Modeling and Analysis of Hybrid Systems

Introduction

Prof. Dr. Erika Ábrahám

Informatik 2 - Theory of Hybrid Systems
RWTH Aachen University

SS 2015

Organizational

Lecture:

- Monday 15:15-16:00 in AH 3
- Tuesday 12:15-13:45 in 5056

Exercise:

- Monday 16:00-16:45 in AH 3

Exam dates:

- 1st: 27.07.2015 14:15-16:45
- 2nd: 16.09.2015 15:45-18:15

Learning materials and contact persons

Learning materials available in L2P:

- Slides
- Lecture notes
- Some research publications
- Exercise sheets, solutions

Lecture:

Erika Ábrahám
room: 4229 (E1, 2nd floor), phone: 0241/80-21242
email: abraham@informatik.rwth-aachen.de

Exercise:

Stefan Schupp
room: 4228 (E1, 2nd floor), phone: 0241/80-21243
email: stefan.schupp@informatik.rwth-aachen.de

Further information (topic, evaluations etc.):

`http://ths.rwth-aachen.de/teaching/ss15/lecture-modelling-and-analysis-of-hybrid-systems/`

YABS Backchannelsystem

access: `lanzarote.informatik.rwth-aachen.de:8080`

username: `tim-username`

password: `tim-password`

Contents

Contents

“Hybrid”

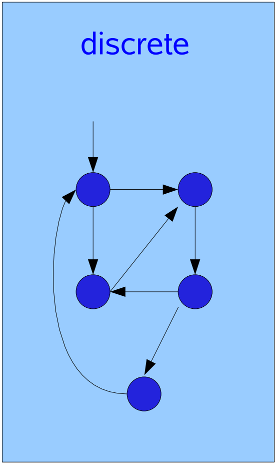
Wikipedia:

> “A hybrid is the combination of two or more different things, aimed at achieving a particular objective or goal.”

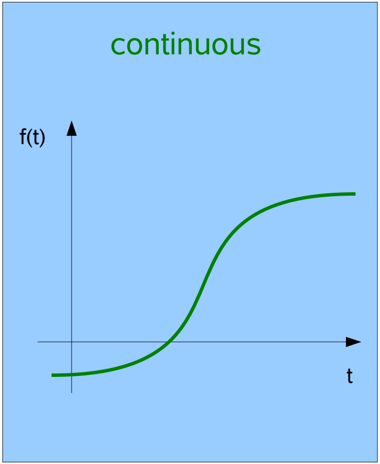
A hybrid rose

A hybrid car

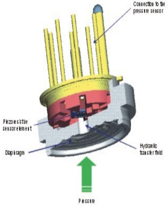
Hybrid in computer science

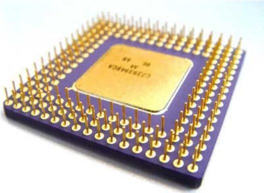


+



The discrete part





Combined with the continuous part

Example: Thermostat

Example: Thermostat

- Temperature x is controlled by switching a heater on and off
- x is regulated by a thermostat:
  - $17° \leq x \leq 18° \rightsquigarrow$ "heater on"
  - $22° \leq x \leq 23° \rightsquigarrow$ "heater off"

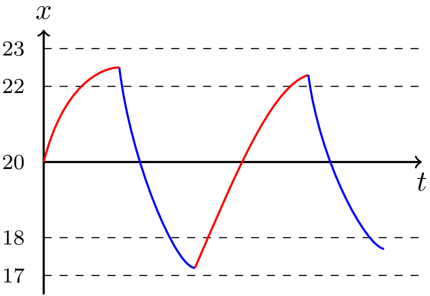
Example: Thermostat

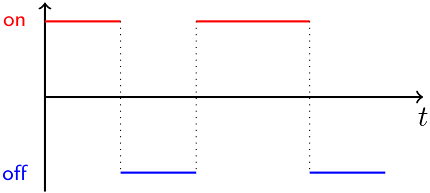
- Temperature x is controlled by switching a heater on and off
- x is regulated by a thermostat:
  - $17^\circ \leq x \leq 18^\circ \rightsquigarrow$ "heater on"
  - $22^\circ \leq x \leq 23^\circ \rightsquigarrow$ "heater off"

