Reachability Analysis of Simulation Models with SpaceEx

Stefano Minopoli

VERIMAG - Grenoble

June 18, 2015

Outline

1. Introduction to SpaceEx Verification Platform
2. SpaceEx Verification Model (and May Semantics)
3. Simulation Models (and Must Semantics)
4. From Simulation to Verification Models
5. Example: from Simulink to SpaceEx
6. Future Work

Introduction to SpaceEx

SpaceEx Tool

- A verification **platform** for hybrid systems (continuous and discrete components which interact)
- To verify that a given **Verification Model** satisfies desired safety properties
  - By computing the sets of reachable states

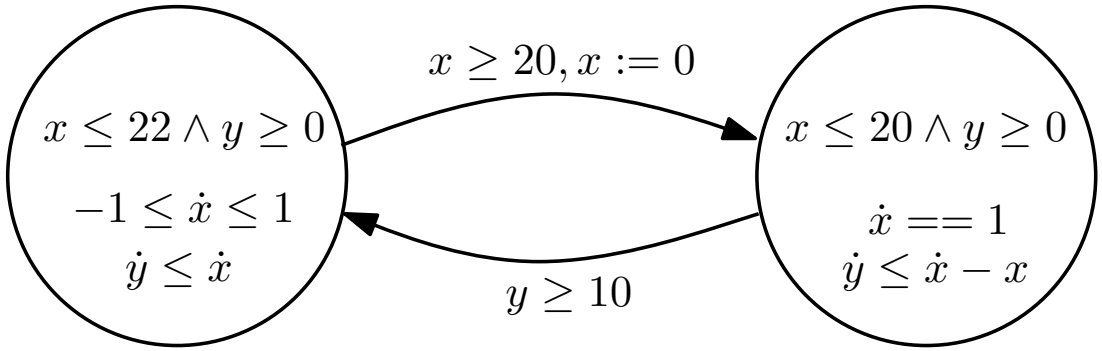
The SpaceEx Platform

- Graphical Model Editor, Analysis Core and Web Interface
- Designed to facilitate the implementation of algorithms for reachability and safety verification

Introduction to SpaceEx

SpaceEx Verification Model

- Similar to the *Hybrid Automata*

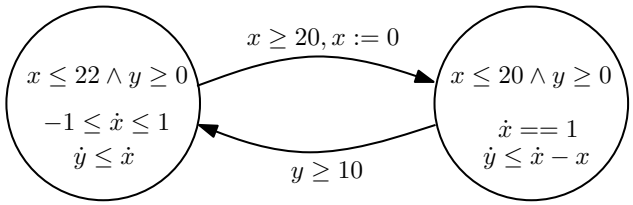


- Consists of one or more *Components*
  - Allowing Structured and Hierarchical models

Introduction to SpaceEx

SX Verification Model and Components

1. **Basic Component**: corresponds to a single HA



2. **Network Component**: one or more instantiation of other components (HA in parallel composition)
   - Recall: Hierarchy can be easily modeled

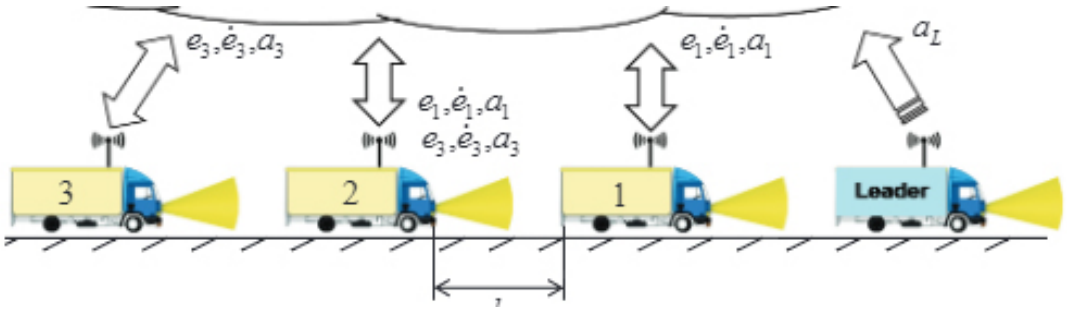
Introduction to SpaceEx

SpaceEx Analysis Core

- Implemented Reachability Algorithm: **Scenarios**
  1. Simulation: trajectory simulation using ODE solver
  2. PHAVer: for Linear Hybrid Automata (Piecewise Constant bounds on derivatives)
  3. LGG Support Function: variant of the Le Guernic Girard algorithm. For Piecewise Affine Dynamics with nondeterministic inputs
  4. STC Support Function: an enhancement of LGG with automatic clustering

Example of SX Verification Model

Networked Cooperative Platoon of Vehicles (ARCH 2014 Benchmark)



- Three controlled vehicles with a manually driven leader
- The vehicles exchange information
- The communication network may be subjected to failure (total loss of communication)
- The leader can proceed by changing speed

Example of SX Verification Model

Networked Cooperative Platoon of Vehicles (ARCH 2014 Benchmark)

- Determine the minimum allowable safe gaps (e_i) among the vehicles
- Reachability analysis to establish the minimum value reachable for e_i

Example of SX Verification Model

Networked Cooperative Platoon of Vehicles (ARCH 2014 Benchmark)

