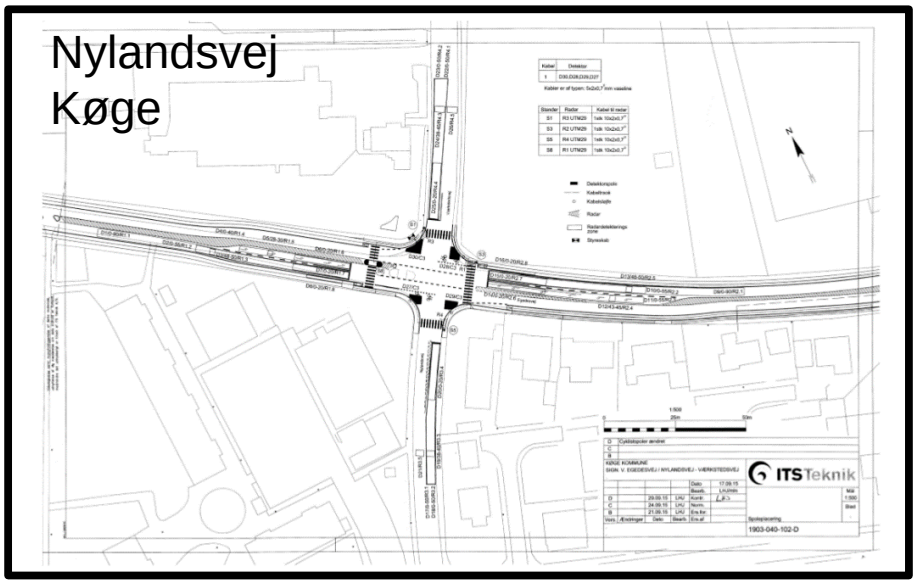
shortest-path reduction

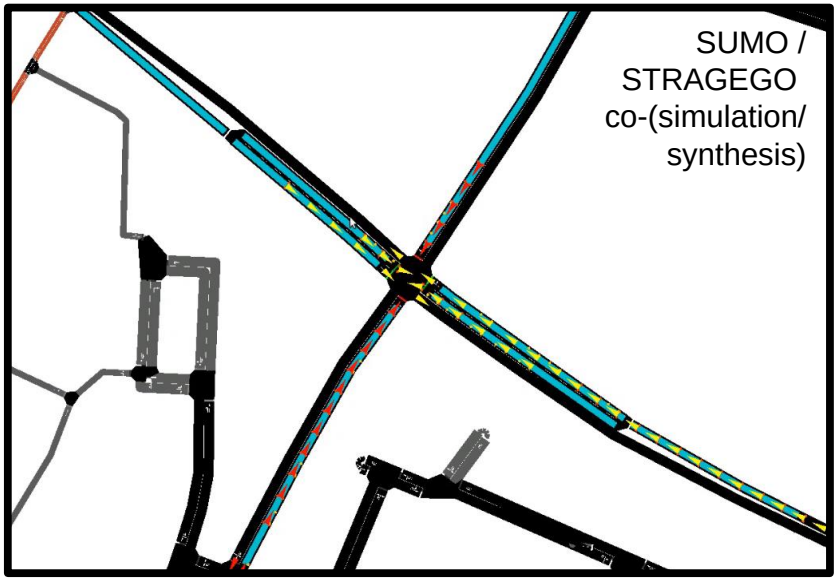
Zones: Algorithms

- Using closures or reductions
- Delay, reset, intersection, inclusion check can be done in $O(|C|^3)$
- In practice: combined Passed-Waiting list
- Each location has a list of zones ($\cong$ union)
- Represented using clock decision diagrams
- Extract DBMs from CDD $\rightsquigarrow$ perform operations on each $\rightsquigarrow$ re-combine to new CDD


