Robust verification of timed systems

Nicolas Markey
LSV, CNRS & ENS Cachan, France

Journée Industrielle
Saclay, 8 February 2013

Verification of computerised systems

- Computers are everywhere

Verification of computerised systems

- Computers are everywhere

- Bugs are everywhere...

News

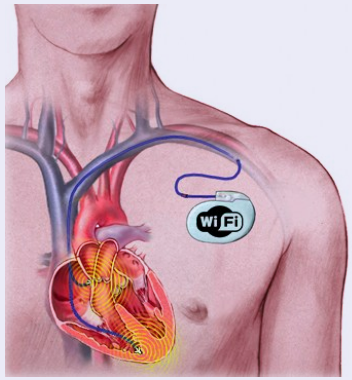
Toyota to recall Prius hybrids over ABS software

See video, below

By Martyn Williams

February 9, 2010 04:59 AM ET Comments (6) Recommended (15) Like

IDG News Service - Toyota plans to recall around 400,000 of its Prius hybrid cars to replace software that controls the antilock braking system (ABS), the auto maker said Tuesday.



Verification of computerised systems

- Computers are everywhere

- Bugs are everywhere...

News

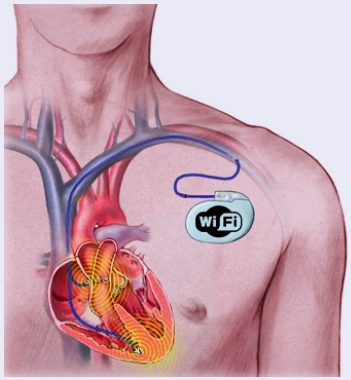
Toyota to recall Prius hybrids over ABS software

See video, below

By Martyn Williams

February 9, 2010 04:59 AM ET Comments (6) Recommended (15) Like

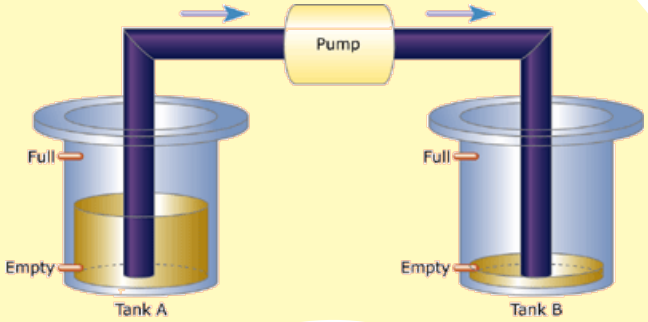
IDG News Service - Toyota plans to recall around 400,000 of its Prius hybrid cars to replace software that controls the antilock braking system (ABS), the auto maker said Tuesday.



- Verification should be everywhere!

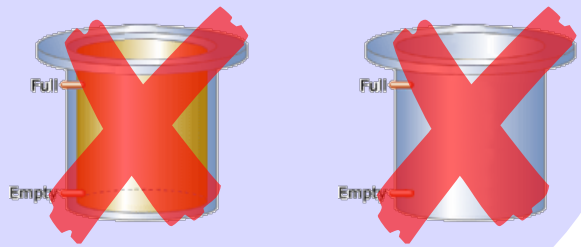
Model checking and synthesis

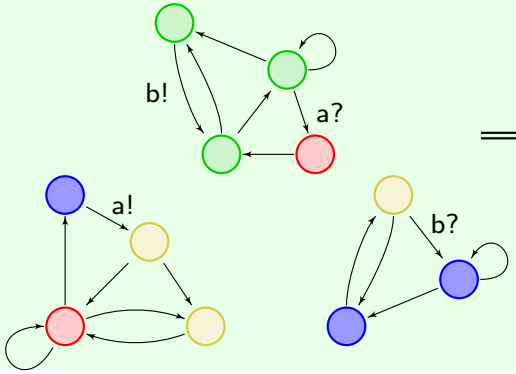
system:



[http://www.embedded.com]

property:



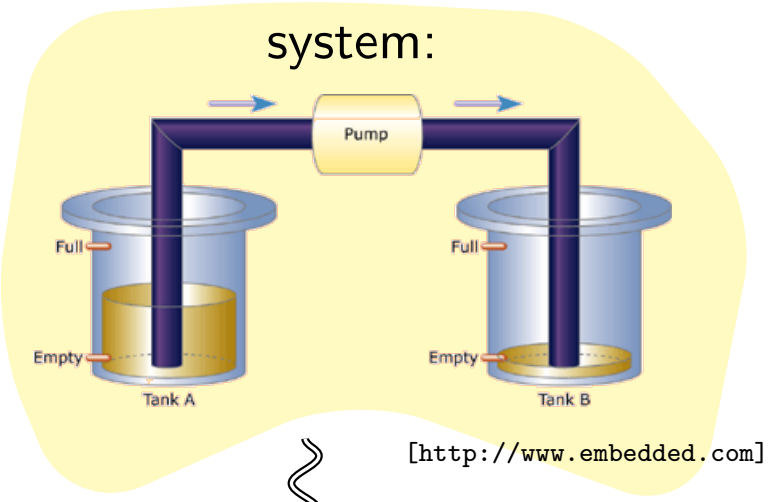


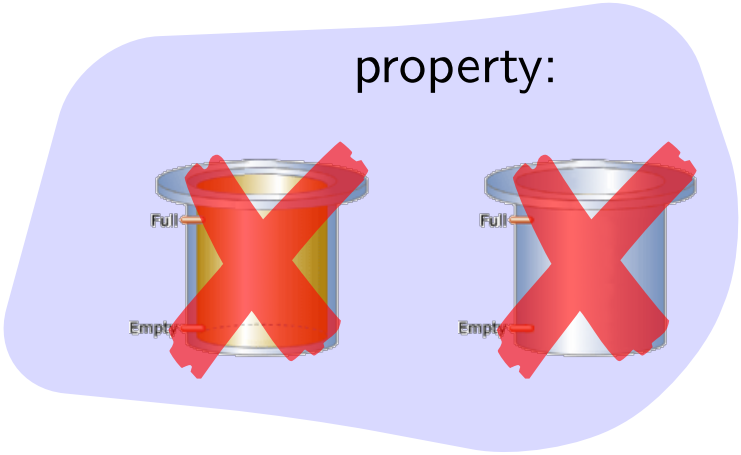
model-checking algorithm

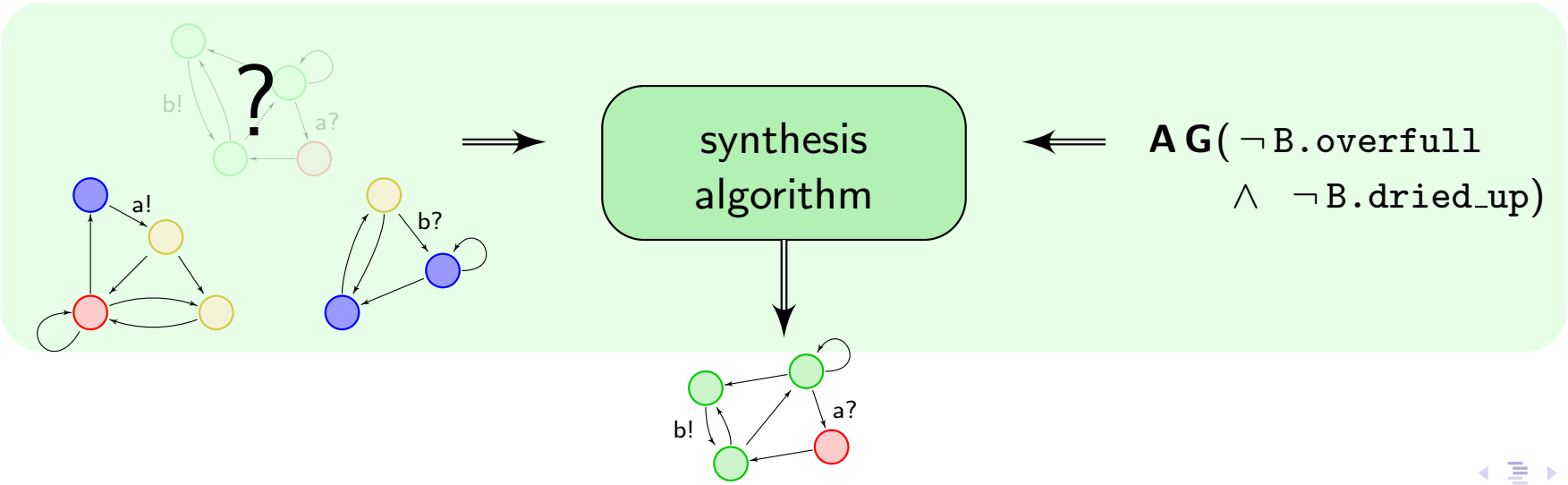
$\mathbf{A}\,\mathbf{G}(\neg\,\texttt{B.overfull} \wedge \neg\,\texttt{B.dried_up})$

yes/no

Model checking and synthesis

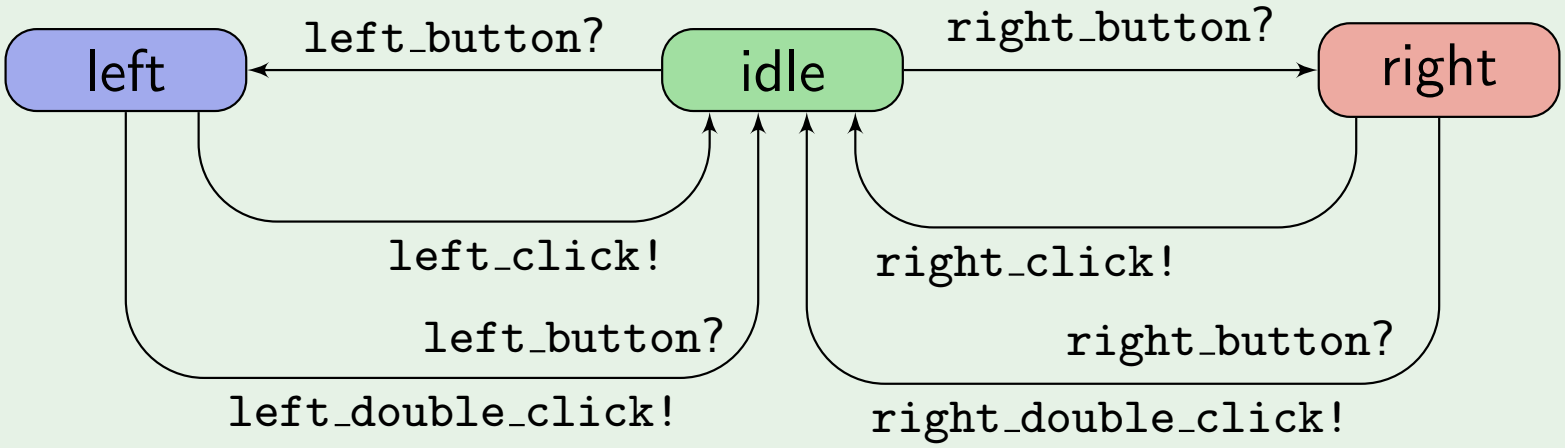






Reasoning about real-time systems

Example (A computer mouse)