Continuous: temperature
Discrete: switching

- Temperature x is controlled by switching a heater on and off
- x is regulated by a thermostat:
  - $17° \leq x \leq 18° \leadsto$ "heater on"
  - $22° \leq x \leq 23° \leadsto$ "heater off"

Continuous: temperature
Discrete: switching





Contents

Modeling

Modeling

- To be able to apply formal (mathematical) methods to a real system, we need a formal model of it.
- A model never exactly corresponds to the modeled real system.
- Abstract away unnecessary details.

Modeling

- To be able to apply formal (mathematical) methods to a real system, we need a formal model of it.
- A model never exactly corresponds to the modeled real system.
- Abstract away unnecessary details.
- What you probably already know: Kripke structures (state transition systems)

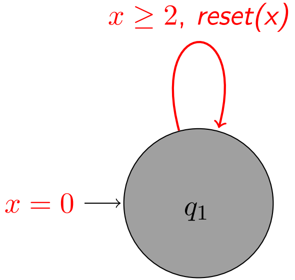
Modeling

- To be able to apply formal (mathematical) methods to a real system, we need a formal model of it.
- A model never exactly corresponds to the modeled real system.
- Abstract away unnecessary details.
- What you probably already know: Kripke structures (state transition systems)
- What you probably also know: Transition systems

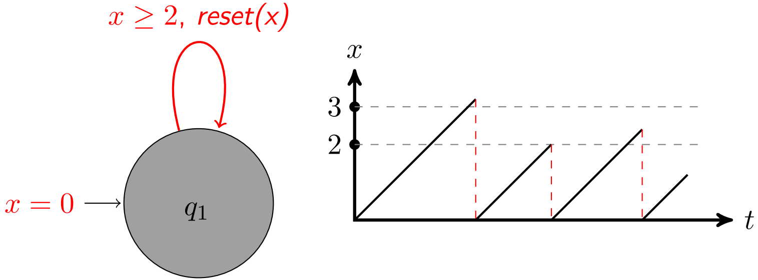
Modeling

- To be able to apply formal (mathematical) methods to a real system, we need a formal model of it.
- A model never exactly corresponds to the modeled real system.
- Abstract away unnecessary details.
- What you probably already know: Kripke structures (state transition systems)
- What you probably also know: Transition systems
- What you perhaps know: Timed automata