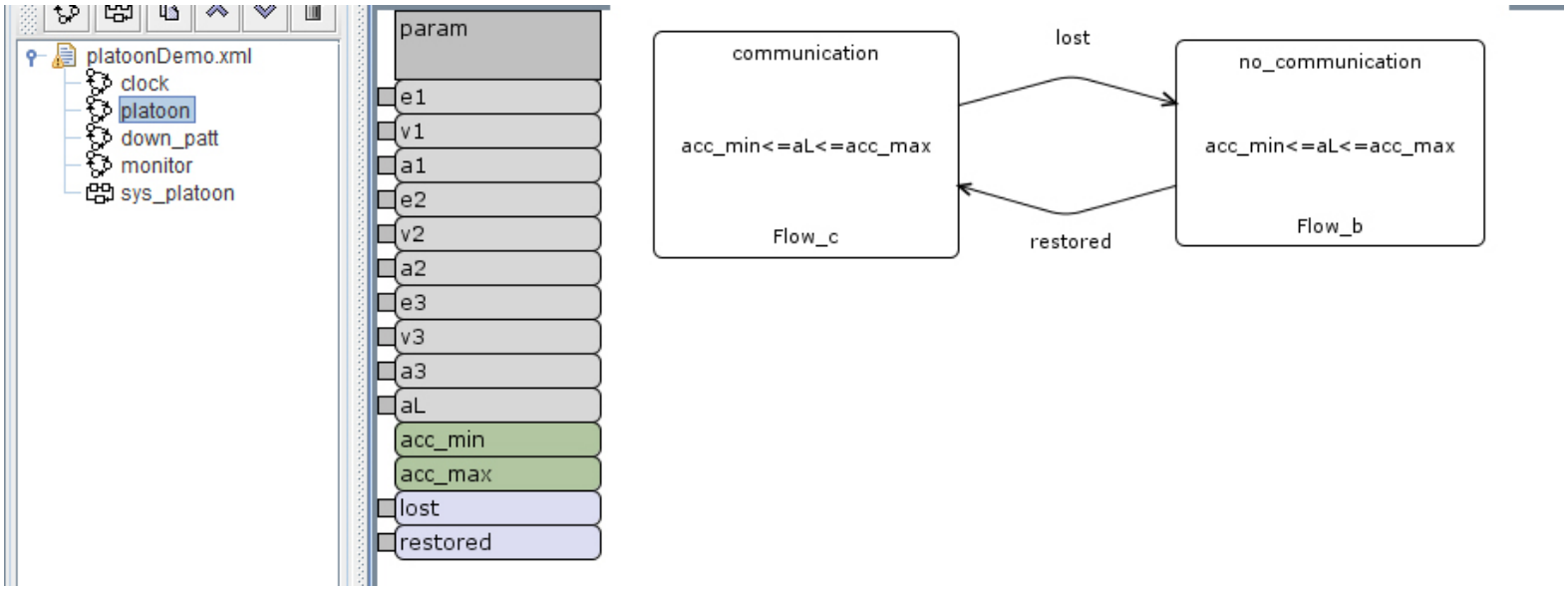


Figure: Basic Component for Vehicles

Example of SX Verification Model

Networked Cooperative Platoon of Vehicles

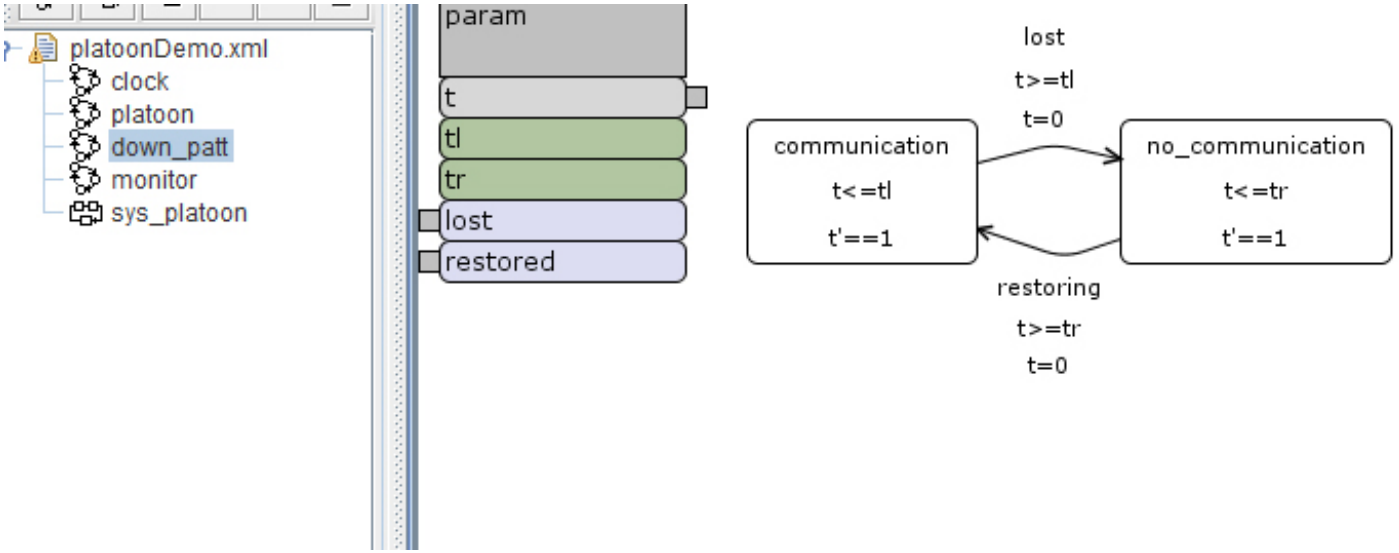


Figure: Basic Component for Breakdown Pattern

Example of SX Verification Model

Networked Cooperative Platoon of Vehicles

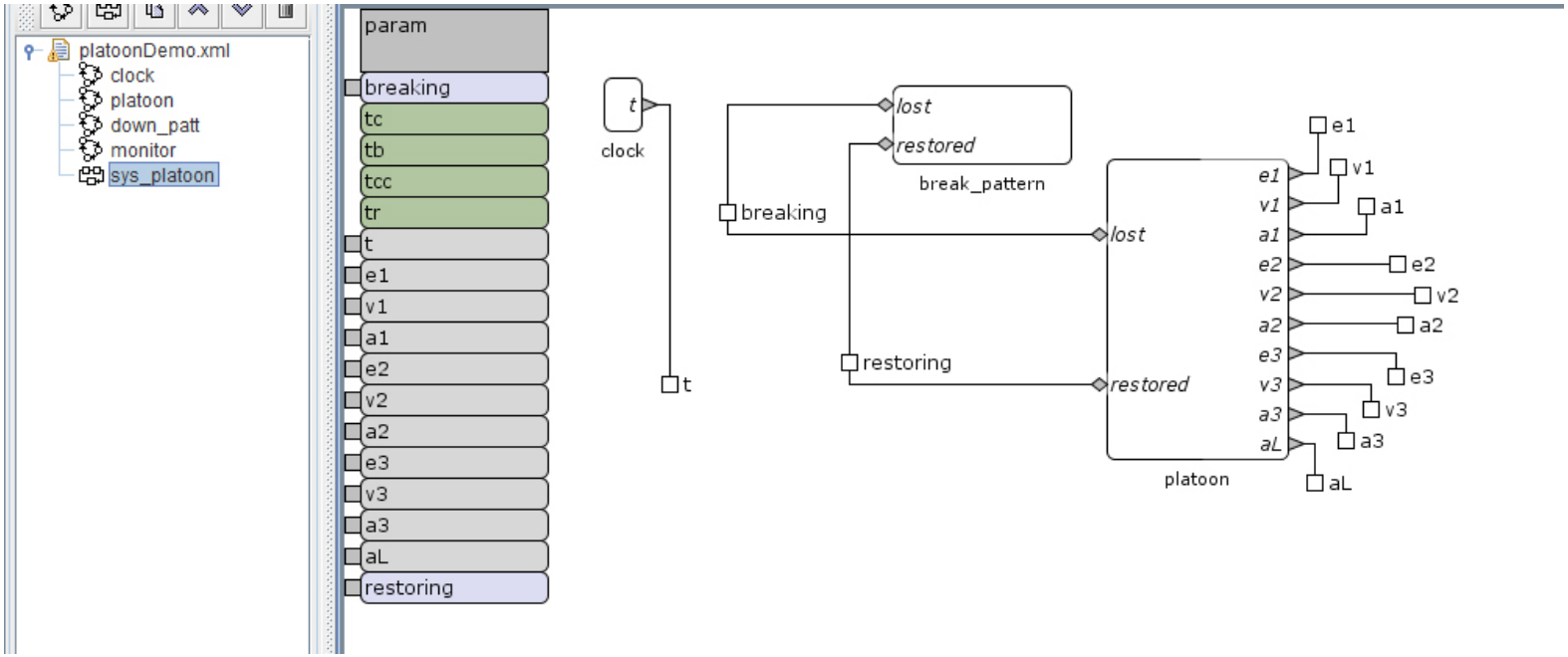


Figure: Network Component for Main System

Performing Reachability Analysis

Parameters

- Breakdown: may happen every $[20, 22]$ sec (note: interval and may)
- Restore: every 20 sec.

Reachability Analisys with STC Scenario

- Reachability Result: minimum value $e_3 = -18.42$
- (the minimum safe distance between second and third vehicle is $18.42m$)

Performing Reachability Analysis

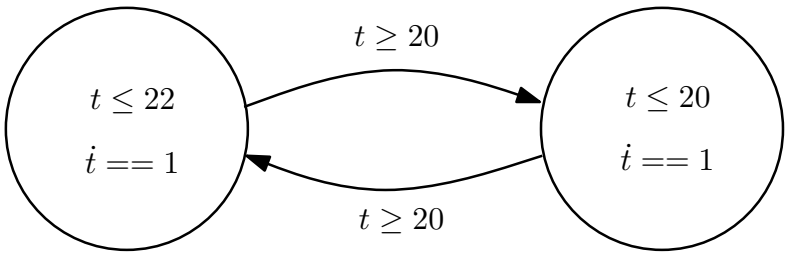
Parameters

- Breakdown: may happen every $[20, 22]$ sec (note: interval and may)
- Restore: every 20 sec.

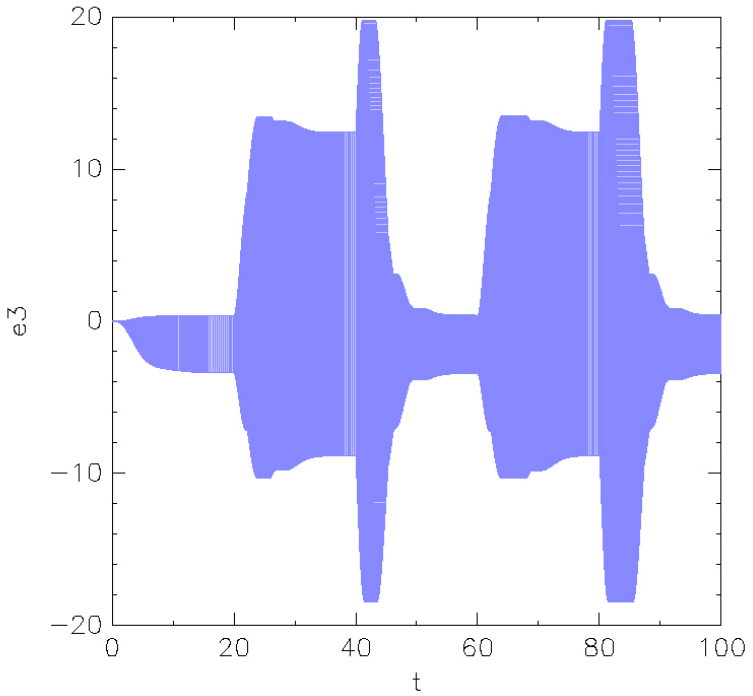
Reachability Analisys with STC Scenario

- Reachability Result: minimum value $e_3 = -18.42$
- (the minimum safe distance between second and third vehicle is $18.42m$)