Uppaal TIGA
strategy NS = control: A<> goal
strategy NS = control: A[] safe
Uppaal
E<> error under NS
A[] safe under NS
𝒢
Timed Game
φ
synthesis
σ
Strategy
𝒢 | σ
Timed Automata
abstraction
𝒫
Stochastic Priced Timed Game
𝒫 | σ
minE(cost)
maxE(gain)
σ°
optimized Strategy
𝒫 | σ°
Stochastic Priced Timed Automata
Statistical Learning
strategy DS = minE (cost) [<=10]: <> done under NS
strategy DS = maxE (gain) [<=10]: <> done under NS
Uppaal SMC
simulate 5 [<=10]{e1, e2} under SS
Pr[<=10](<> error) under SS
E[<=10;100](max: cost) under SS
UPPAAL STRATEGO

UPPAAL Stratego for Traffic Control





Scenario	Static		Loop Induction		Stratego		Imp W time over LI %
	Jam Km	W time s	Jam Km	W time s	Jam Km	W time s	
MAX	1451	191990	1185	157200	551	73001	53.5%
MID	456	60362	369	48936	331	43878	
LOW	138	18425	139	18566	101	13451	27.5%

Scenario:
2 hours traffic

UPPAAL Stratego for Traffic Control

Number of cars waiting in each lane *(full information)*

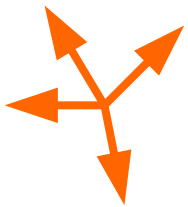
1: Every 5 to 8 sec read sensor data
2: **if** Traffic Light in yellow phase **then**
3: Run UPPAAL STRATEGO – decide next green phase
4: **else if** Traffic Light in green phase **then**
5: Run UPPAAL STRATEGO – extend green phase or go to yellow
6: **end if**

ONLINE Synthesis

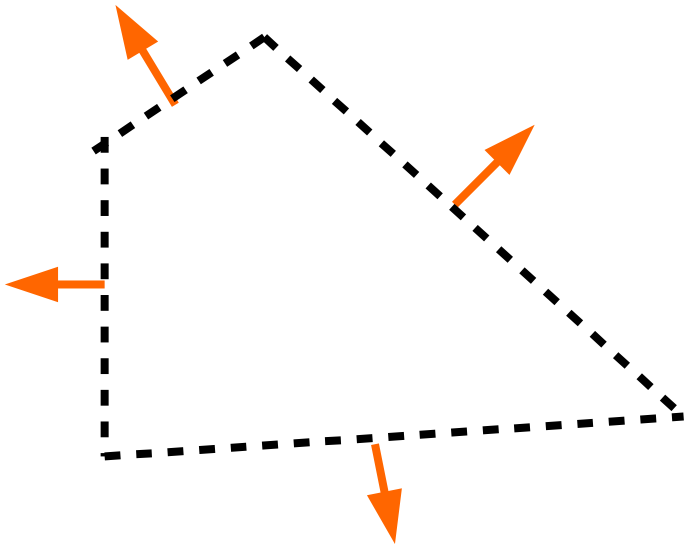
- Identify optimal strategy up to horizon H=90sec.
  - Strategy changes phase (at least 5 sec).
    - Modelling of stochastic arrival of cars in different directions (from 60-850 cars/hour)
- Minimize waiting time or jam (# of waiting >2sec)

A Bit of SpaceEx: Template Polyhedra

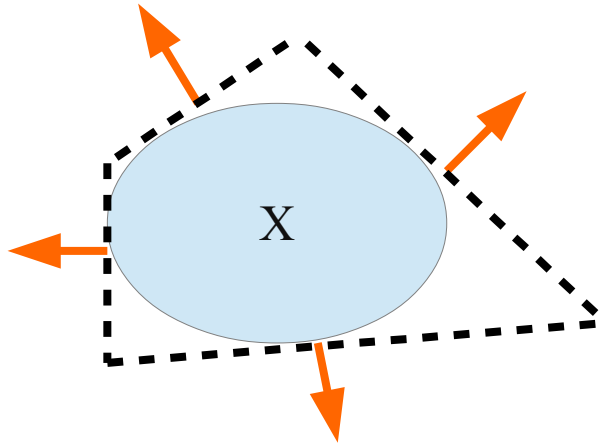
Bogomolov, Frehse, Giacobbe, Henzinger: TACAS 2017