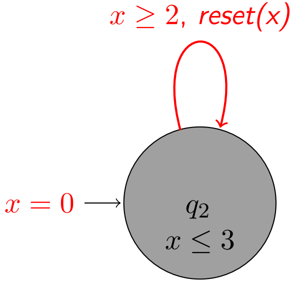
Example: Timed automaton

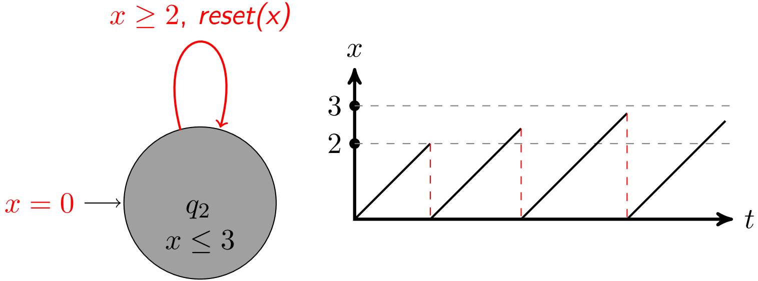


Example: Timed automaton



Example: Timed automaton




$x \geq 2$, reset(x)
$x = 0$
q_2
$x \leq 3$
x
3
2
t

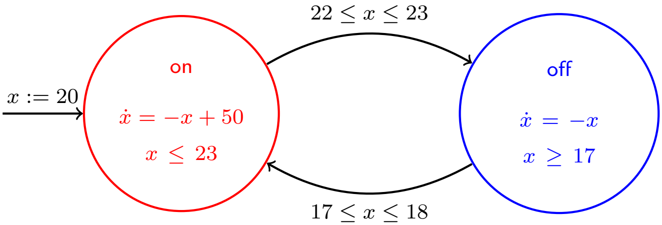
Modeling general hybrid systems: Hybrid automata

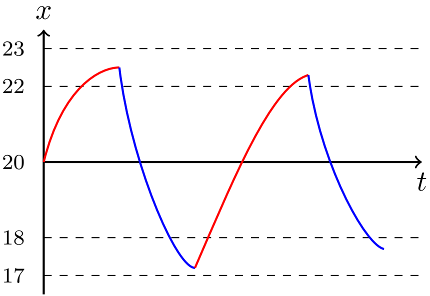
Modeling general hybrid systems: Hybrid automata

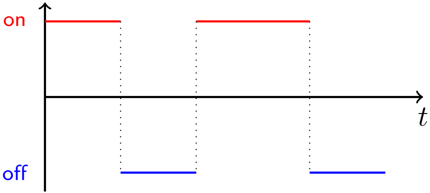
Let's take again the thermostat as an example.

Modeling general hybrid systems: Hybrid automata

Let's take again the thermostat as an example.







Contents

Logic

Logic

- We want to specify how a hybrid system is expected to behave.

Logic

- We want to specify how a hybrid system is expected to behave.
- We are especially interested in safety and liveness.

Logic

- We want to specify how a hybrid system is expected to behave.
- We are especially interested in safety and liveness.
- E.g., we can use the logic TCTL for specification.

Logic

- We want to specify how a hybrid system is expected to behave.
- We are especially interested in safety and liveness.
- E.g., we can use the logic TCTL for specification.
- In TCTL we can express properties like:

  "The temperature is always below $20°C$."

Logic

- We want to specify how a hybrid system is expected to behave.
- We are especially interested in safety and liveness.
- E.g., we can use the logic TCTL for specification.
- In TCTL we can express properties like:

  "The temperature is always below $20°C$."

- Or

  "If the temperature is above $20°C$ it will get below $20°C$ within 5 seconds."

Logic

- We want to specify how a hybrid system is expected to behave.
- We are especially interested in safety and liveness.
- E.g., we can use the logic TCTL for specification.
- In TCTL we can express properties like:

  "The temperature is always below $20°C$."

- Or

  "If the temperature is above $20°C$ it will get below $20°C$ within 5 seconds."

- Or

  "It is always the case that the temperature will somewhen in the future get above $20°C$."

Contents

The analysis of hybrid systems

The analysis of hybrid systems

- Assume we modeled a hybrid system as a hybrid automaton.
- Assume we specified a property of the system.
- Can we prove that the system satisfies the property?

The analysis of hybrid systems

- Assume we modeled a hybrid system as a hybrid automaton.
- Assume we specified a property of the system.
- Can we prove that the system satisfies the property?
- Well, it depends...

The analysis of hybrid systems

- Assume we modeled a hybrid system as a hybrid automaton.
- Assume we specified a property of the system.
- Can we prove that the system satisfies the property?
- Well, it depends...
- ...on the fact if the logic is decidable for the underlying modeling language.

The analysis of hybrid systems

- Assume we modeled a hybrid system as a hybrid automaton.
- Assume we specified a property of the system.
- Can we prove that the system satisfies the property?
- Well, it depends...
- ...on the fact if the logic is decidable for the underlying modeling language.
- We will see for which classes of hybrid automata the reachability question is decidable.