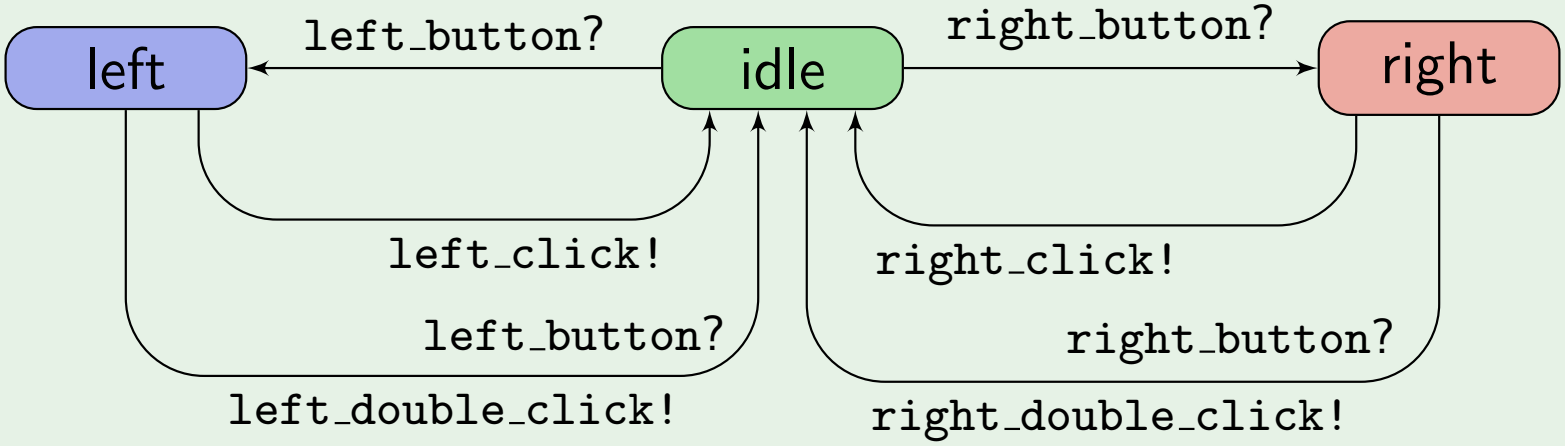
Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,

Example (A computer mouse)



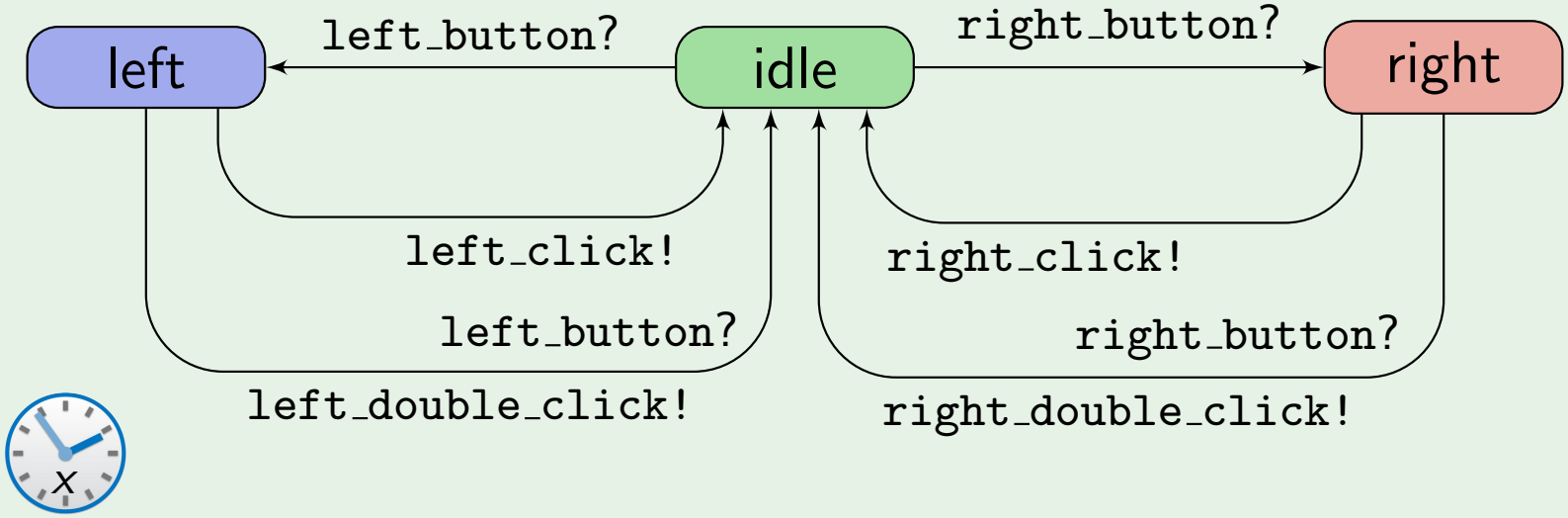
Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,
- a set of clocks,

Example (A computer mouse)



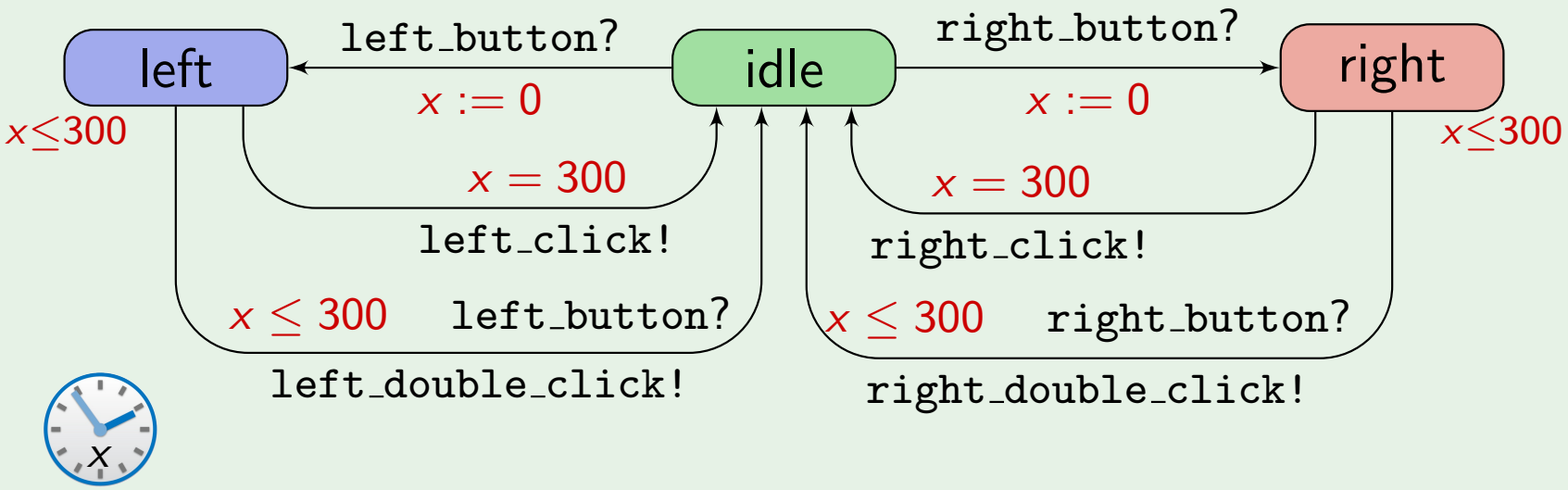
Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,
- a set of clocks,
- timing constraints on states and transitions.

Example (A computer mouse)



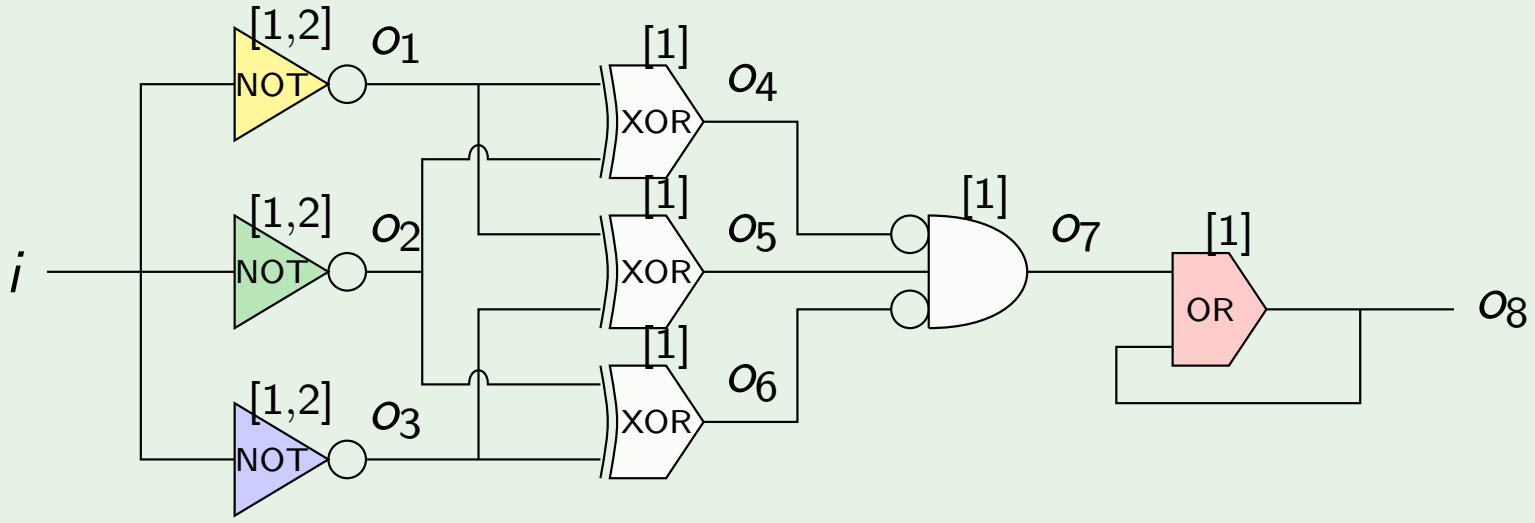
Discrete-time semantics

...because computers are digital!

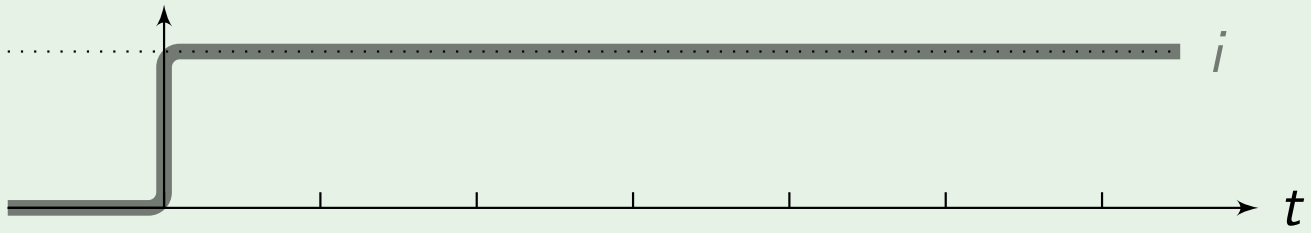
Discrete-time semantics

...because computers are digital!