Reachable States

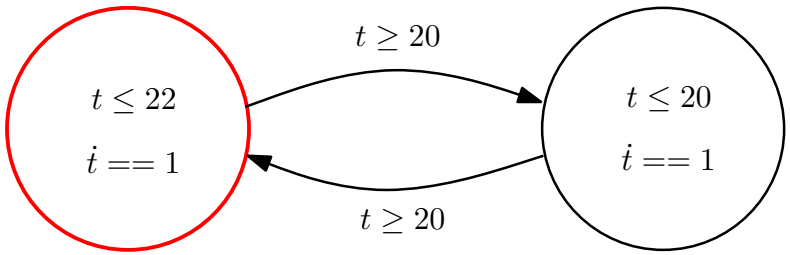


4.1

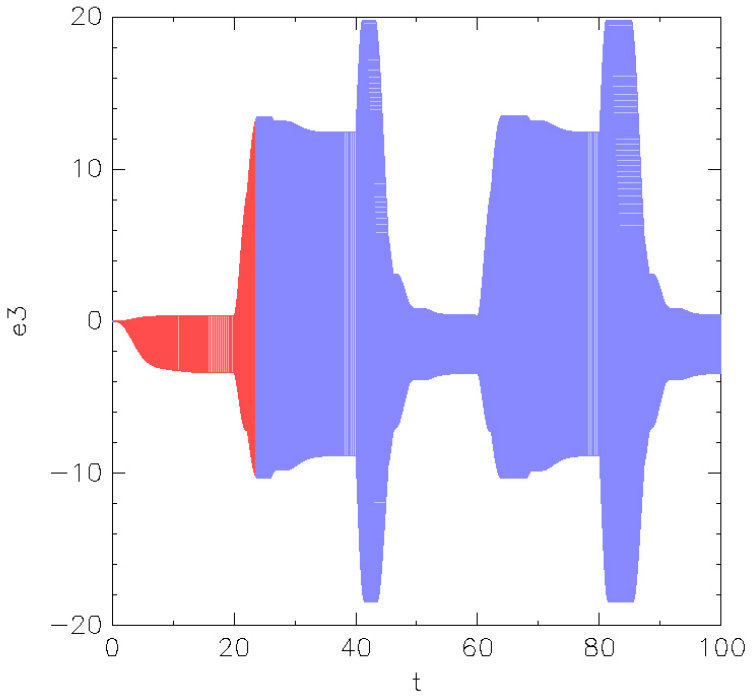


4.2

Reachable States

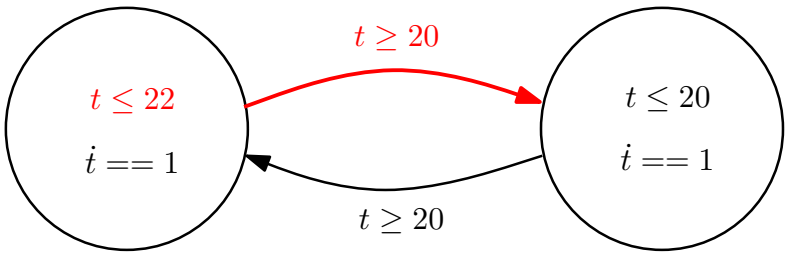


4.3

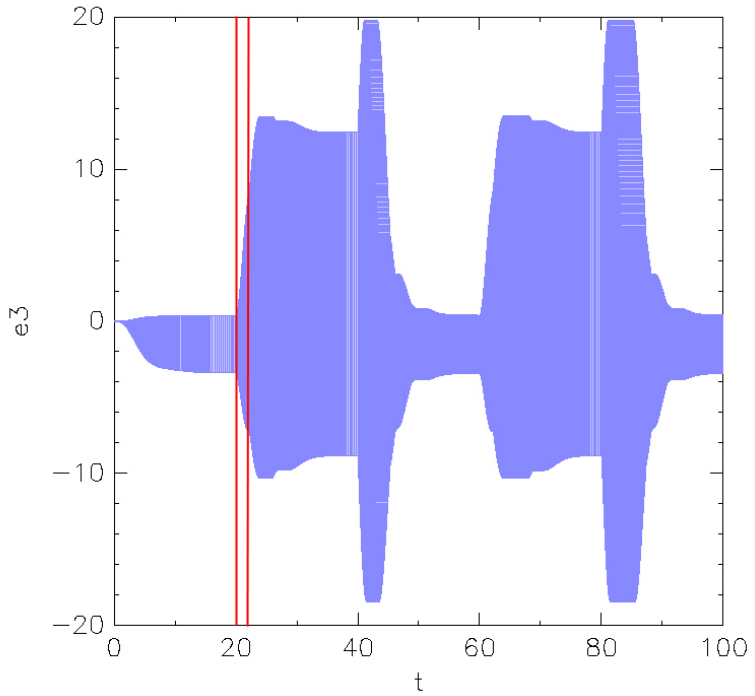


4.4

Reachable States

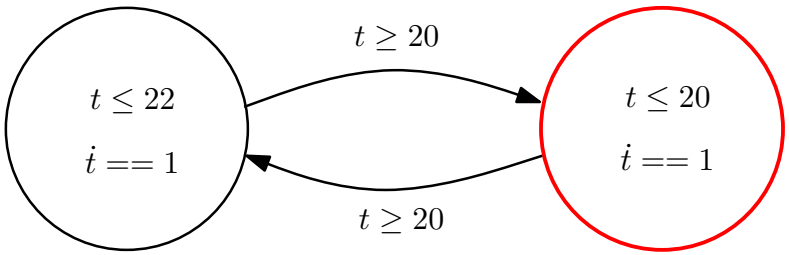


4.5

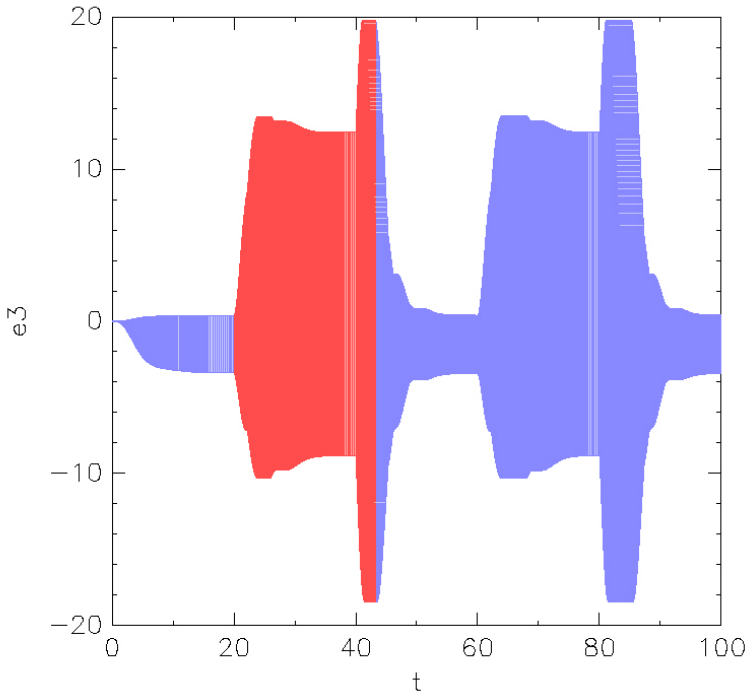


4.6

Reachable States

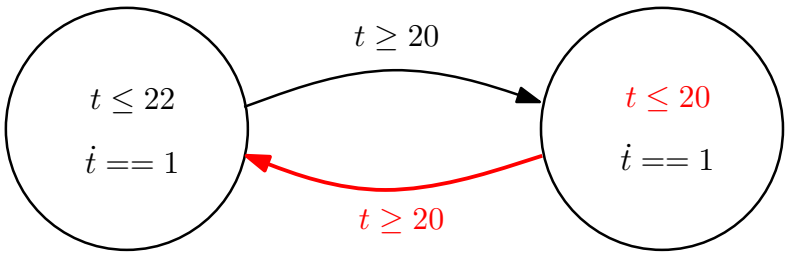


4.7

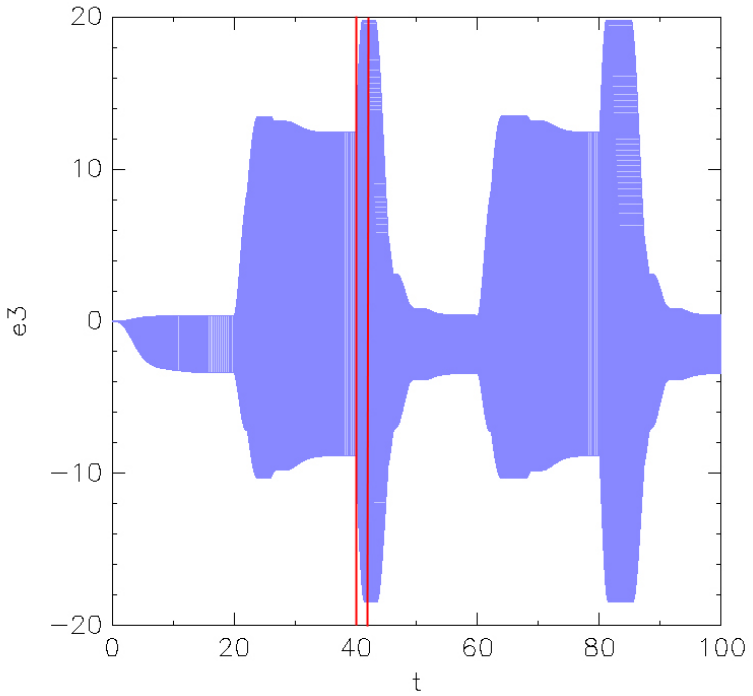


4.8

Reachable States

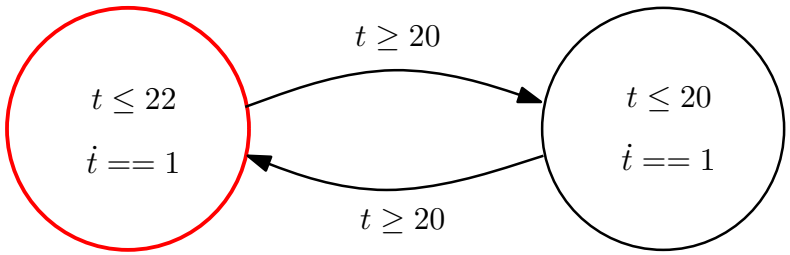


4.9

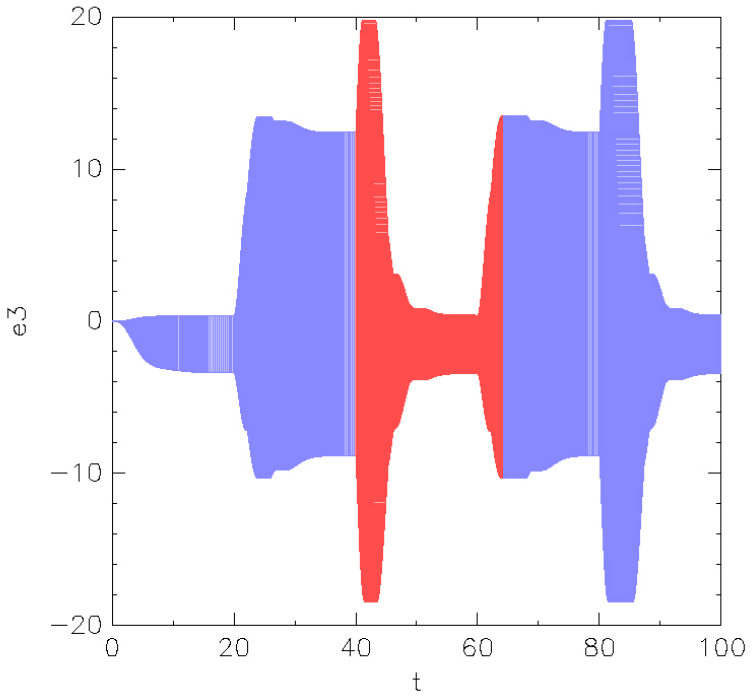


4.10

Reachable States

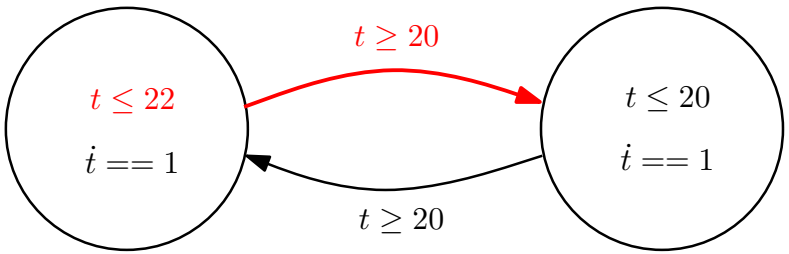


4.11

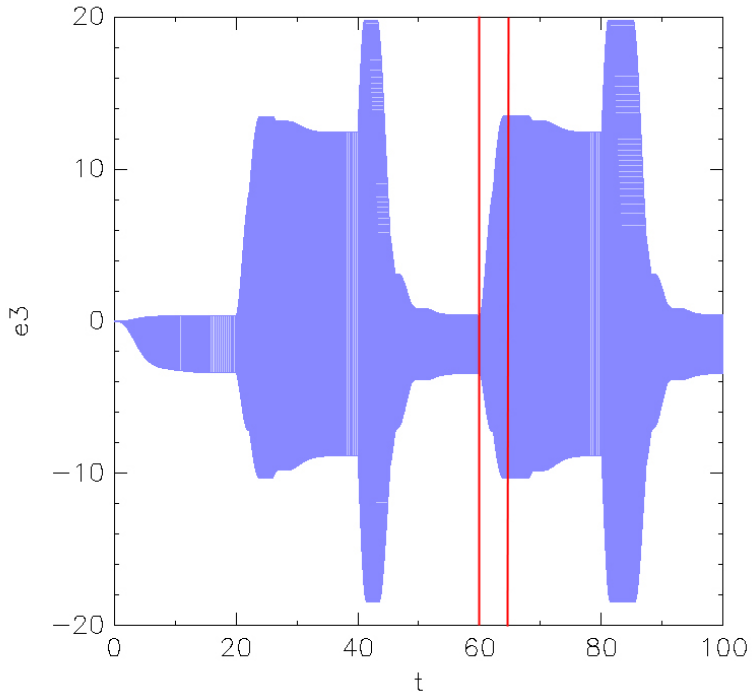


4.12

Reachable States

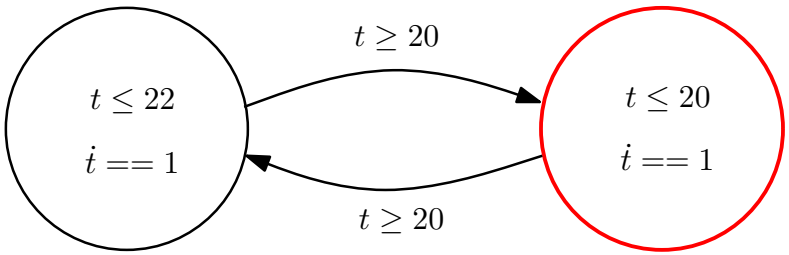


4.13

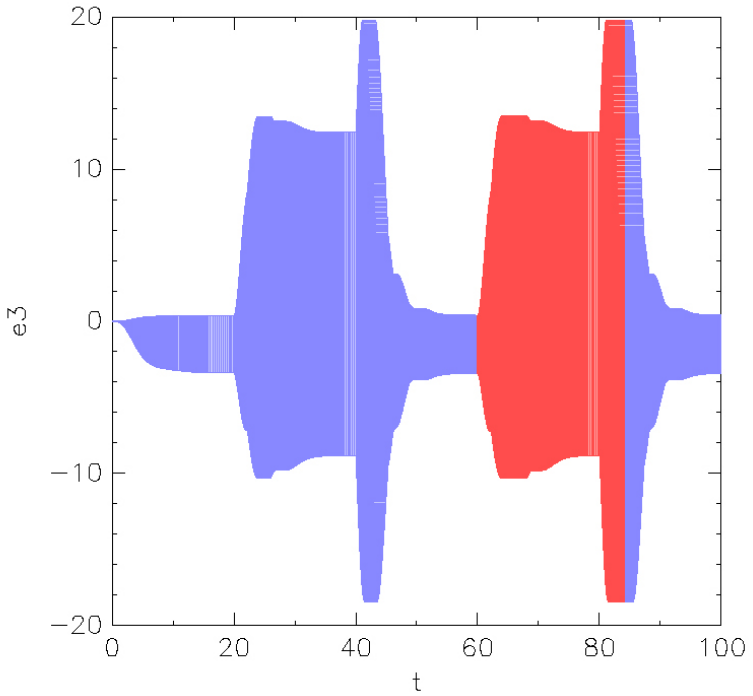


4.14

Reachable States

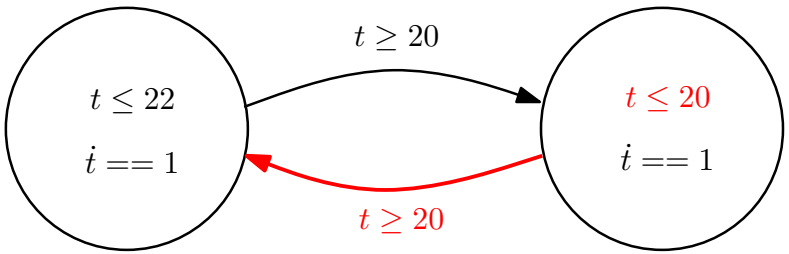


4.15

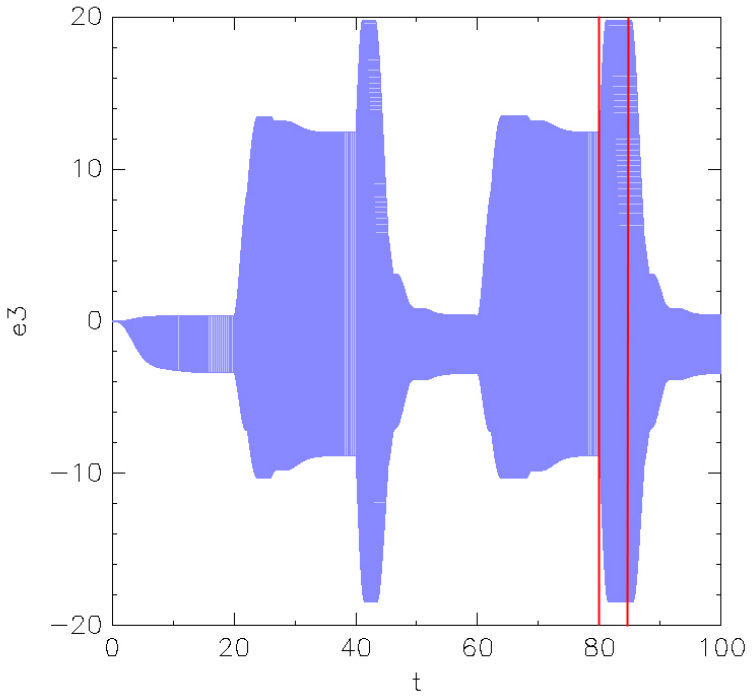


4.16

Reachable States

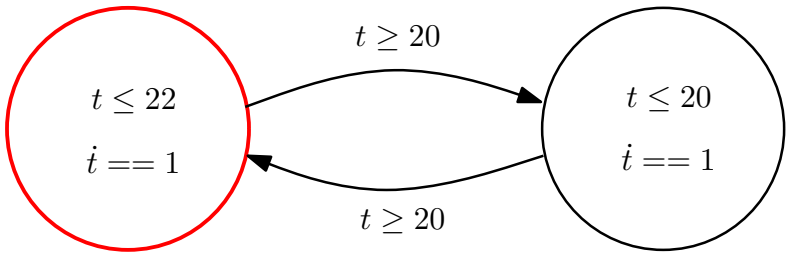


4.17

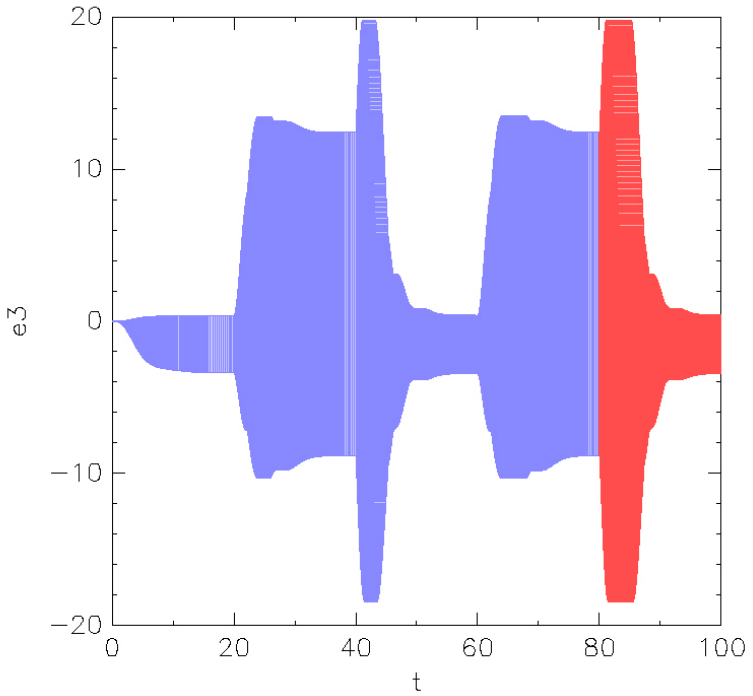


4.18

Reachable States



4.19



4.20

Verification Models

Sources of Non-Determinism

1. Initial Conditions: *Space* of States
2. Dynamics (polyhedra inclusion, perturbations, ...)
3. Transitions: *may* be taken when guards are satisfied (**May Semantics**)
   - Reachable States: *infinite* trajectories

Verification Models

Sources of Non-Determinism

1. Initial Conditions: *Space* of States
2. Dynamics (polyhedra inclusion, perturbations, ...)
3. Transitions: *may* be taken when guards are satisfied (**May Semantics**)
   - Reachable States: *infinite* trajectories

Simulation World

Numerical Simulation Tools

- Widely used in industry
- Validation of systems in model-based design methodology
  - Simulink by MathWorks, Modelica (which are the de-facto standard in many industries)
  - Ptolemy (academic formalism)
  - ...
- Systems designed by **Simulation Models**