The analysis of hybrid systems

- Assume we modeled a hybrid system as a hybrid automaton.
- Assume we specified a property of the system.
- Can we prove that the system satisfies the property?
- Well, it depends...
- ...on the fact if the logic is decidable for the underlying modeling language.
- We will see for which classes of hybrid automata the reachability question is decidable.
- We will deal with
  - (unbounded) reachability for timed automata.
  - (unbounded) reachability for initialized rectangular automata.
  - bounded reachability for linear hybrid automata.
  - reachability approximation for general hybrid automata.

Method for timed automata: Finite abstraction

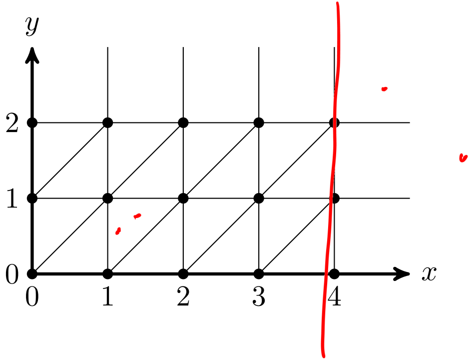
Method for timed automata: Finite abstraction

Constructive proof of decidability via finite abstraction:

Method for timed automata: Finite abstraction

Constructive proof of decidability via finite abstraction:

Constructive proof of decidability via finite abstraction:



Method for initialized rectangular automata: Transformation

Method for initialized rectangular automata: Transformation

Leading back the proof of decidability to a known problem:

Method for initialized rectangular automata: Transformation

Leading back the proof of decidability to a known problem:

Timed automaton

$\uparrow$

Initialized stopwatch automaton

$\uparrow$

Initialized singular automaton

$\uparrow$

Initialized rectangular automaton

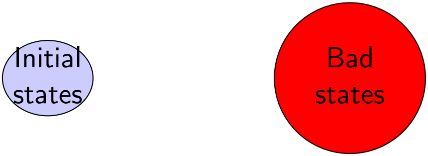
Method for linear hybrid automata: Fixedpoint computation

Method for linear hybrid automata: Fixedpoint computation

Forward reachability computation:

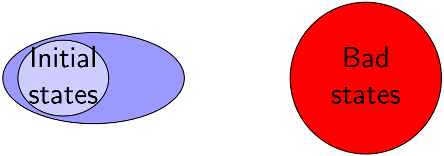
Method for linear hybrid automata: Fixedpoint computation

Forward reachability computation:



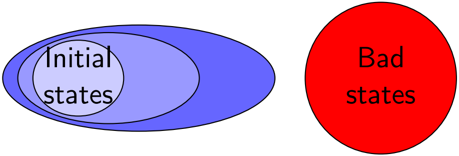
Method for linear hybrid automata: Fixedpoint computation

Forward reachability computation:

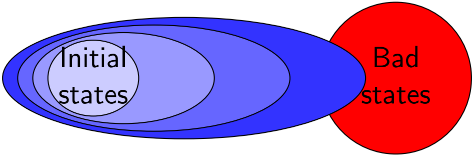


Method for linear hybrid automata: Fixedpoint computation

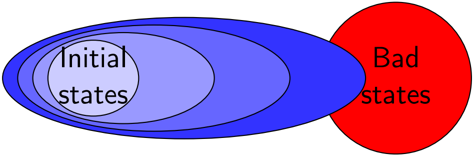
Forward reachability computation:



Forward reachability computation:



Forward reachability computation:



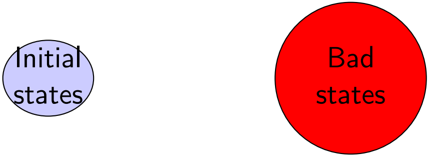
Note: the method is incomplete

Method for linear hybrid automata: Fixedpoint computation

Backward reachability computation:

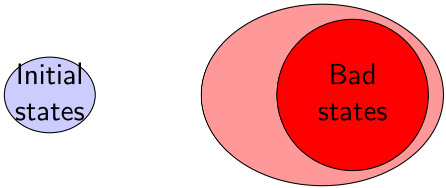
Method for linear hybrid automata: Fixedpoint computation

Backward reachability computation:



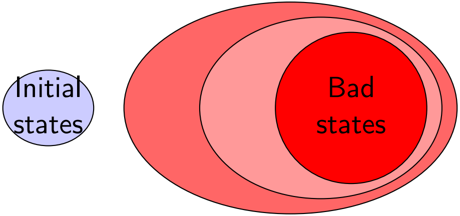
Method for linear hybrid automata: Fixedpoint computation

Backward reachability computation:



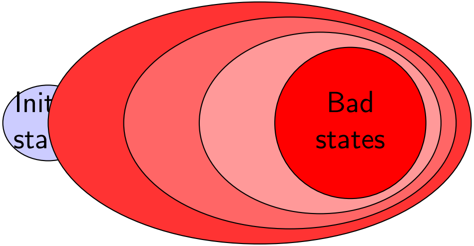
Method for linear hybrid automata: Fixedpoint computation

Backward reachability computation:



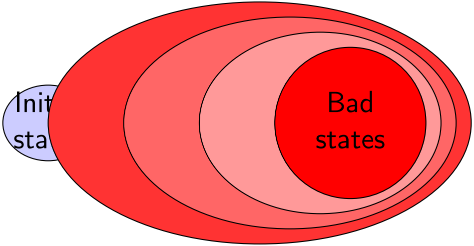
Method for linear hybrid automata: Fixedpoint computation

Backward reachability computation:



Method for linear hybrid automata: Fixedpoint computation

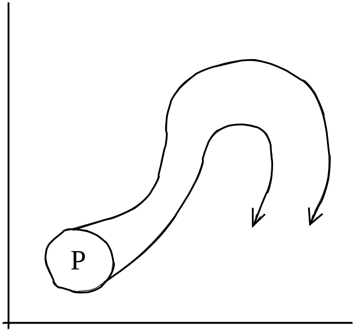
Backward reachability computation:

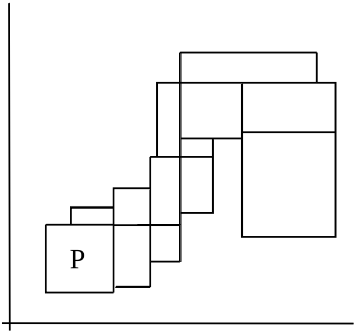


Note: also the backward method is incomplete

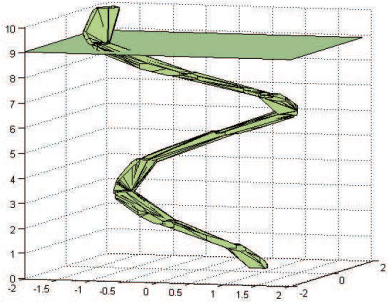
Method for hybrid automata: Approximation

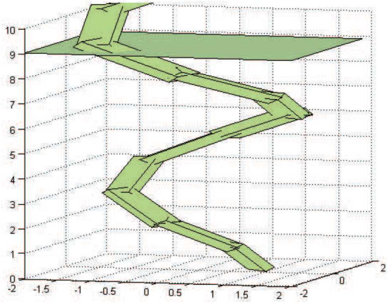
Method for hybrid automata: Approximation



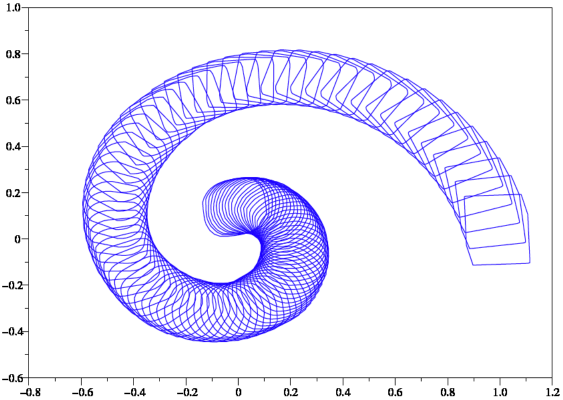


Polyhedra (left) and oriented rectangular hulls (right) in reachability computation