Example



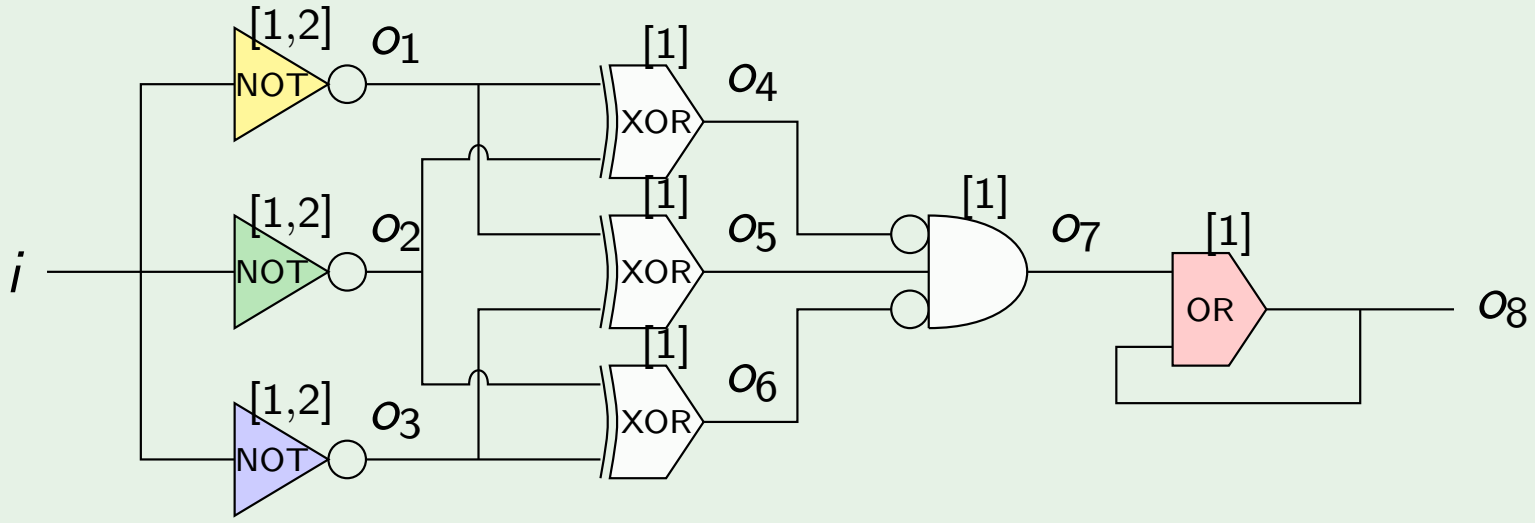
- under discrete-time, the output is always 0:



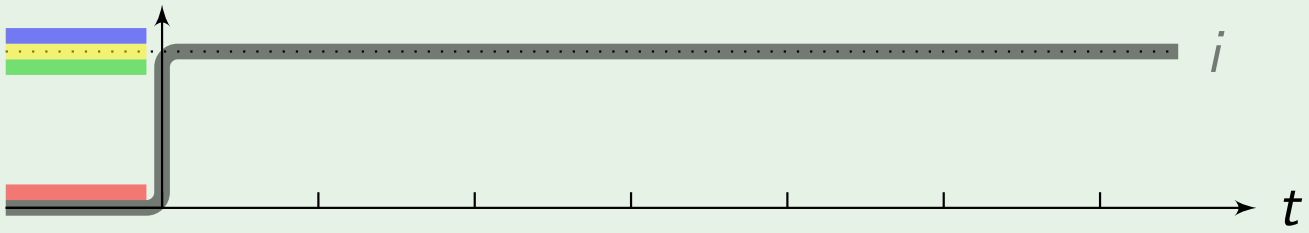
Discrete-time semantics

...because computers are digital!

Example



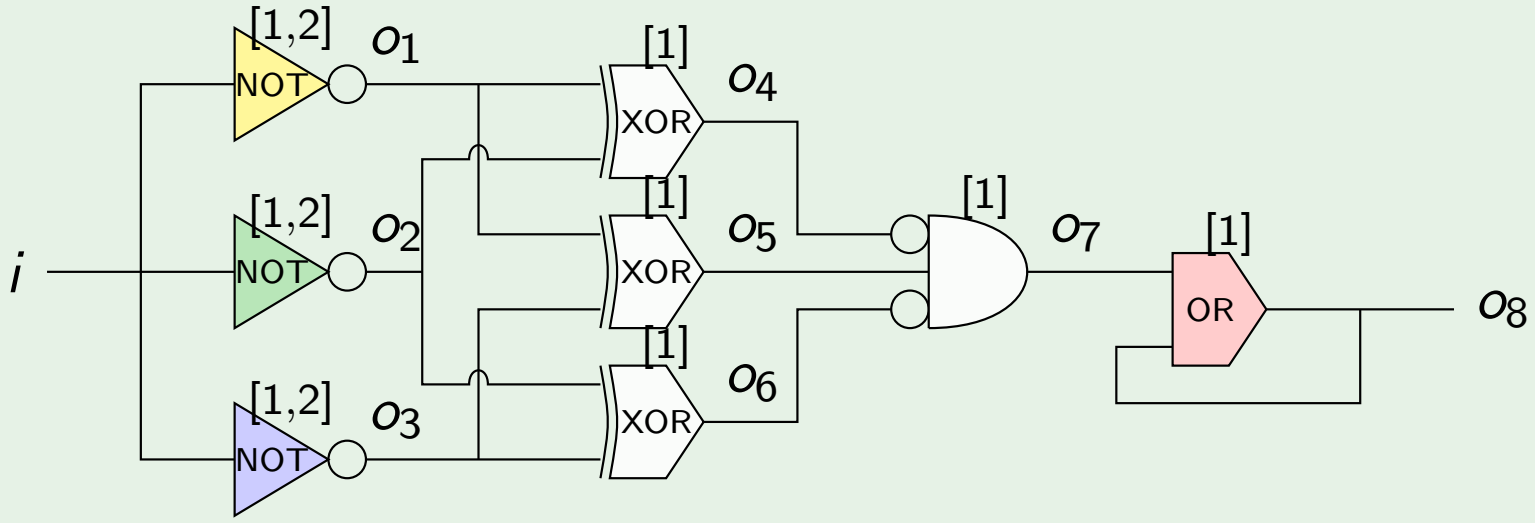
- under discrete-time, the output is always 0:



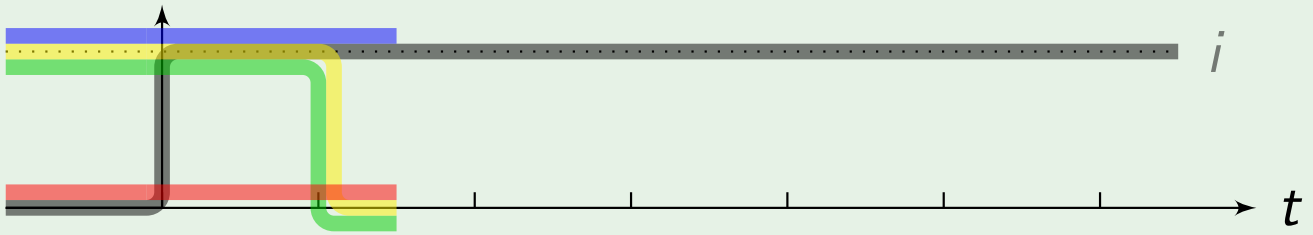
Discrete-time semantics

...because computers are digital!

Example



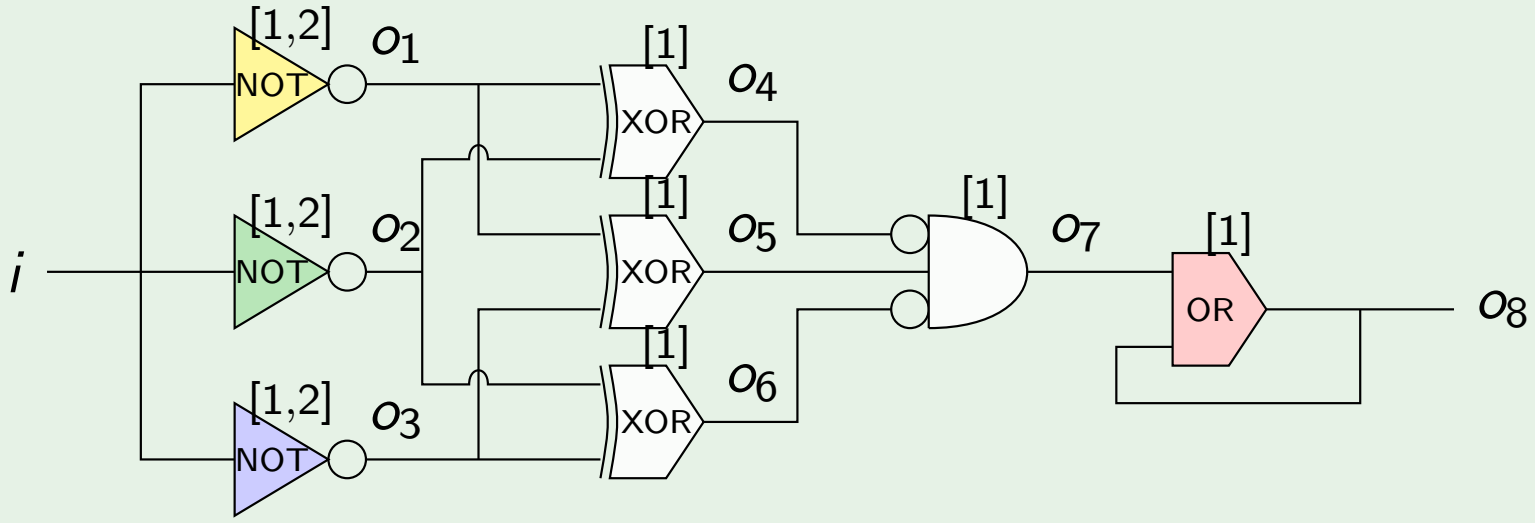
- under discrete-time, the output is always 0:



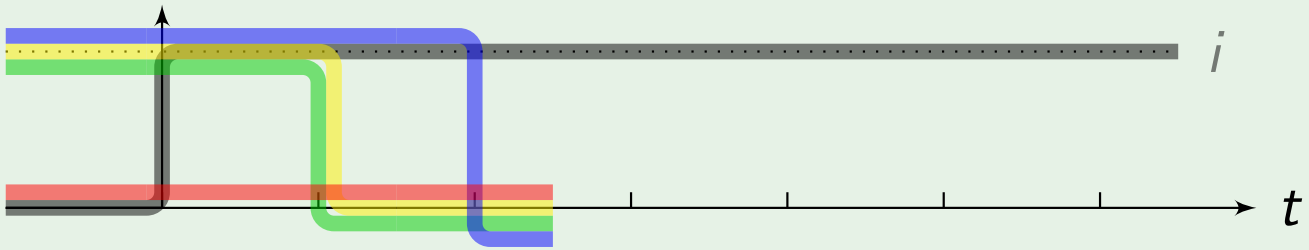
Discrete-time semantics

...because computers are digital!