Template:
set of directions

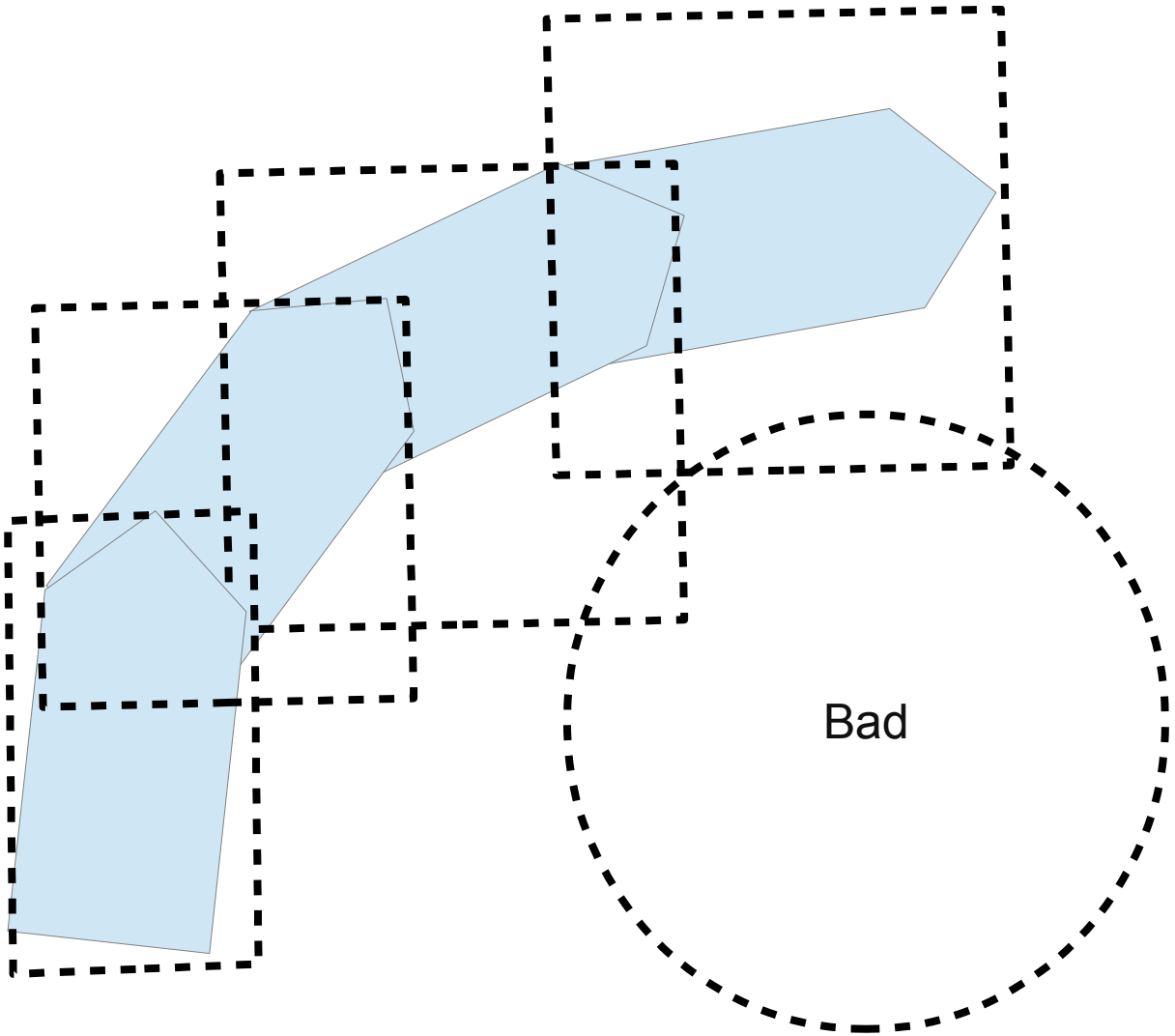


Template polyhedron:
facets are normal to