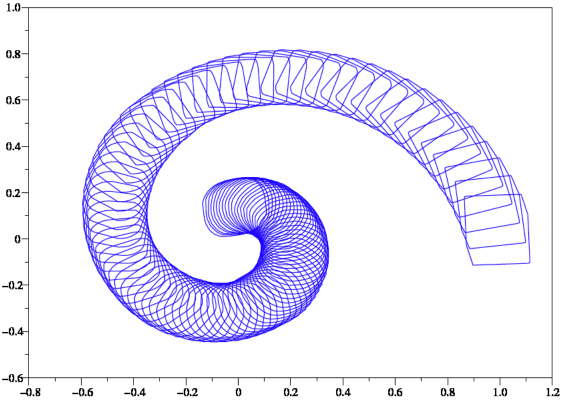


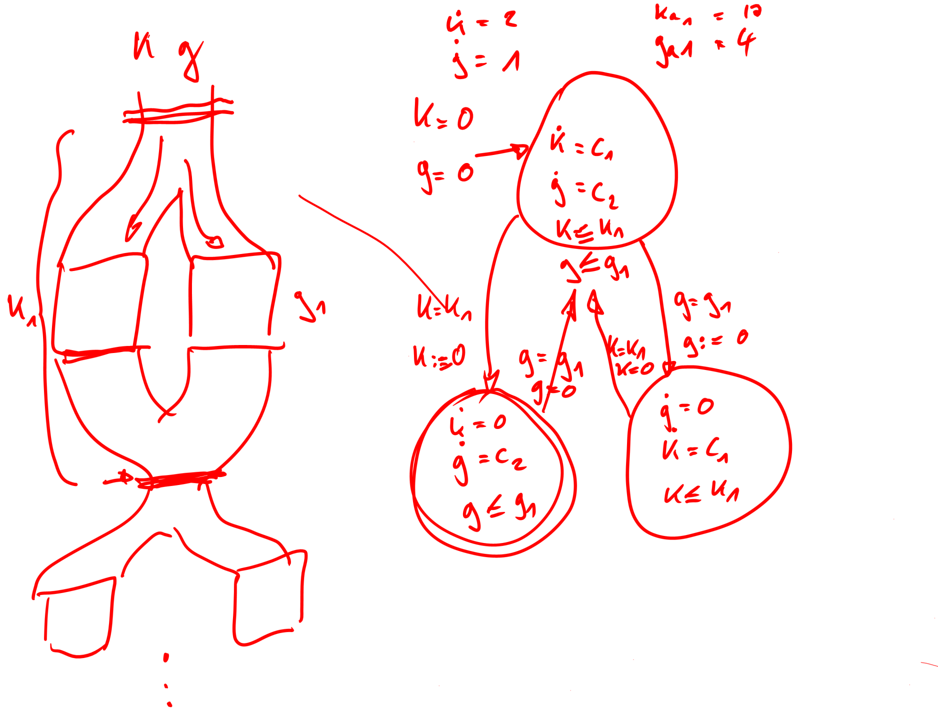


Zonotopes in reachability computation



Zonotopes in reachability computation





k g
k1
g1
i = 2
j = 1
ka1 = 12
ga1 = 4
k = 0
g = 0
k = c1
g = c2
k ≤ k1
g ≤ g1
k = k1
k := 0
g = g1
g := 0
g = g1
g := 0
k = k1
k := 0
g = g1
g := 0
i = 0
g = c2
g ≤ g1
g = 0
k = c1
k ≤ k1