Example



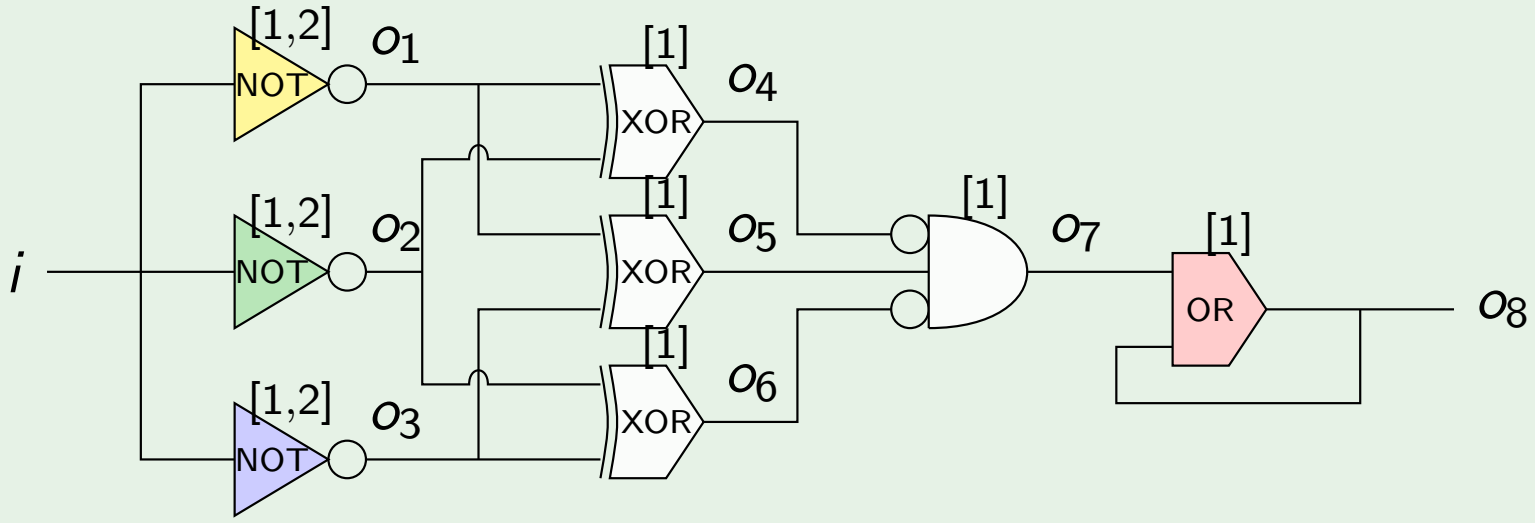
- under discrete-time, the output is always 0:



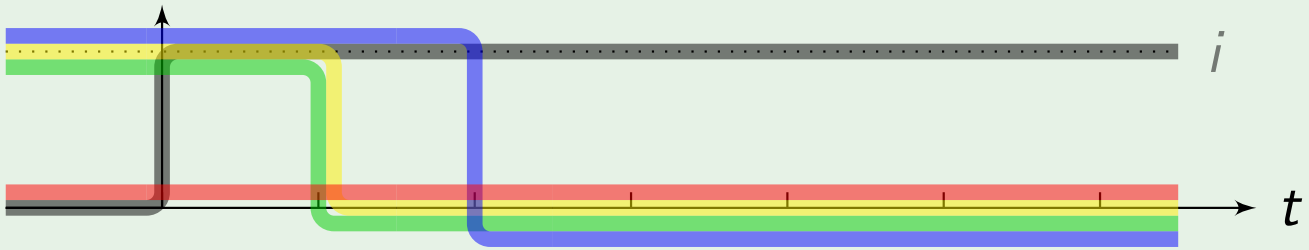
Discrete-time semantics

...because computers are digital!

Example



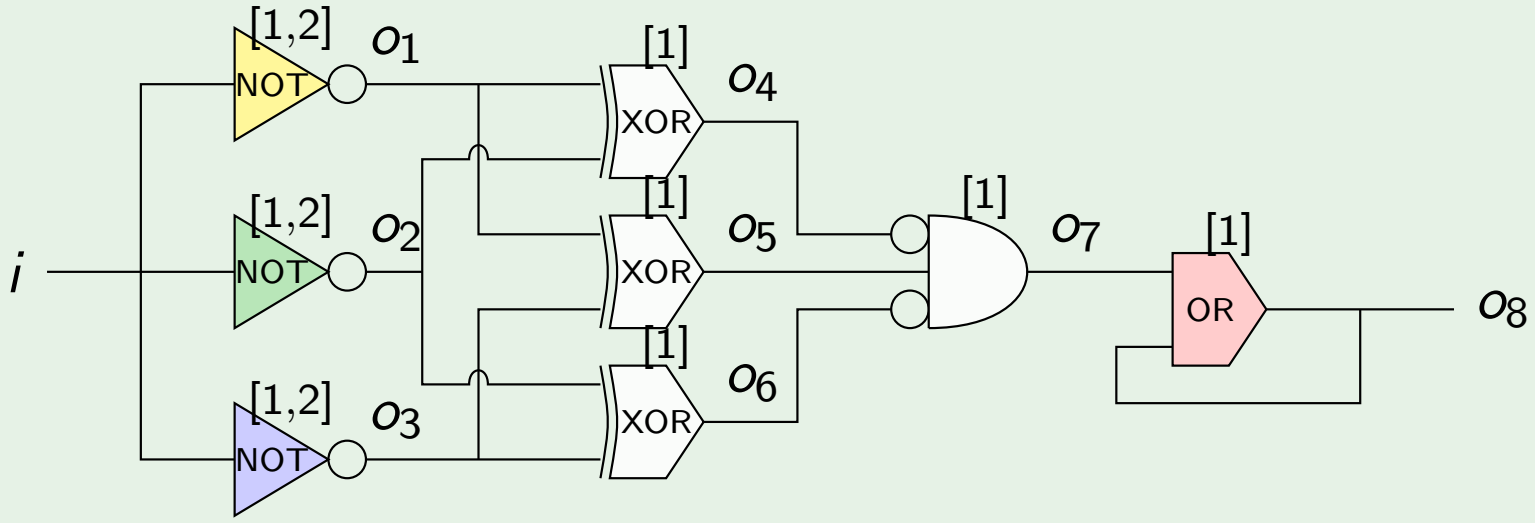
- under discrete-time, the output is always 0:



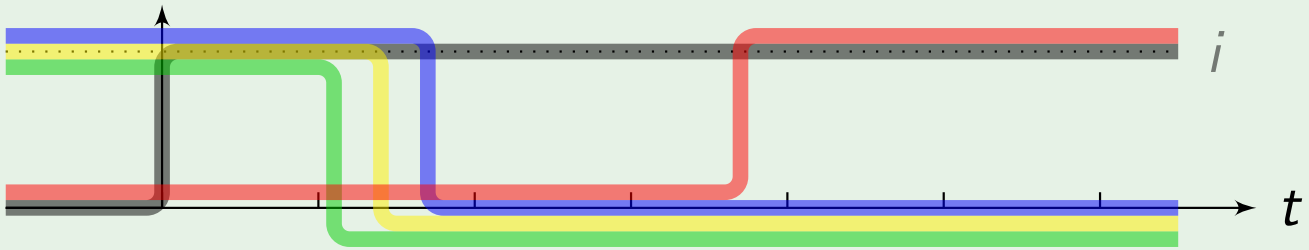
Discrete-time semantics

...because computers are digital!

Example



- under continuous-time, the output can be 1:



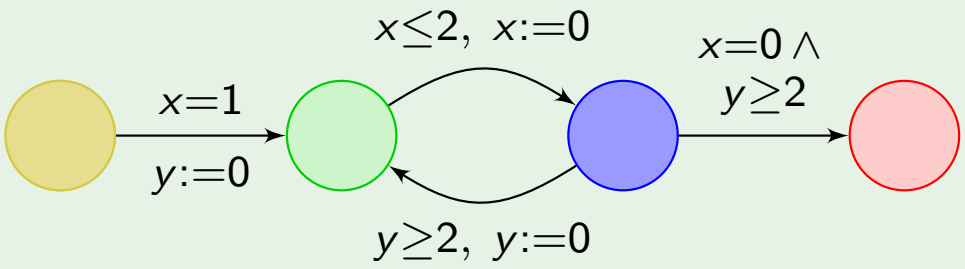
Continuous-time semantics

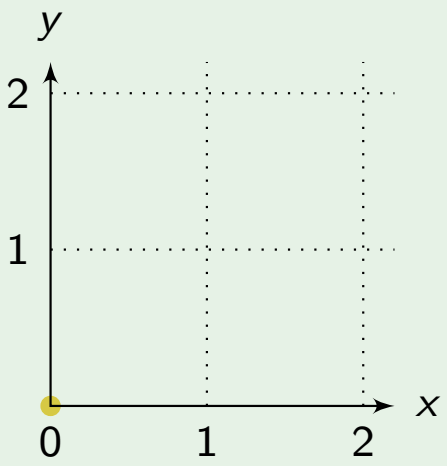
...real-time models for real-time systems!

Continuous-time semantics

...real-time models for real-time systems!

Example

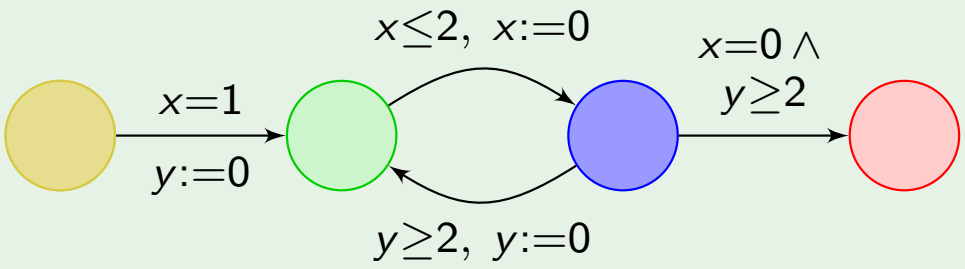


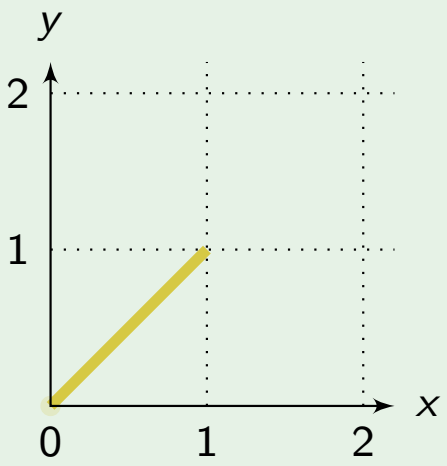


Continuous-time semantics

...real-time models for real-time systems!

Example

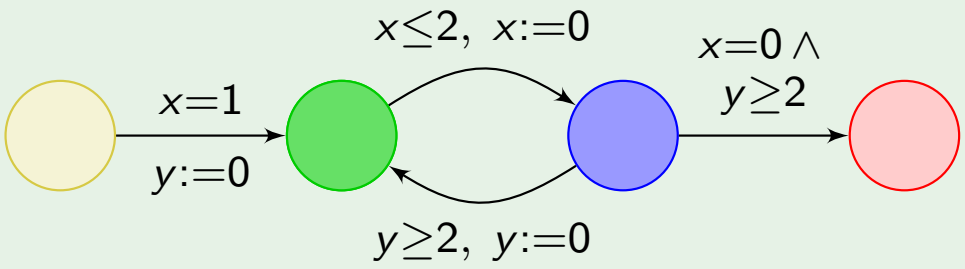


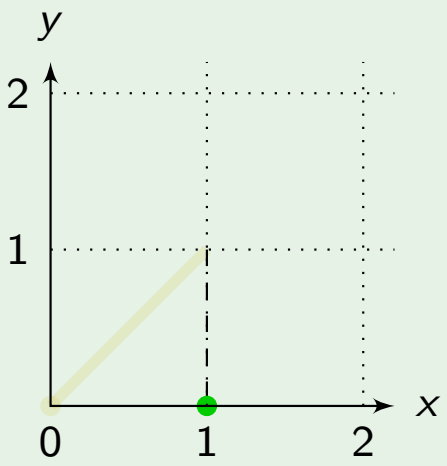


Continuous-time semantics

...real-time models for real-time systems!