- ODE Solvers

Simulation Models

No-source of non-determinism

1. Initial States: *single* point in the space
2. Dynamics
3. Transitions: **MUST** be taken (ASAP) when guards are satisfied (**Must Semantics**)
   - Reachable states: deterministic trajectory
     - Limited analysis

Simulation Models

No-source of non-determinism

1. Initial States: *single* point in the space
2. Dynamics
3. Transitions: **MUST** be taken (ASAP) when guards are satisfied (**Must Semantics**)
   - Reachable states: deterministic trajectory
     - Limited analysis

Simulation Models

No-source of non-determinism

1. Initial States: *single* point in the space
2. Dynamics
3. Transitions: **MUST** be taken (ASAP) when guards are satisfied (**Must Semantics**)
   - Reachable states: deterministic trajectory
     - Limited analysis

From Simulation to Verification

Performing Verification of existing Simulation Models

- To allow exhaustive analysis
  - Verification Models as abstraction of Simulation Models

Main Issues

- Not all the deterministic aspects can be expressed by Verification Models
- Manually rewriting all the existing (simulation) models could be no feasible

From Simulation to Verification

Solving Issues / 1

- Initial States: a single point in the space
  - (Trivial) Polyhedra and Zonotopes for states space allow to model single points
- Deterministic Dynamics
  - (Trivial) By non-deterministic dynamics

From Simulation to Verification

Solving Issues / 2

- Transitions: **MUST** be taken (ASAP) when guards are satisfied (**Must Semantics**)
  - Must semantics can not be directly modeled by may semantics
  - WHY?

From Simulation to Verification

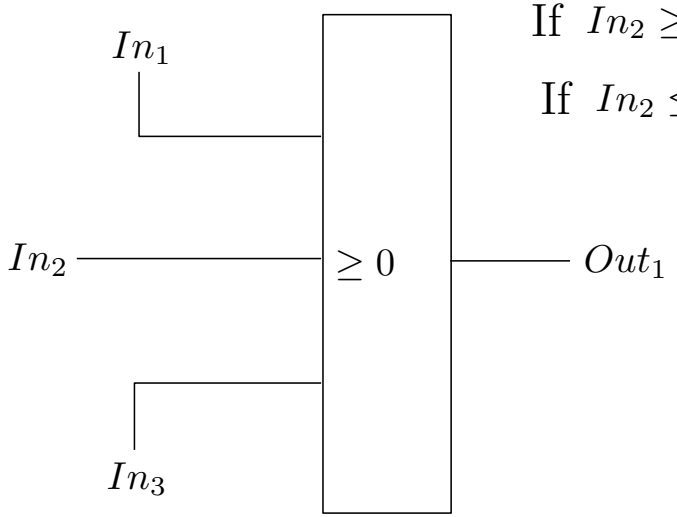
Solving Issues / 2

- Transitions: **MUST** be taken (ASAP) when guards are satisfied (**Must Semantics**)
  - Must semantics can not be directly modeled by may semantics
  - WHY?