the template

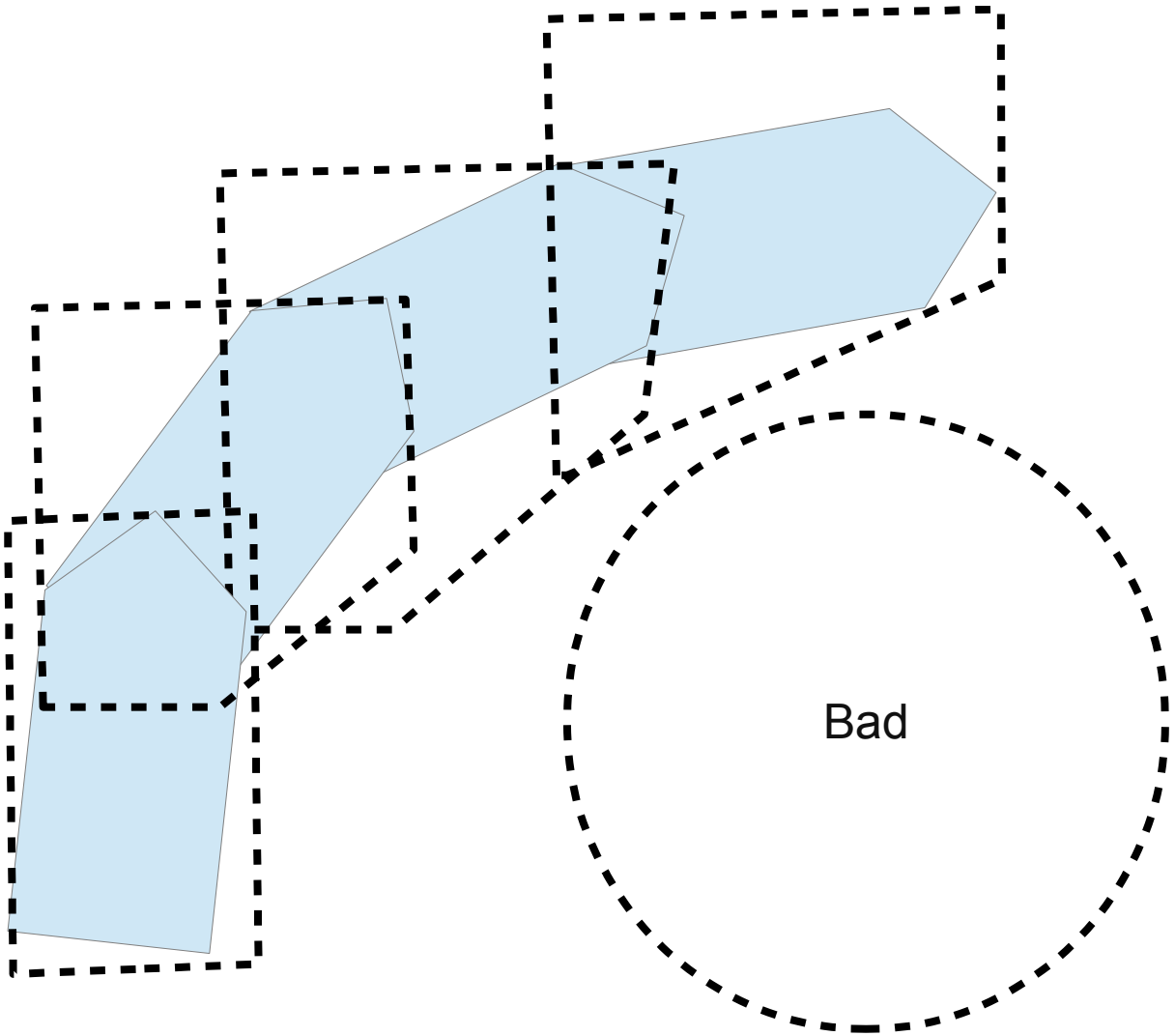


Template polyhedron of a set X
tightest template polyhedron