Example

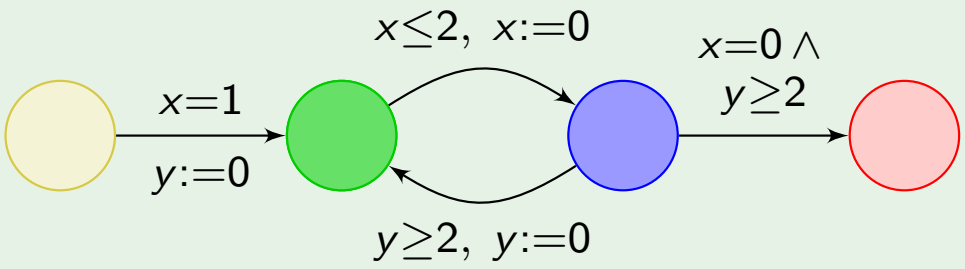


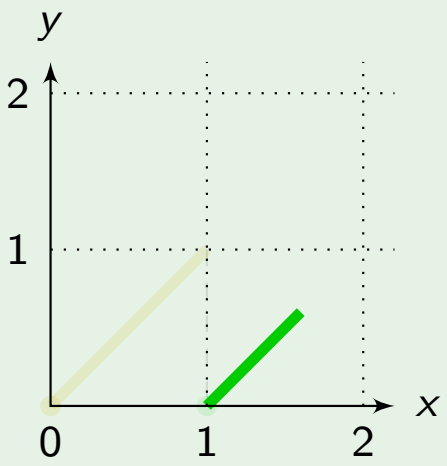


Continuous-time semantics

...real-time models for real-time systems!

Example

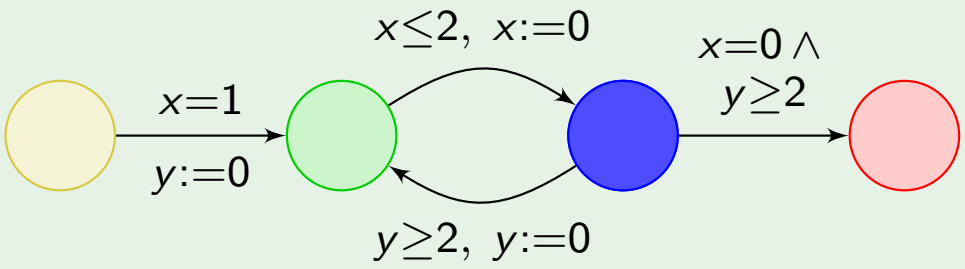


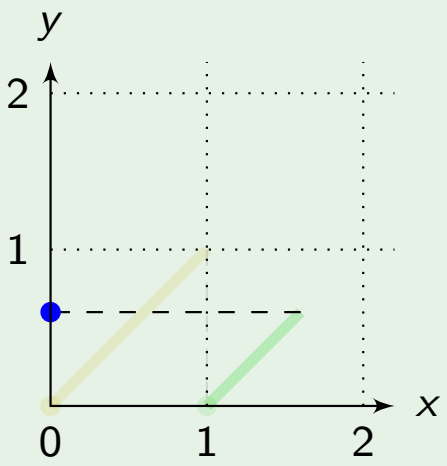


Continuous-time semantics

...real-time models for real-time systems!

Example

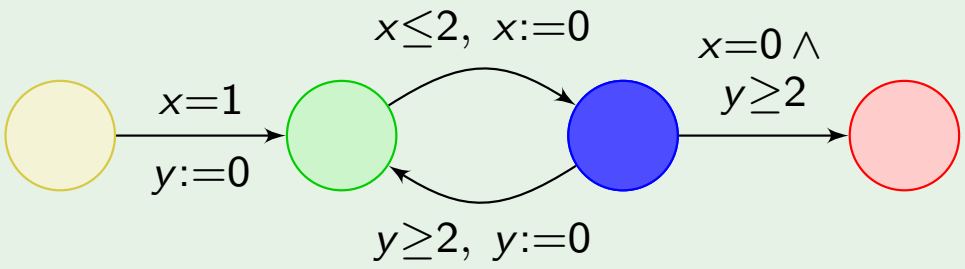


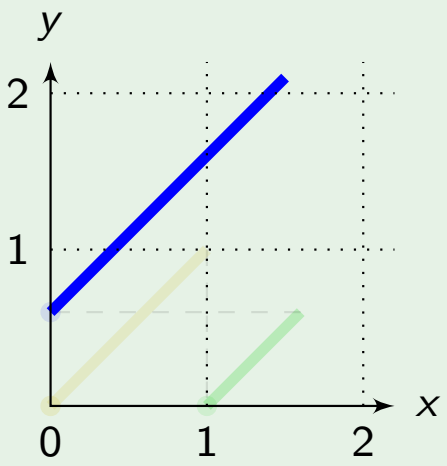


Continuous-time semantics

...real-time models for real-time systems!

Example

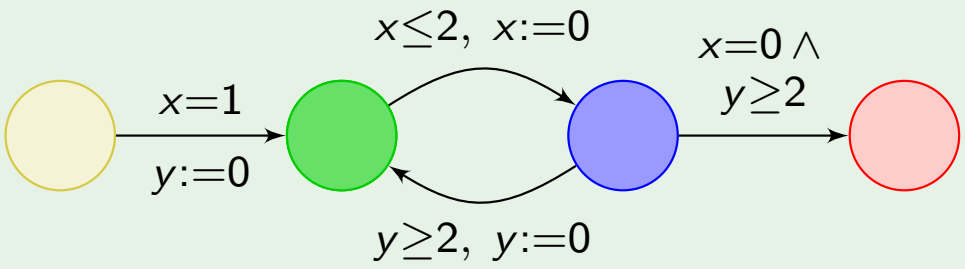


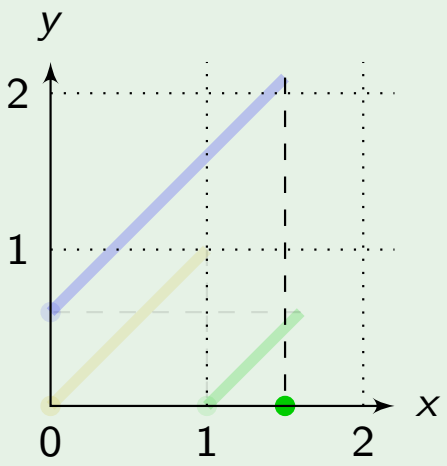


Continuous-time semantics

...real-time models for real-time systems!

Example

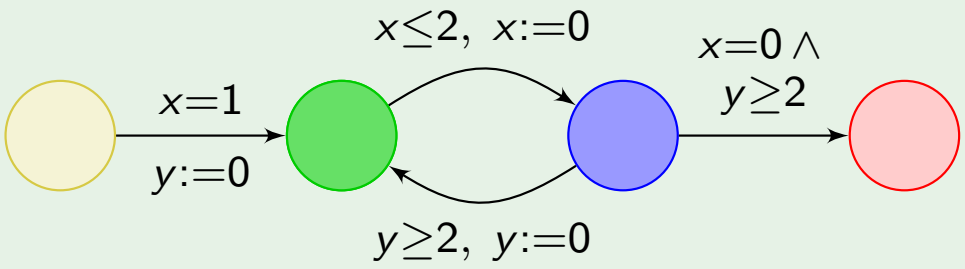


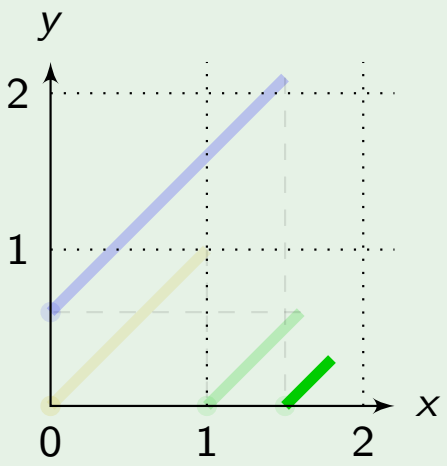


Continuous-time semantics

...real-time models for real-time systems!

Example

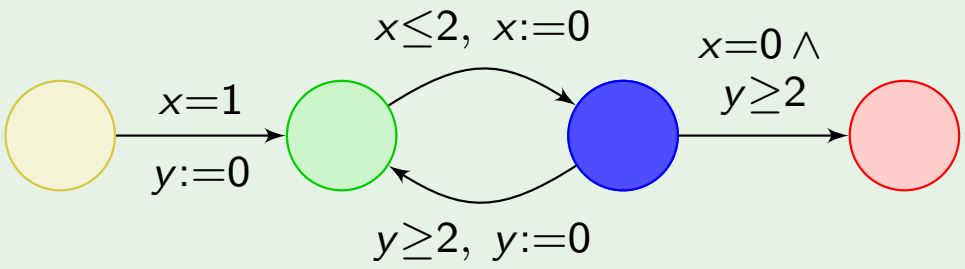


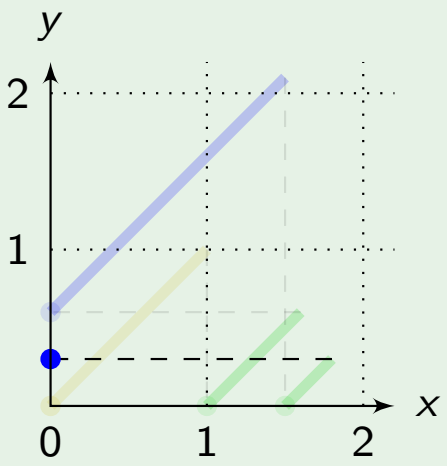


Continuous-time semantics

...real-time models for real-time systems!

Example

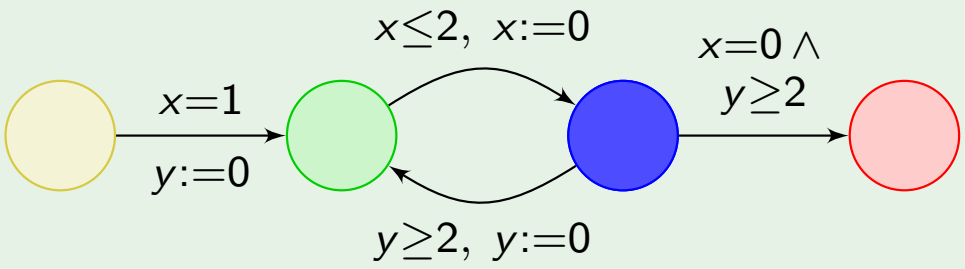


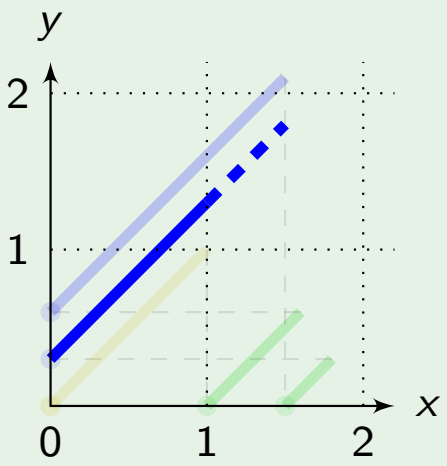


Continuous-time semantics

...real-time models for real-time systems!