Must Semantics: Example

Simulink Switch

- Switch: common block to model a discrete jump by Simulink

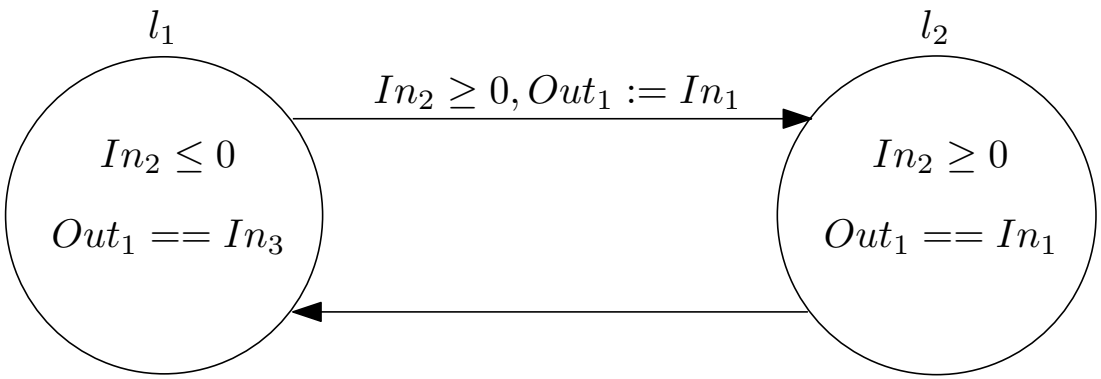


If $In_2 \geq 0$ Then $Out_1 = In_1$

If $In_2 \leq 0$ Then $Out_1 = In_3$

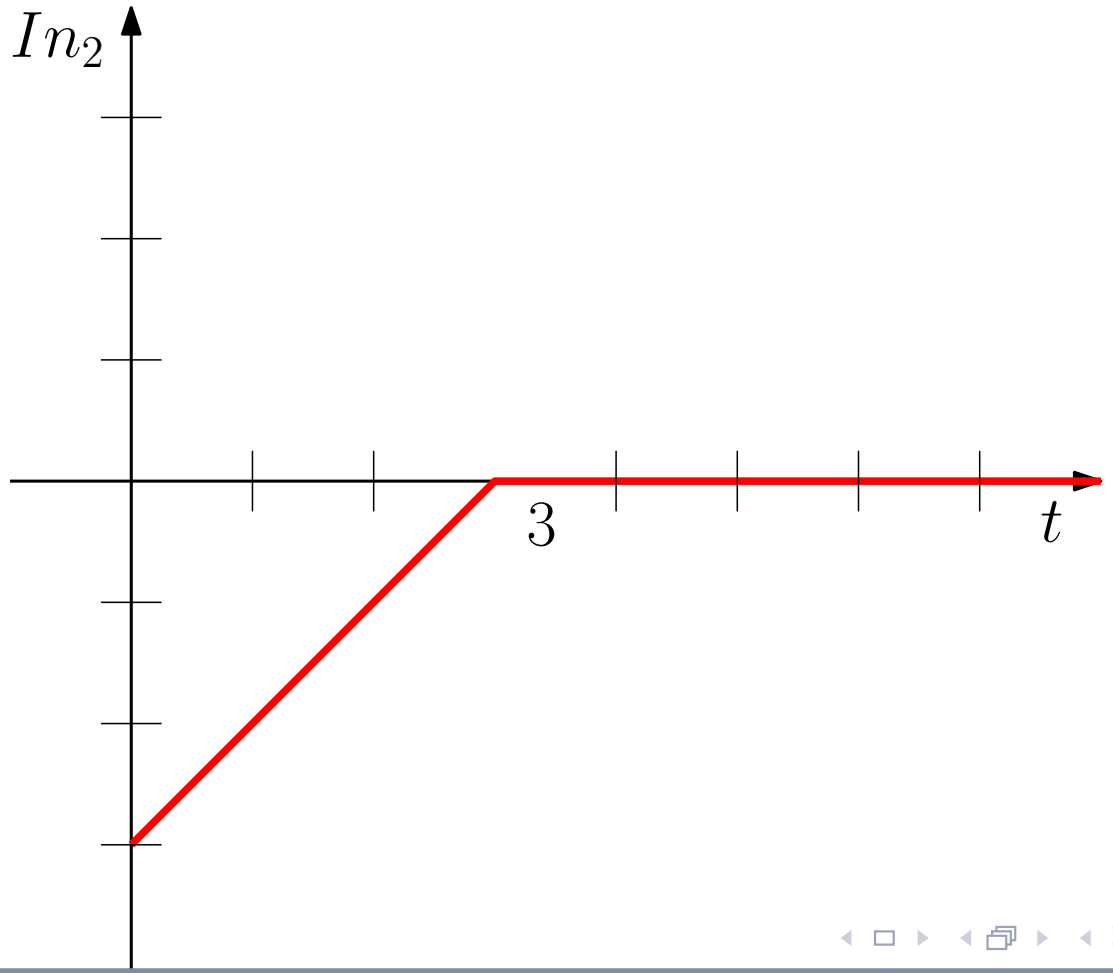
Modeling Must Semantics by HA

HA for Switch



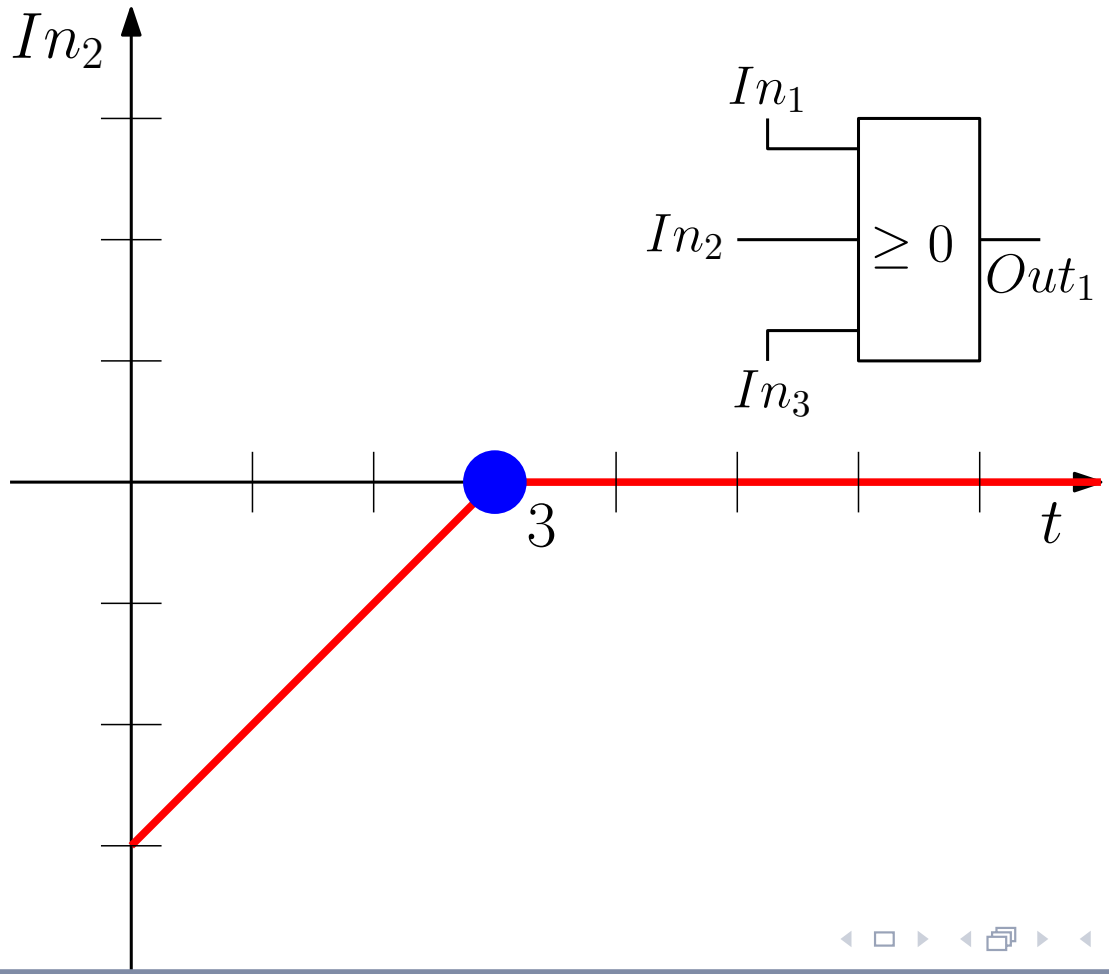
Modeling Must Semantics by HA

Case with zero-derivative



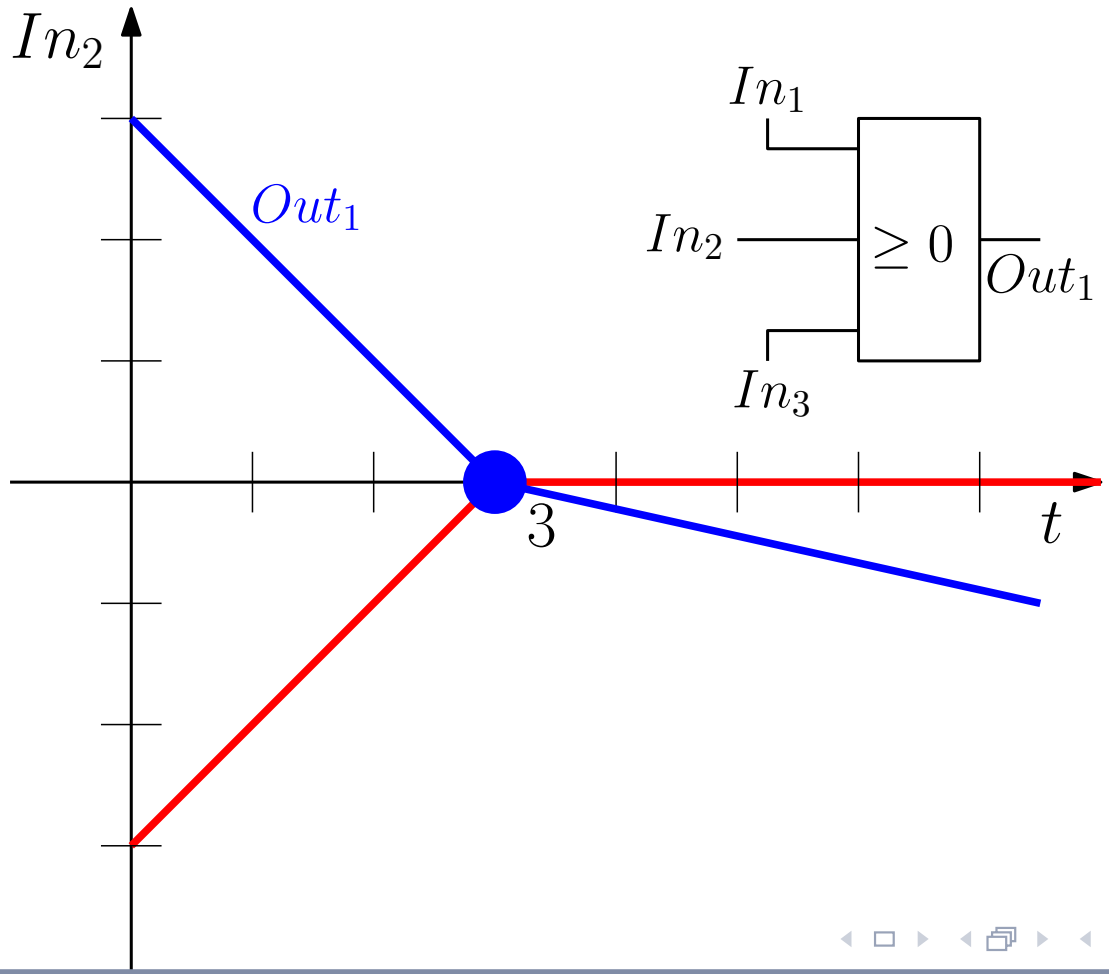
Modeling Must Semantics by HA

Case with zero-derivative



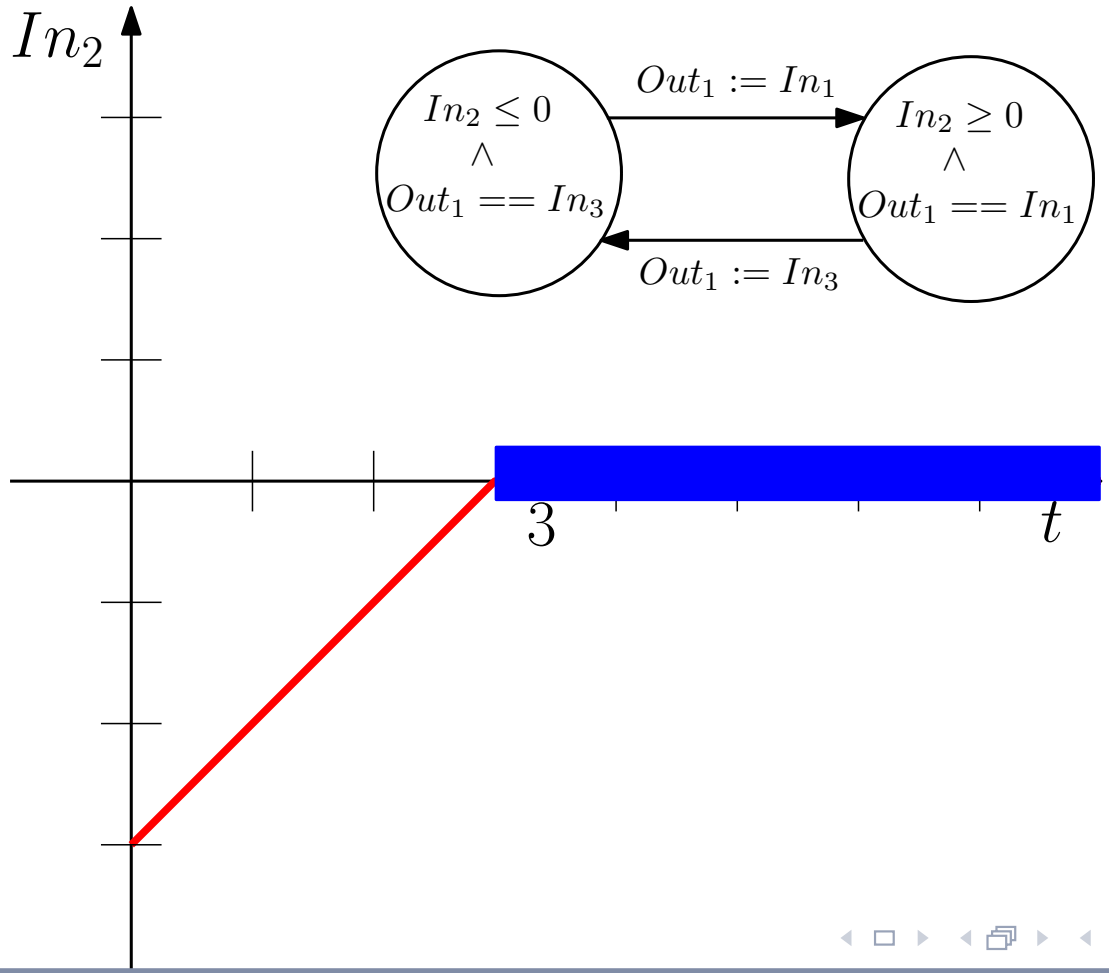
Modeling Must Semantics by HA

Case with zero-derivative



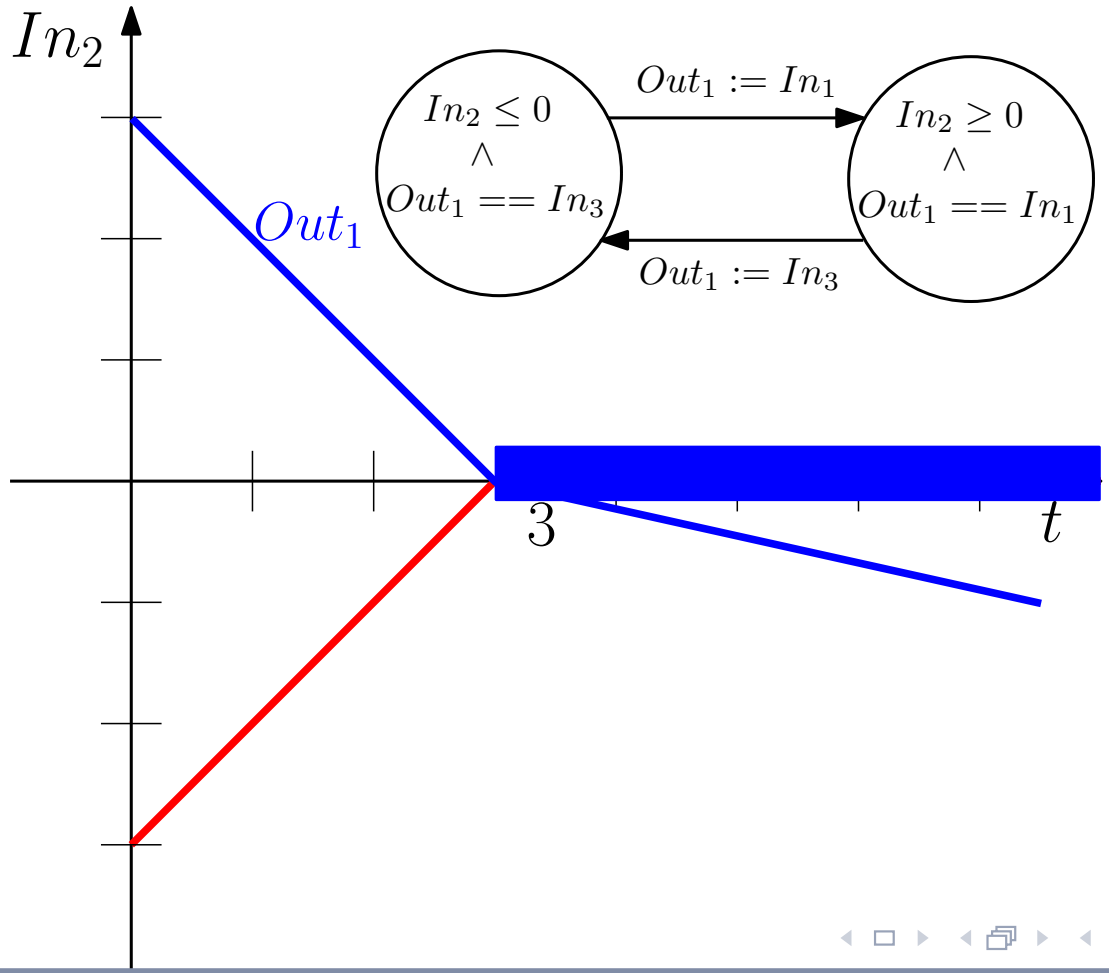
Modeling Must Semantics by HA

Case with zero-derivative



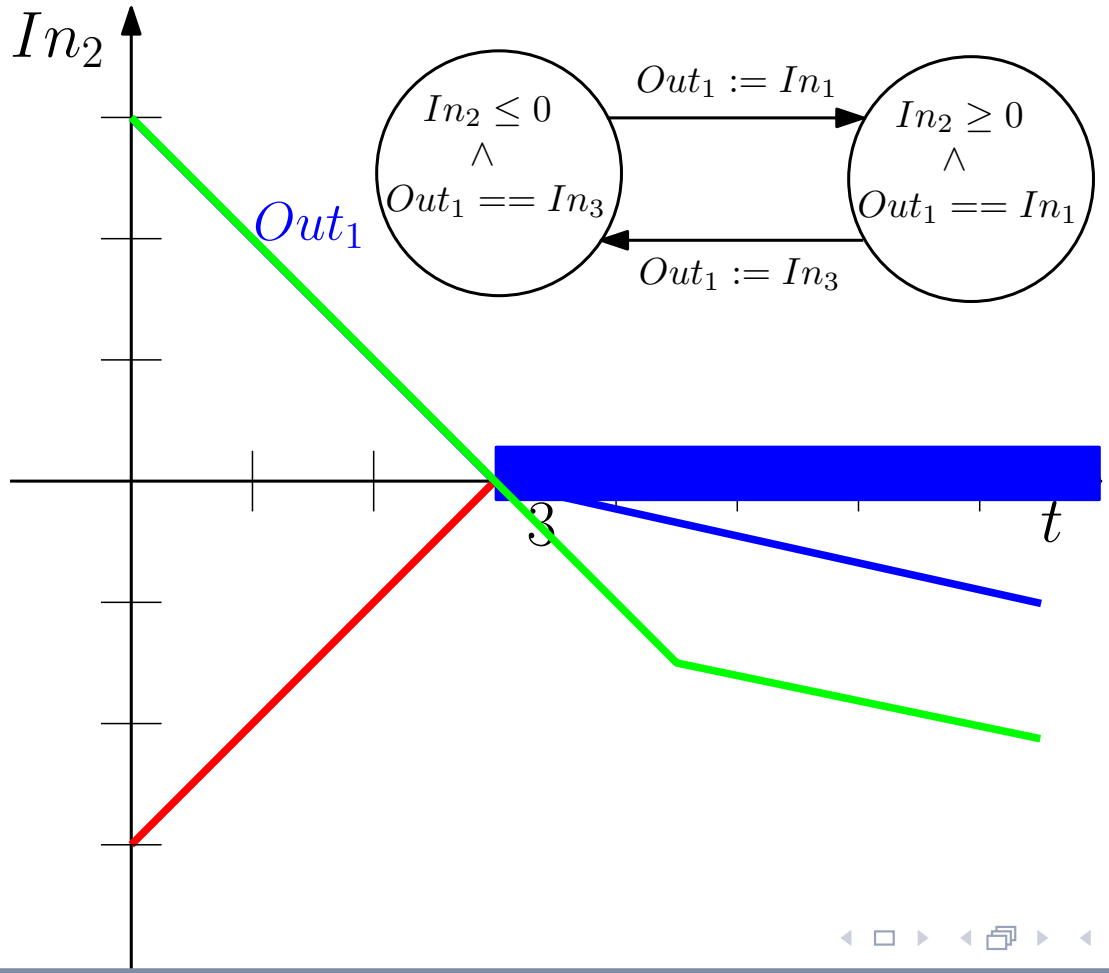
Modeling Must Semantics by HA

Case with zero-derivative



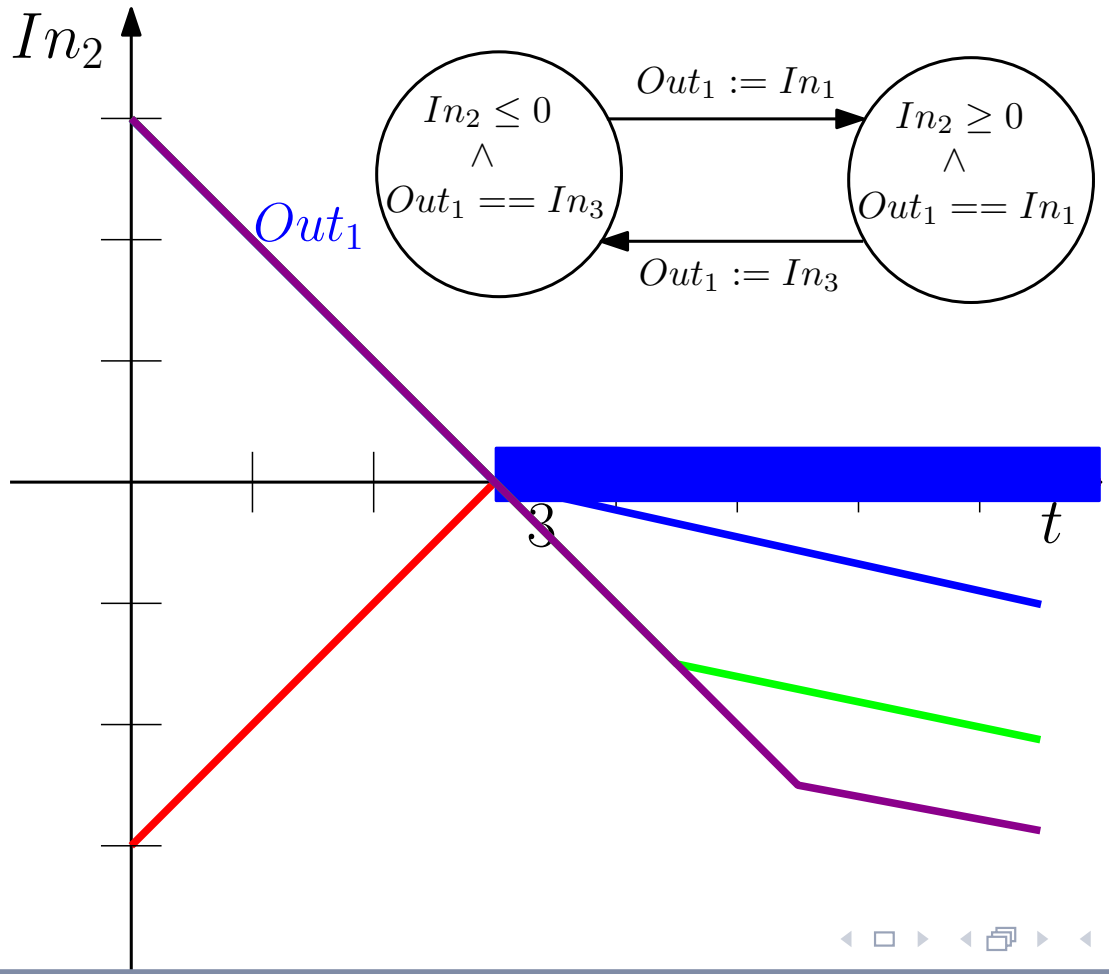
Modeling Must Semantics by HA

Case with zero-derivative



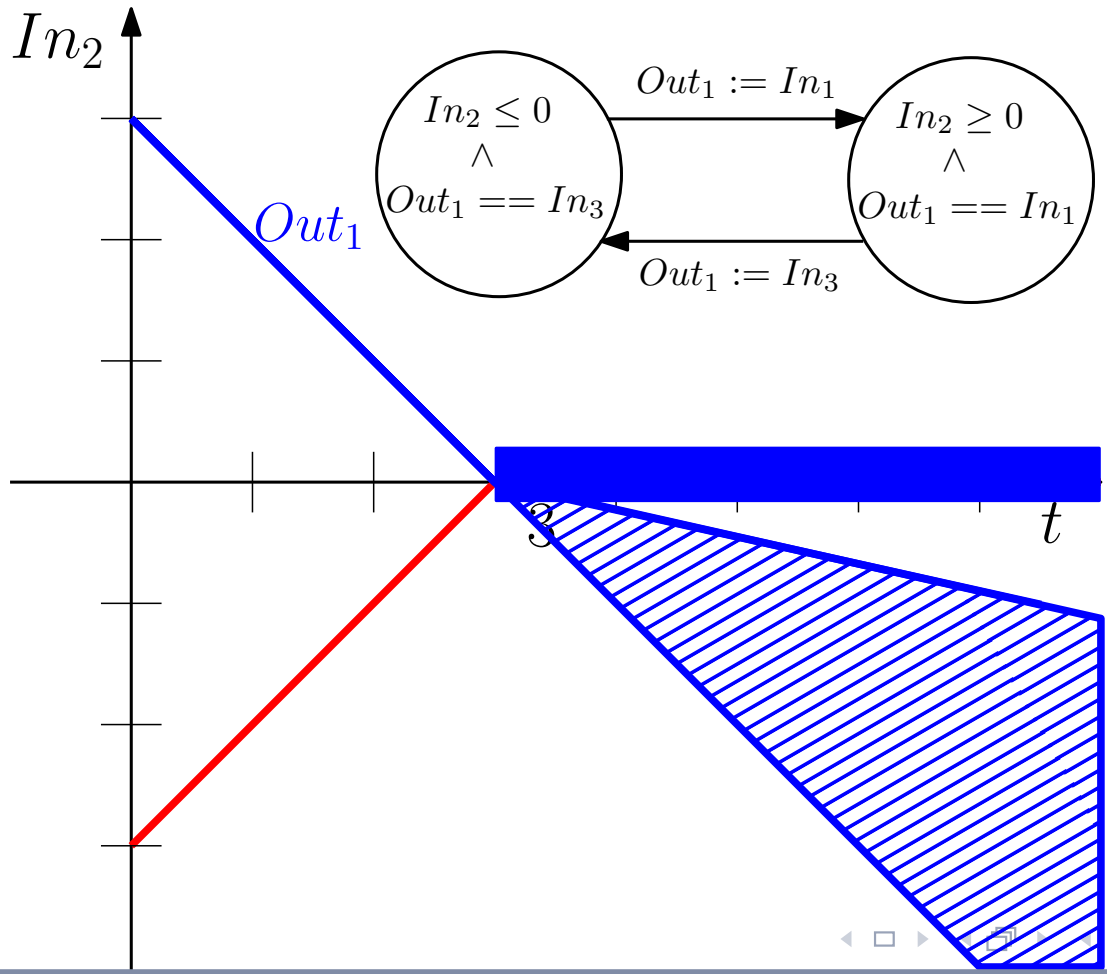