that encloses X

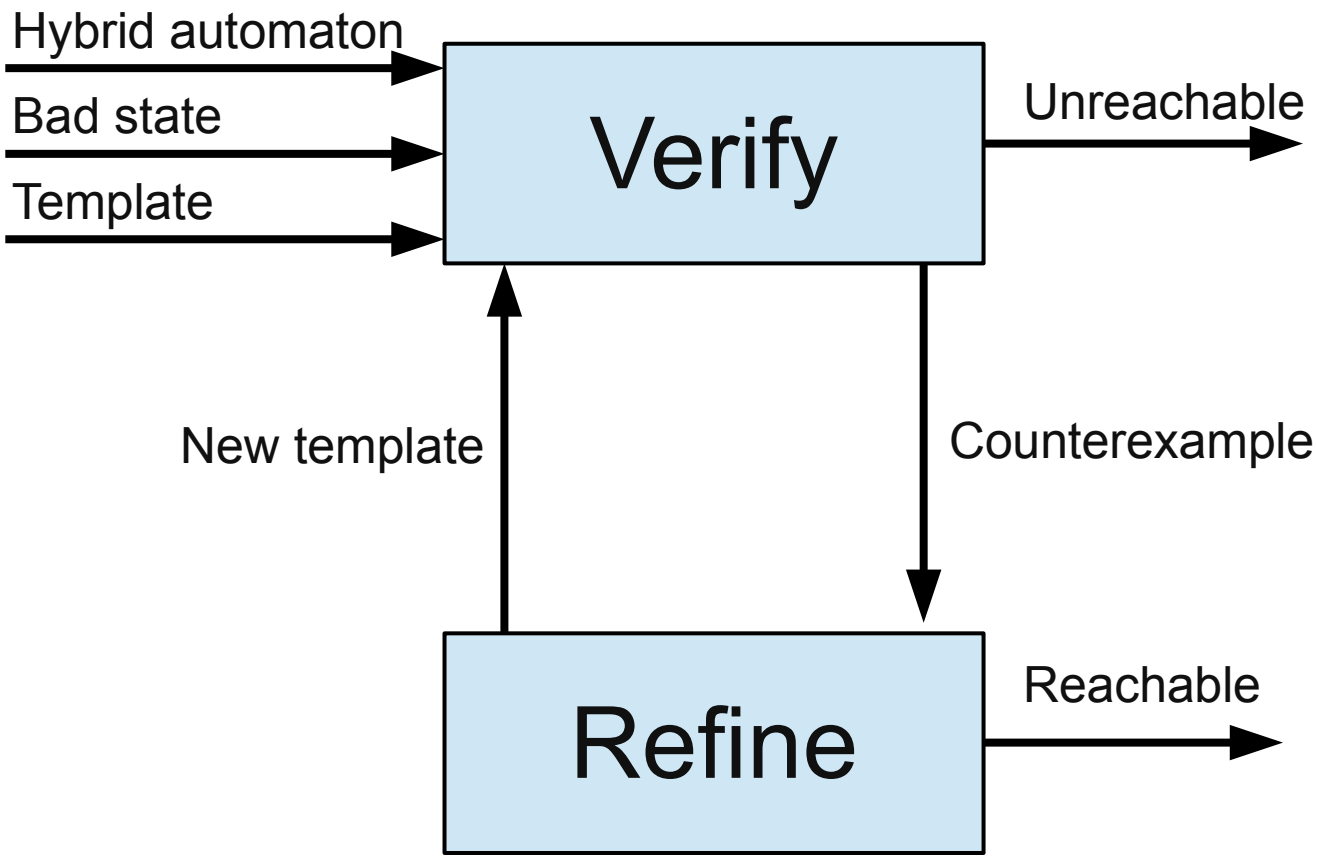
Template Polyhedra: Reachability Analysis