Example

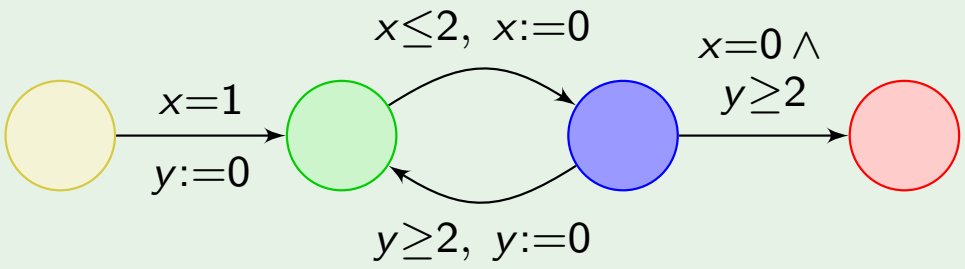


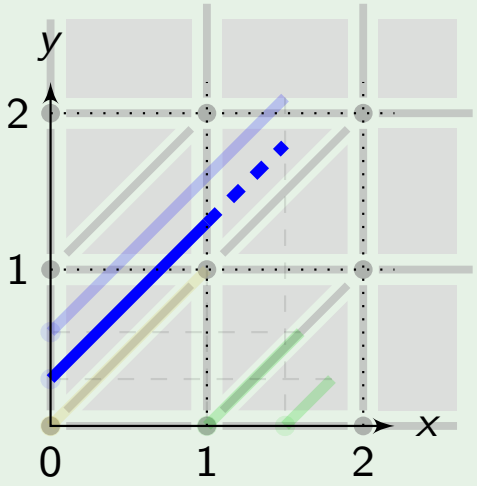


Continuous-time semantics

...real-time models for real-time systems!

Example

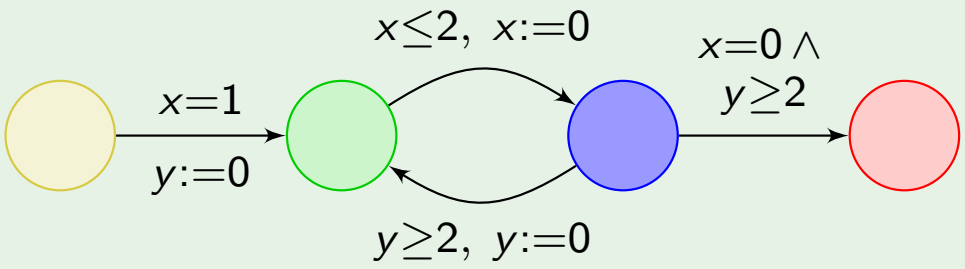


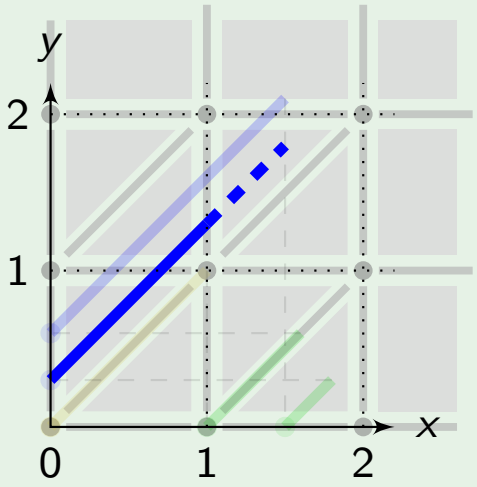


Continuous-time semantics

...real-time models for real-time systems!

Example





Theorem

Reachability in timed automata is decidable (as well as many other important properties).

Continuous-time semantics

the continuous-time semantics is a mathematical idealisation

- it assumes zero-delay transitions;
- it assumes infinite precision of the clocks;
- it assumes immediate communication between systems.

Continuous-time semantics

the continuous-time semantics is a mathematical idealisation

- it assumes zero-delay transitions;
- it assumes infinite precision of the clocks;
- it assumes immediate communication between systems.

Parametrised semantics

- parametrised discrete-time semantics:

  Does there exists a time step δ under which the system behaves correctly?

Continuous-time semantics

the continuous-time semantics is a mathematical idealisation

- it assumes zero-delay transitions;
- it assumes infinite precision of the clocks;
- it assumes immediate communication between systems.

Parametrised semantics

- parametrised discrete-time semantics:

  Does there exists a time step δ under which the system behaves correctly? $\rightsquigarrow$ undecidable

Continuous-time semantics

the continuous-time semantics is a mathematical idealisation

- it assumes zero-delay transitions;
- it assumes infinite precision of the clocks;
- it assumes immediate communication between systems.

Parametrised semantics

- parametrised discrete-time semantics:

  Does there exists a time step δ under which the system behaves correctly? $\rightsquigarrow$ undecidable

- parametrised continuous-time semantics:

  Does the system behave correctly under continuous-time semantics with imprecision up to some $\delta > 0$?

Continuous-time semantics

the continuous-time semantics is a mathematical idealisation

- it assumes zero-delay transitions;
- it assumes infinite precision of the clocks;
- it assumes immediate communication between systems.

Parametrised semantics

- parametrised discrete-time semantics:

  Does there exists a time step δ under which the system behaves correctly? $\rightsquigarrow$ undecidable

- parametrised continuous-time semantics:

  Does the system behave correctly under continuous-time semantics with imprecision up to some $\delta > 0$? $\rightsquigarrow$ decidable

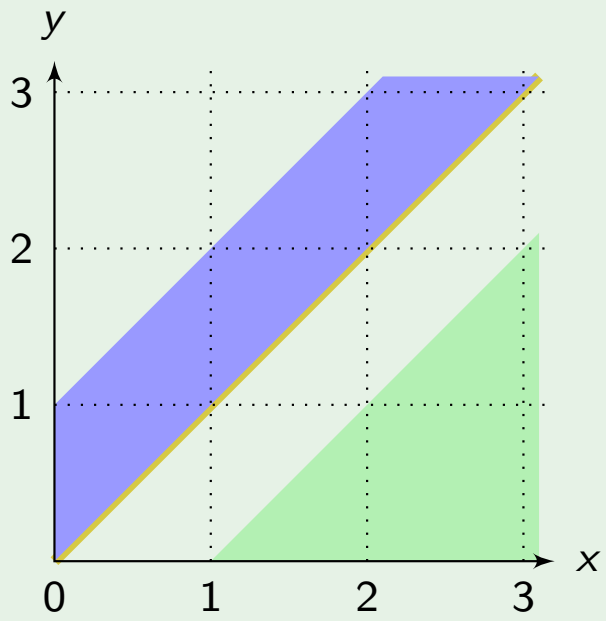
Enlarged semantics for timed automata

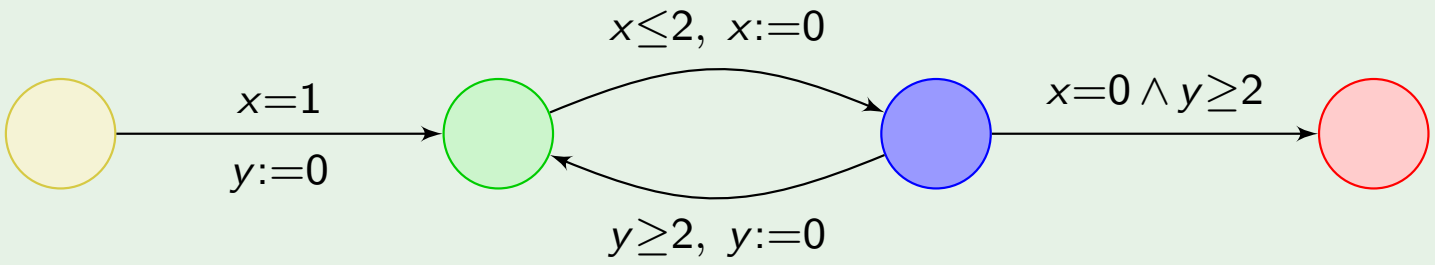
a transition can be taken at any time in $[t - \delta; t + \delta]$.

Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example

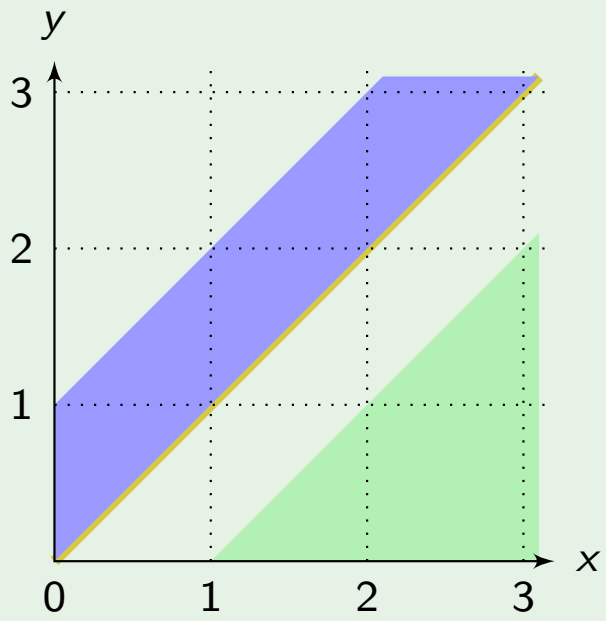


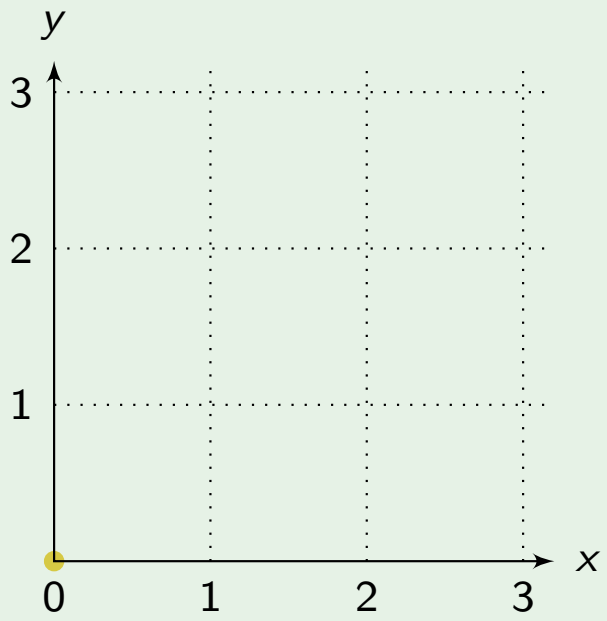


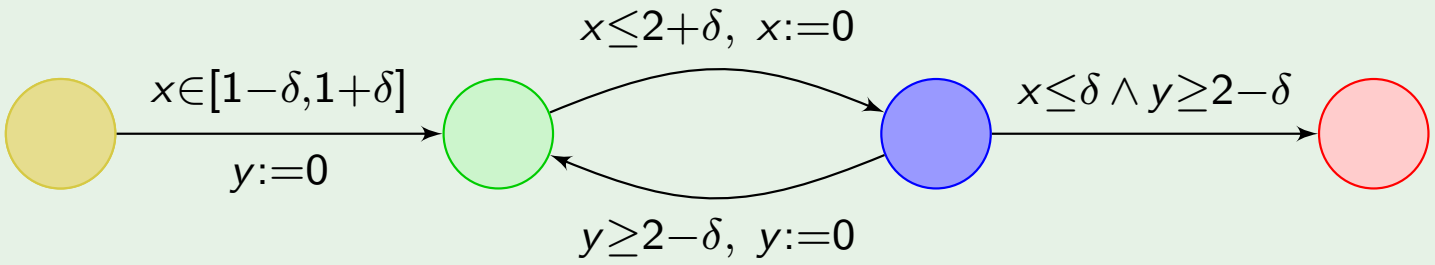
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