Modeling Must Semantics by HA

Case with zero-derivative



Modeling Must Semantics by HA

Case with zero-derivative



Previous Solutions

HA to model must semantics

- Adding extra locations and extra variables

1. More complicate (State Space Explosion due to the extra vars and locs)
2. Loss of structure (Due to the extra vars and locs)
3. Loss of hierarchy (Need for the flatten automaton)
4. ...

Our Proposal

HA with Urgent Conditions

- Each location can be associated with a Urgent Condition
  - Expressed by Polyhedra (Finite Union of Convex Polys)
  - Union of the outgoing guards of Urgent Transitions

Reachability Algorithm for LHA with Urgency

- Computation of the time elapse UNTIL urgent condition is meet
  - Time Elapse for each convex component of the Complement of the Urgent Condition

Our Solution

Pros

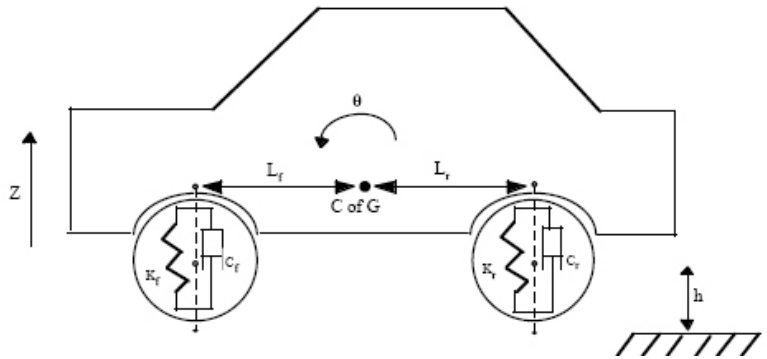
1. + Allows to easily model must semantics (via urgent conditions)