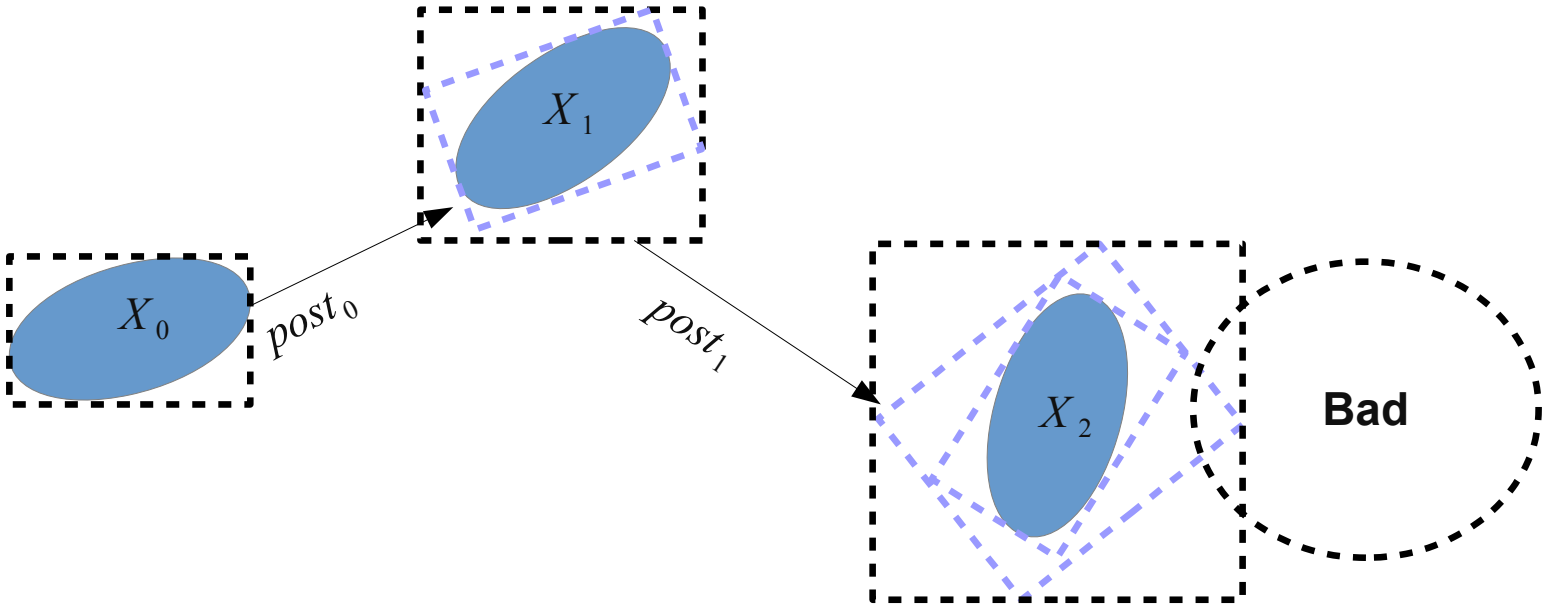
Template Polyhedra: Reachability Analysis