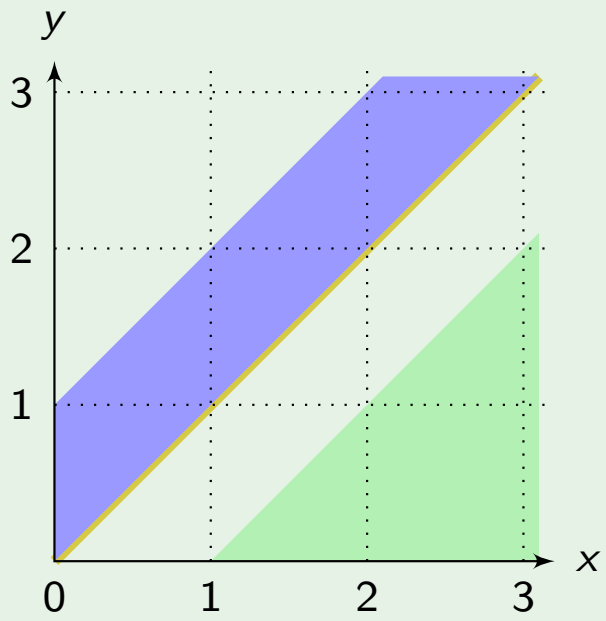


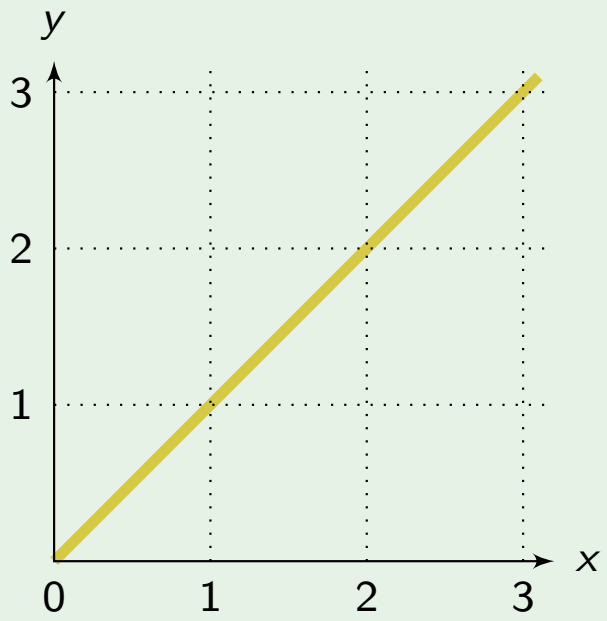


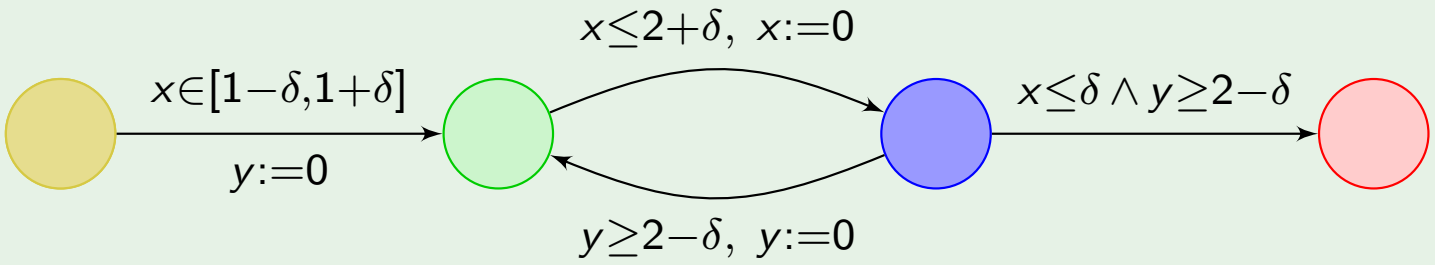
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



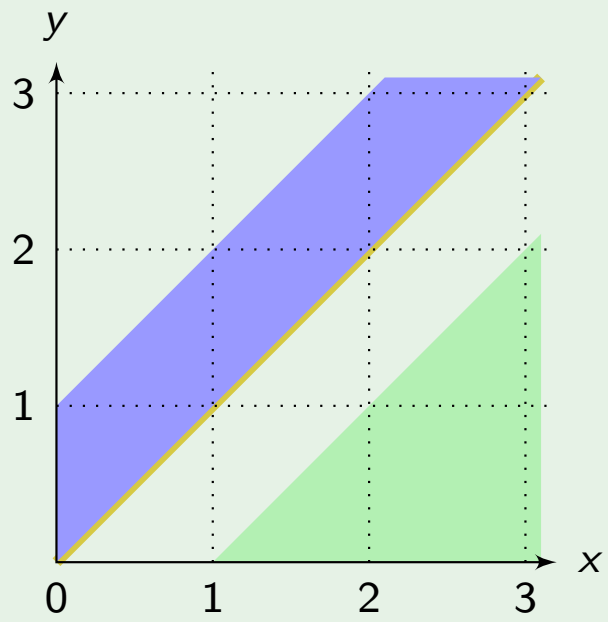


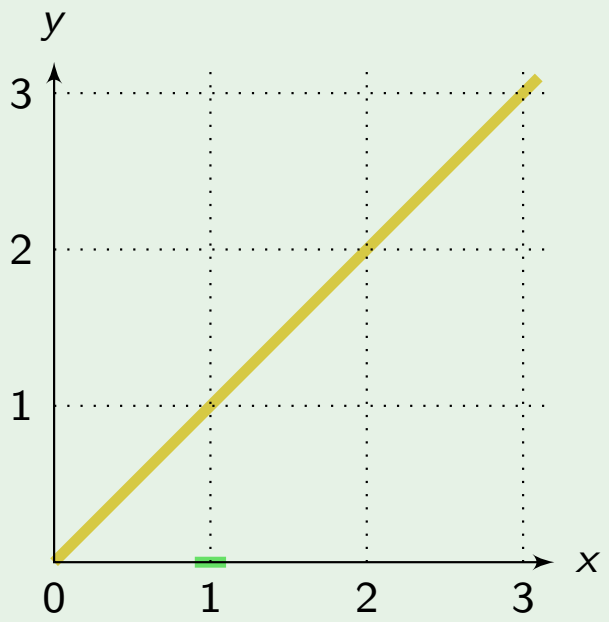


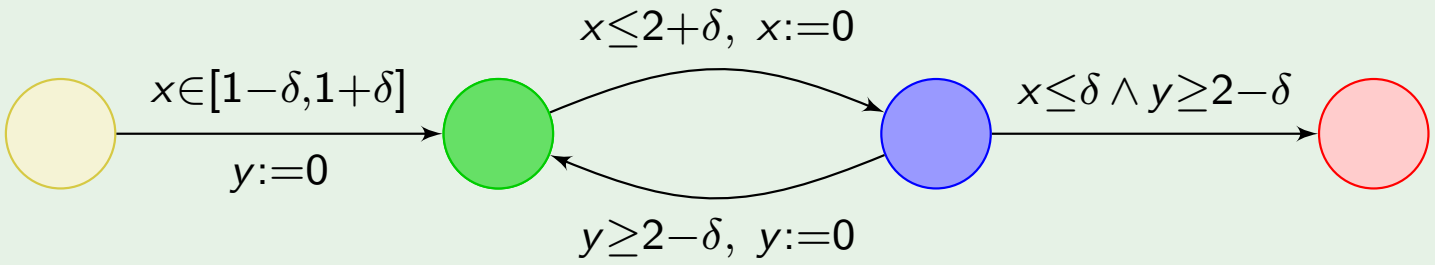
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



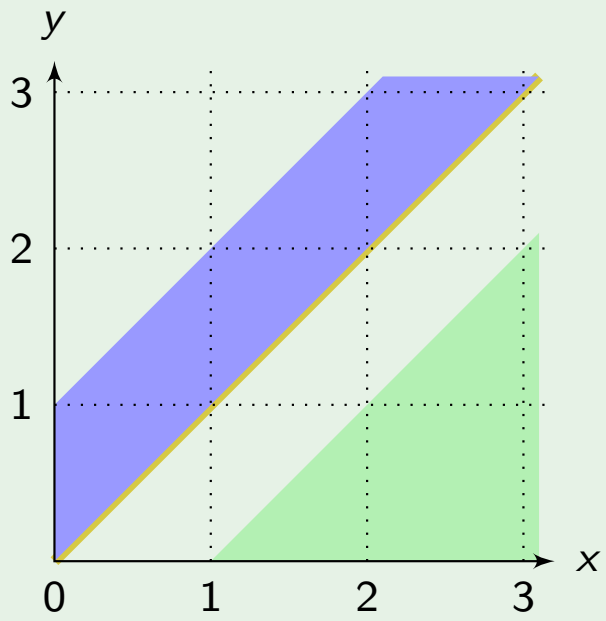


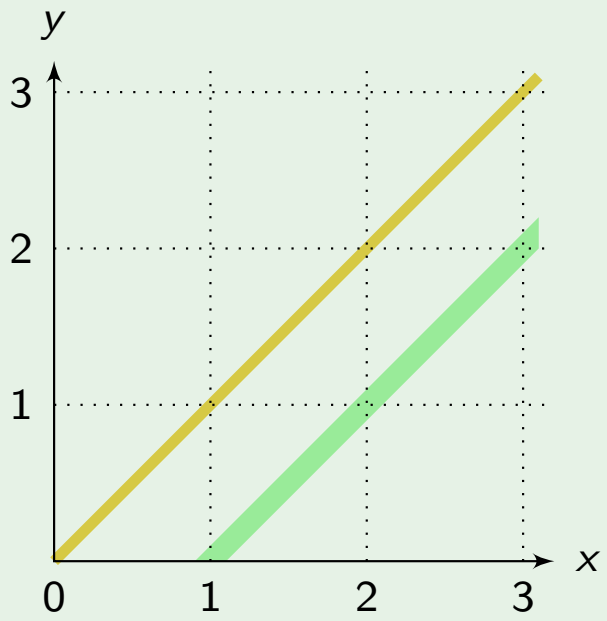


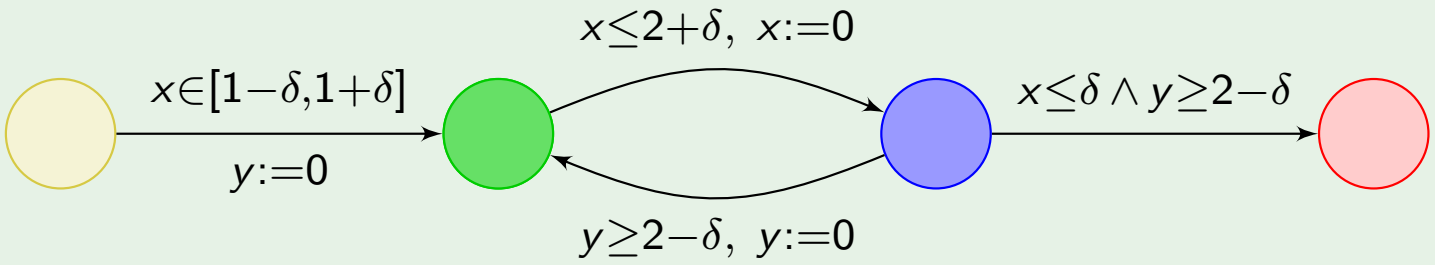
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