   - + Non-convex invariants "for free"
2. + Preserves the hierarchy
3. + Allows to easily translate from simulation to verification models (automatic)
4. + Formal verification on Simulation Models

Cons

- - Currently limited to Linear Hybrid Automata

Reachability Analysis of Simulink Diagram

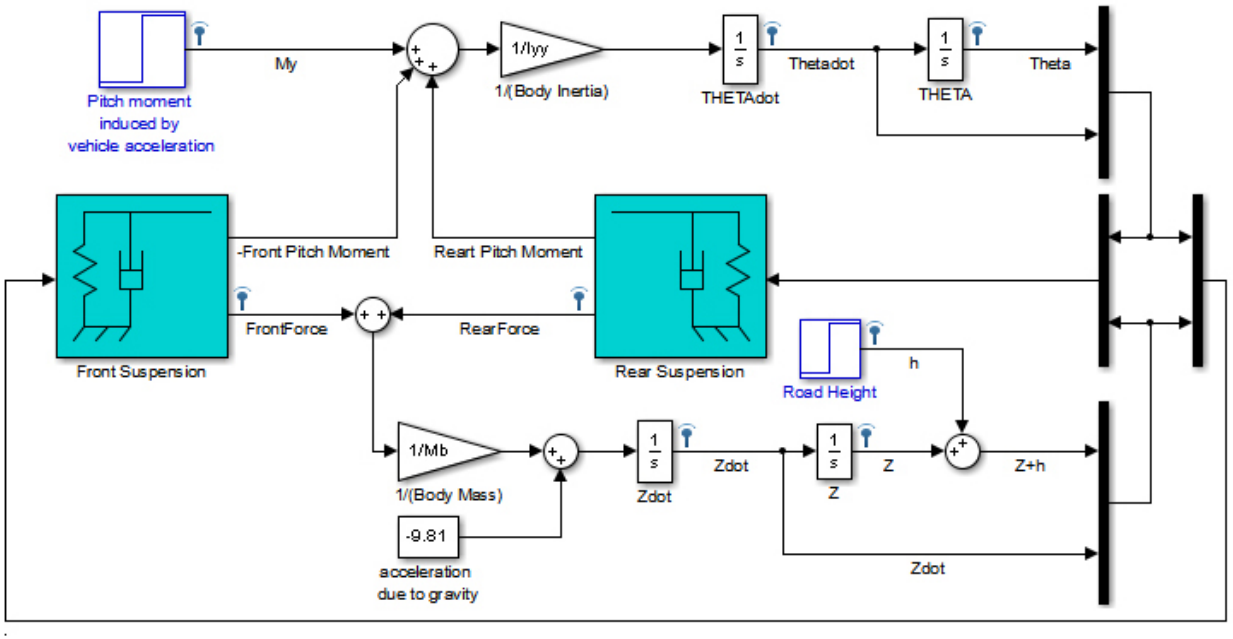
Simulink Diagram for Automotive Suspension



- Front and rear suspension modeled as spring/damper systems
- The vehicle body has pitch (from braking or acceleration maneuvers) and bounce degrees of freedom

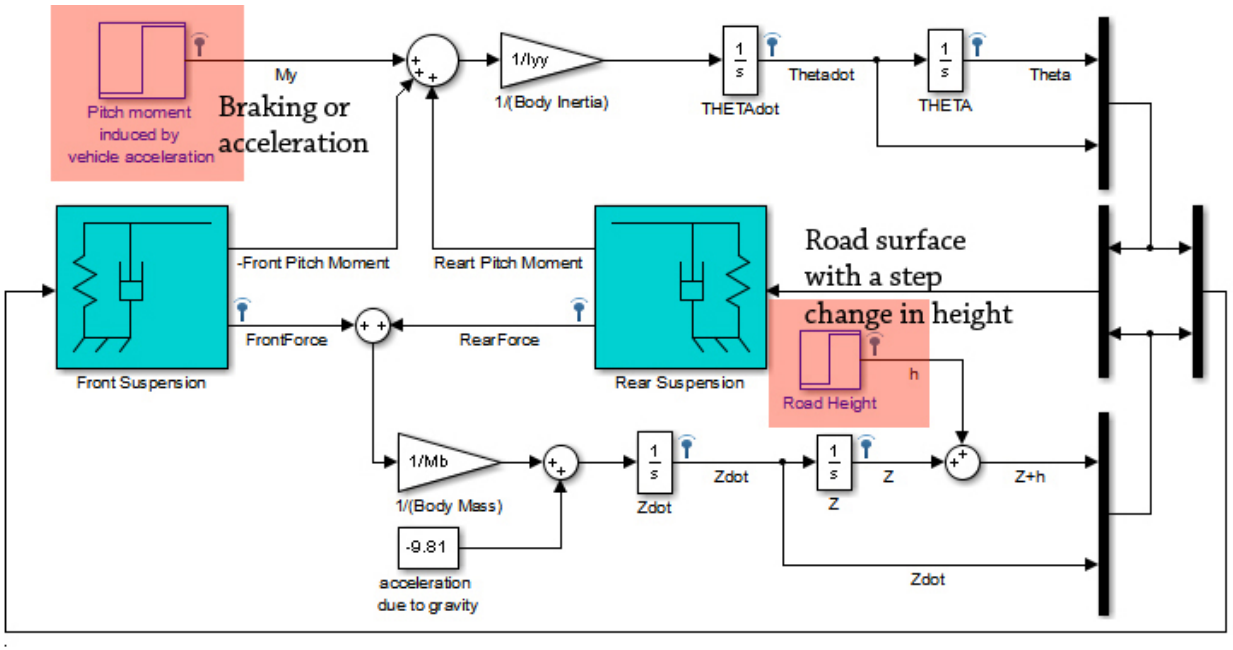
Example

Simulink Diagram for Automotive Suspension



Example

Simulink Diagram for Automotive Suspension



Parameters

Pitch by acceleration

- 0 during the first 3 seconds
- 100 after

Road surface with a step change in height

- 0 during the first 7 seconds
- 0.01 after

Initial Conditions

- Vertical Displacement $z = -0.12m$ (depending on the body mass)