CEGAR With Template Polyhedra

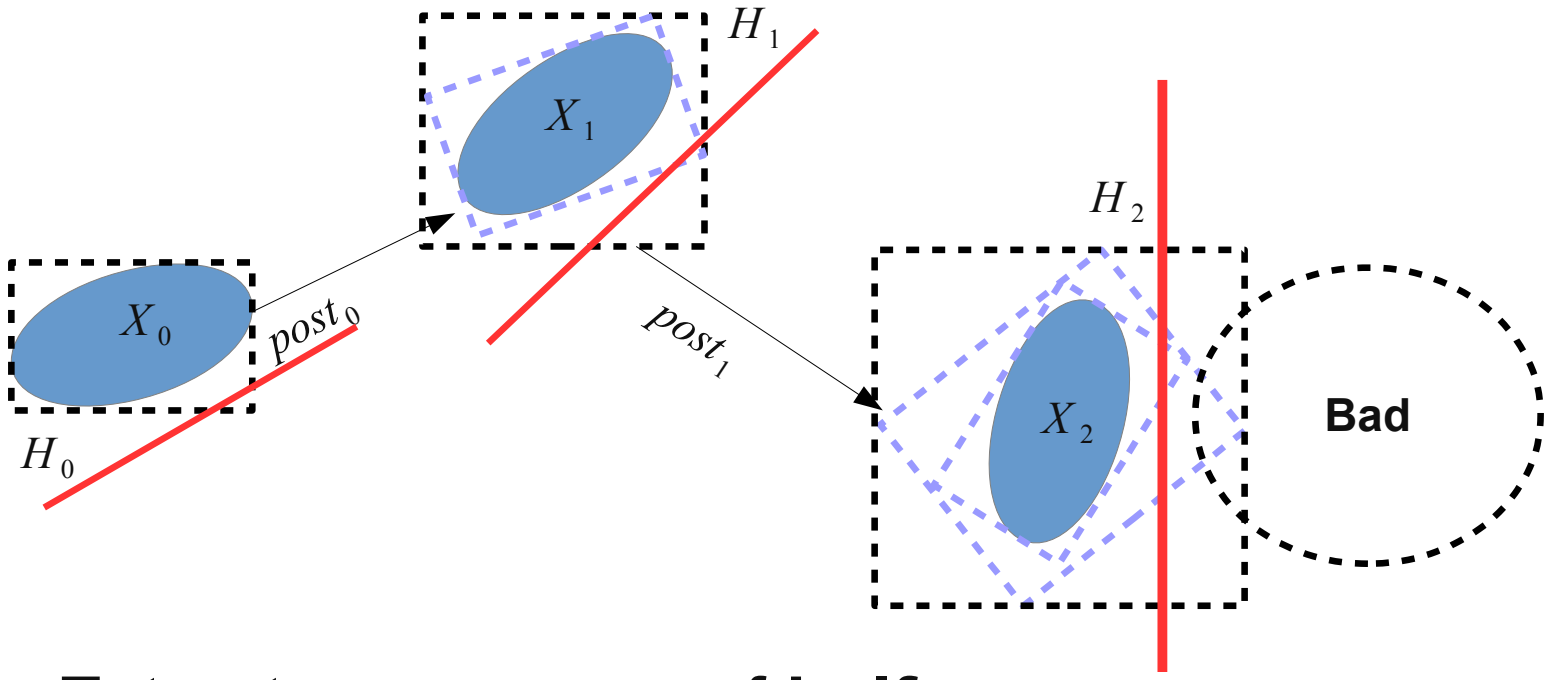


Template refinement by interpolation



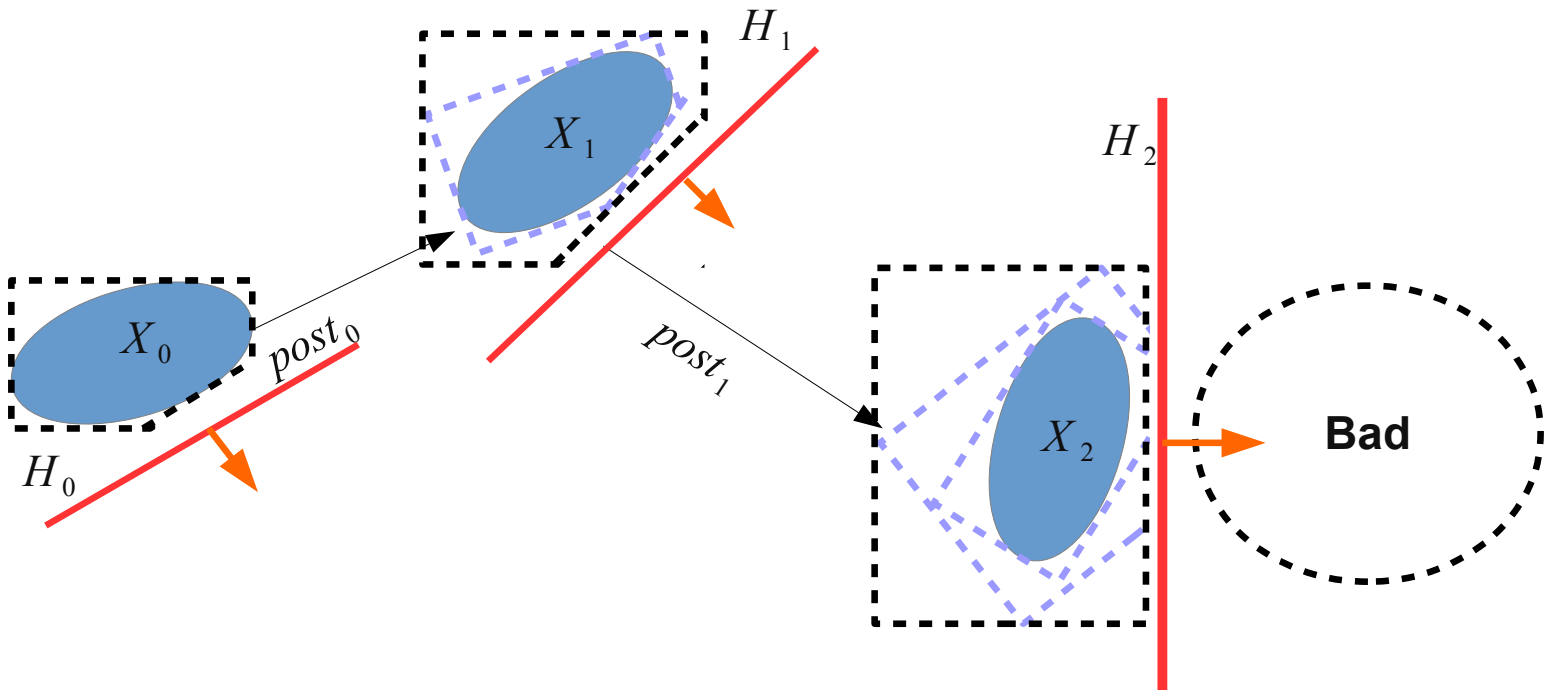
- Error accumulates (wrapping effect)
- Refinement must be inductive

Template refinement by interpolation



- Extract a sequence of **halfspaces** H_0, H_1, H_2 s.t. $X_0 \subseteq H_0, post(H_0) \subseteq H_1, post(H_1) \subseteq H_2$ and $H_2 \cap Bad = \emptyset$

Template refinement by interpolation



- Take the outward pointing directions of H_0, H_1, H_2
- Recompute the abstraction (excludes CE)

Another CPS Problem

Meltdown

Spectre

Conclusion

Formal Verification for Cyber-Physical Systems

- a cyber-physical system: embedded computing system which interacts with its physical environment
- for safety of CPS: simulation
- but formal verification does have a role to play
- challenges: tighness of approximations; state space explosion; curse of dimension; compositionality
- our interest: formal verification for distributed CPS
- example: swarm of AUVs which explore a bay