Performing Numerical Simulation

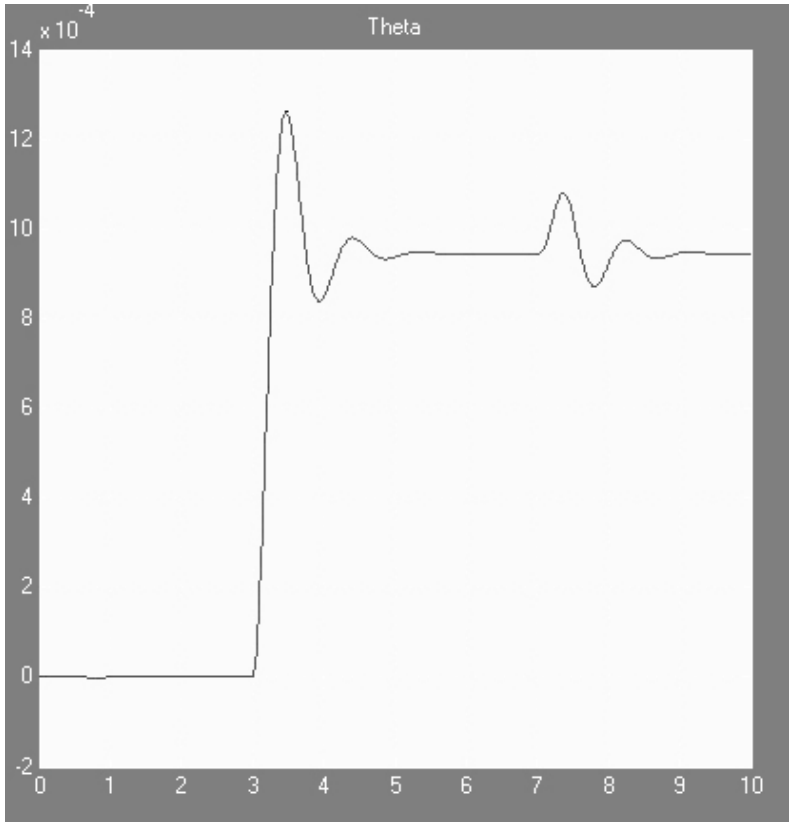
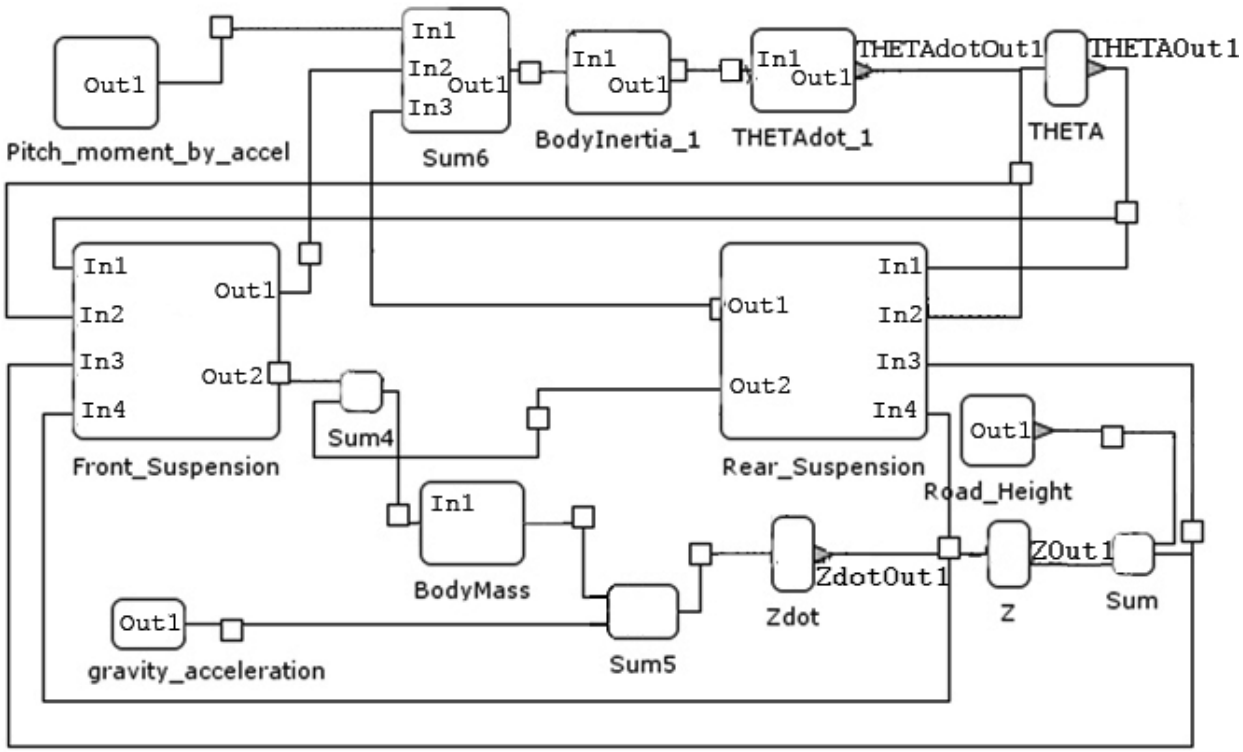


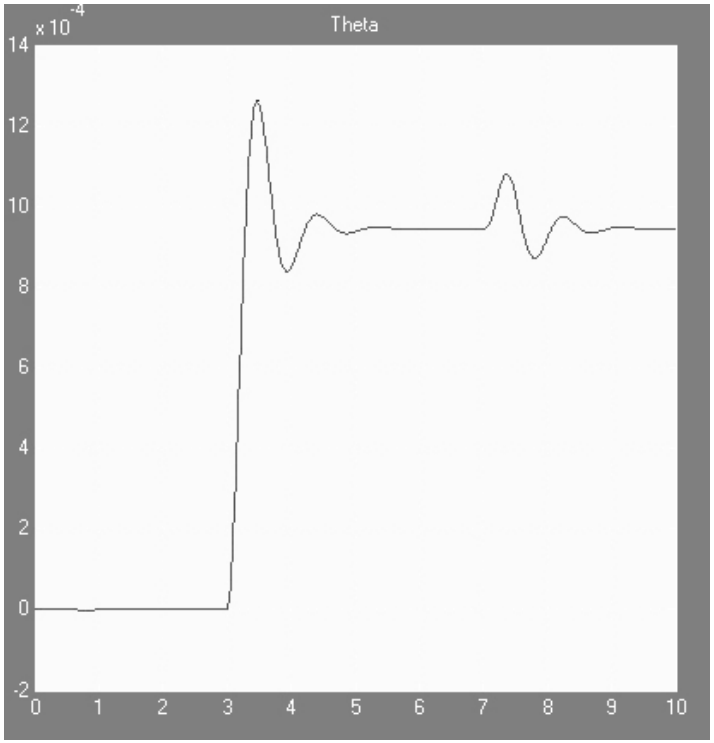
Figure: Simulink Simulation for the pitch

From Simulink to SpaceEx

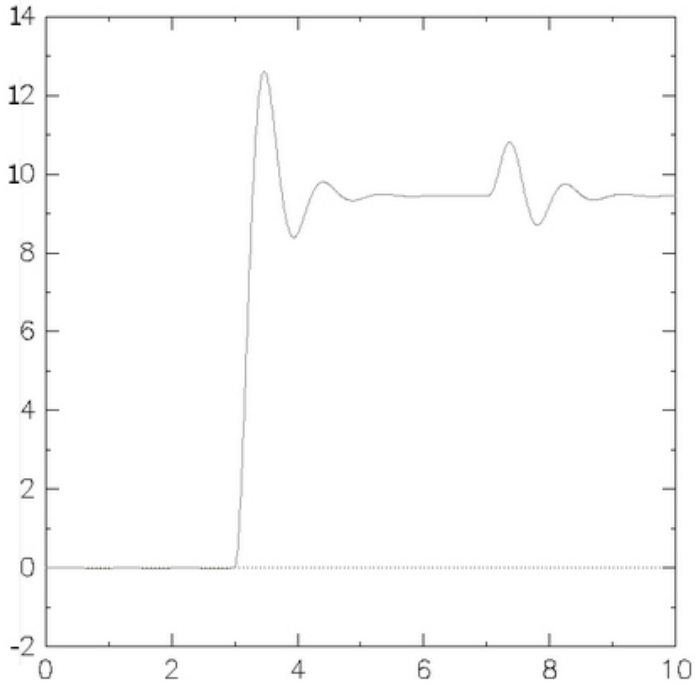
SX Model (Output from SL2SX Tool)



Simulation Comparison



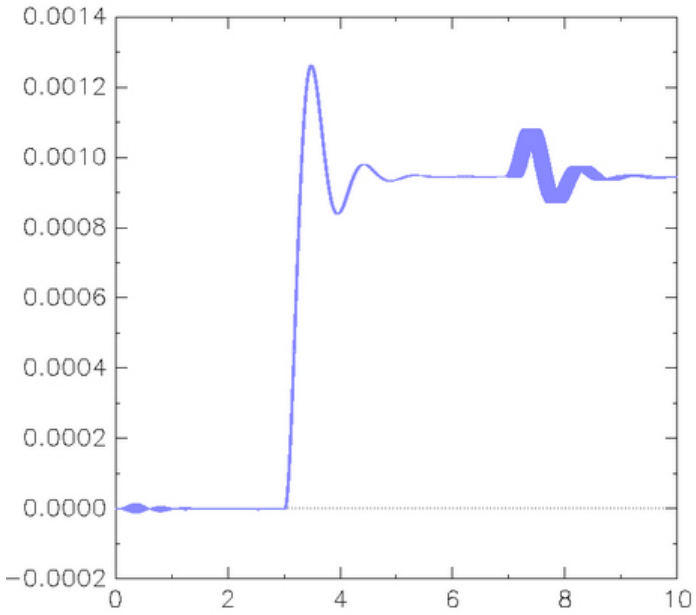
13.1 Simulink Simulation



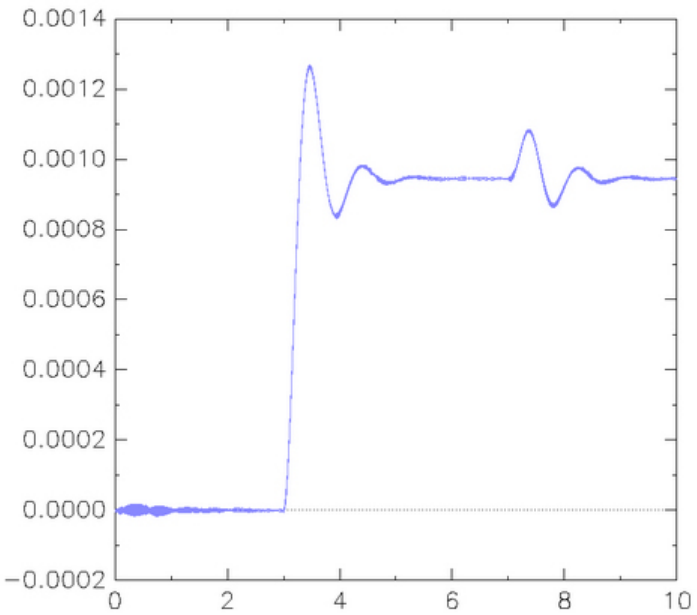
13.2 SpaceEx Simulation

Figure: Simulation for pitch

Performing Reachability



14.1 LGG Algorithm



14.2 STC Algorithm

Figure: Reachability for the pitch

- Perturbation on the Initial Condition:
  - $-0.121 \leq z \leq -0.119$

Conclusion

- SpaceEx Verification Platform
  - Verification Model and May Semantics
  - Features: optimized reachability algorithms, structure-oriented (components, hierarchy, ...), designed to facilitate implementation of new algorithms
- Simulation Tools
  - Simulation Models and Must Semantics
  - From Simulation to Verification Models (Urgent Conditions)
  - SL2SX Tool for automatic translation
- Reachability Analysis of a SL Diagram with SpaceEx

Future Work

- Implementation of Reachability Algorithms for Affine HA with urgency
- Extend SL2SX
- Translation from other Simulation Models (like Modelica, ...)

Thank You!

Merci Beaucoup!