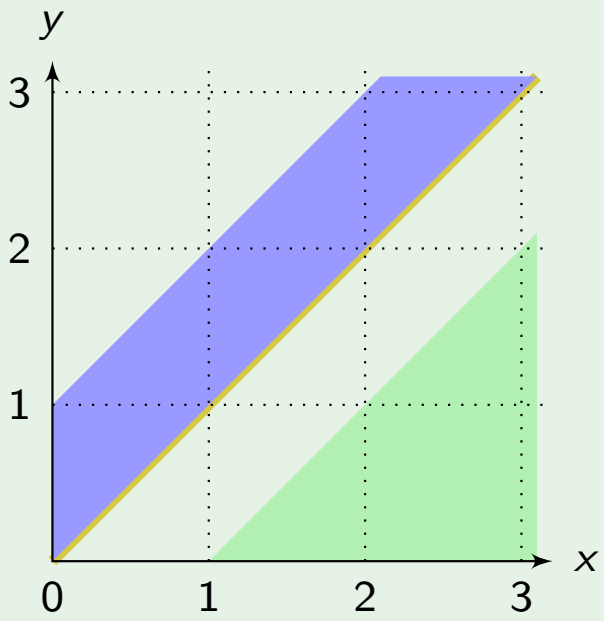


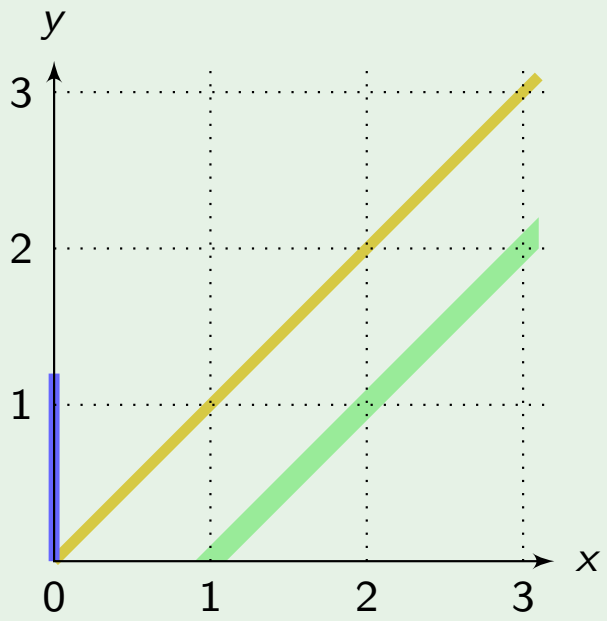


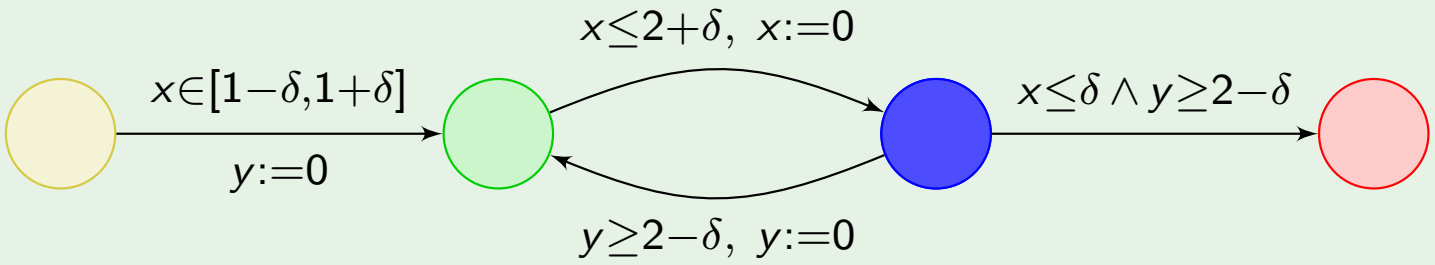
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



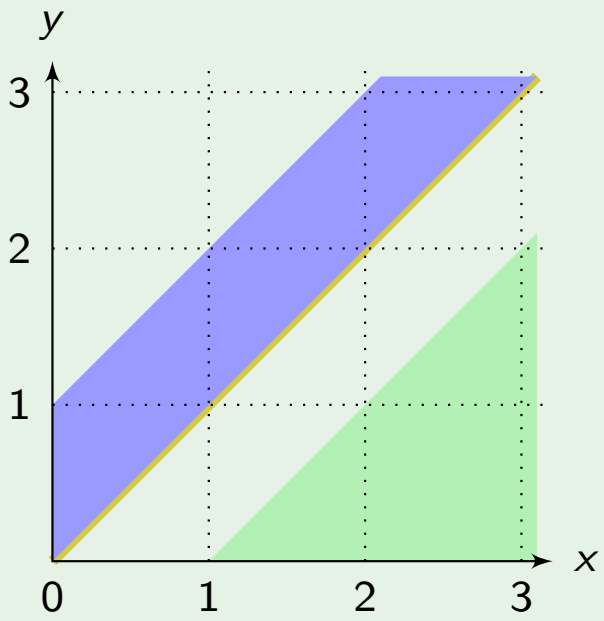


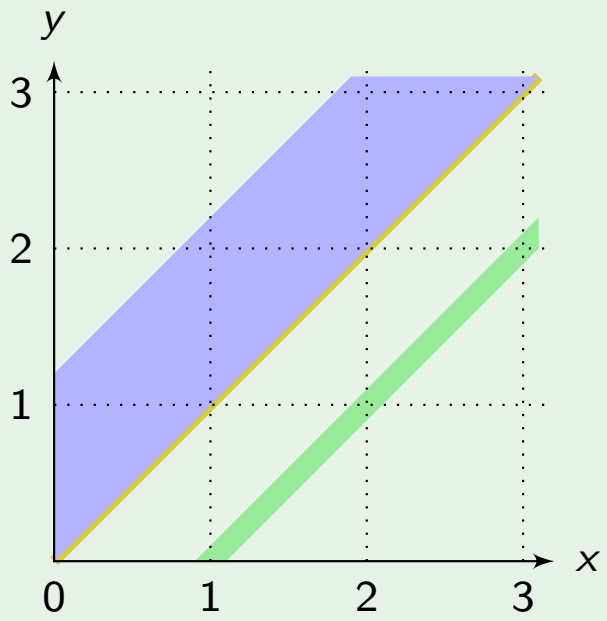


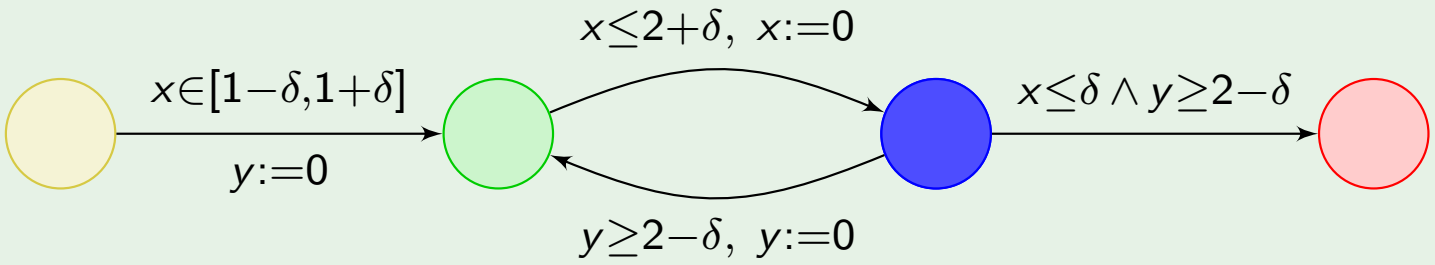
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



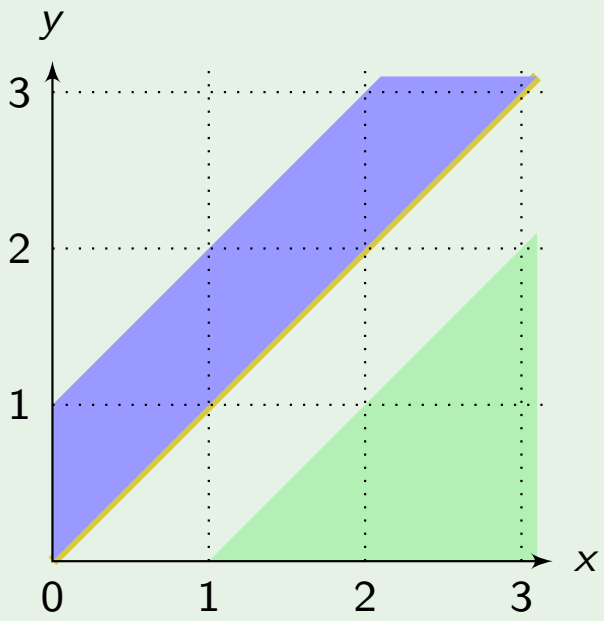


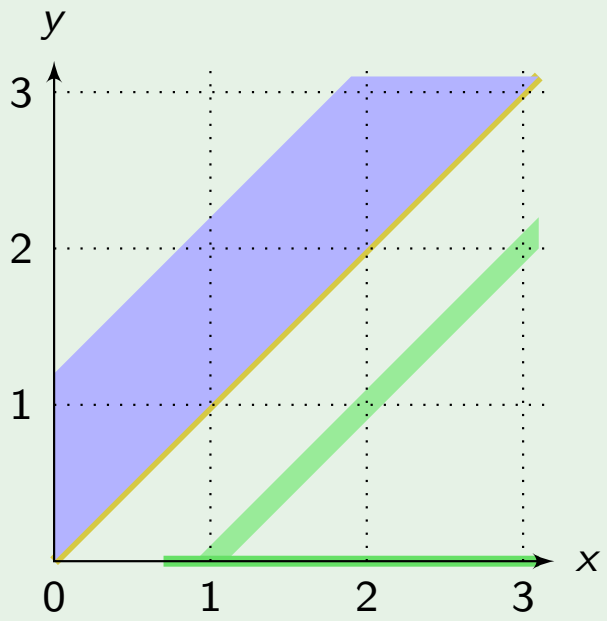


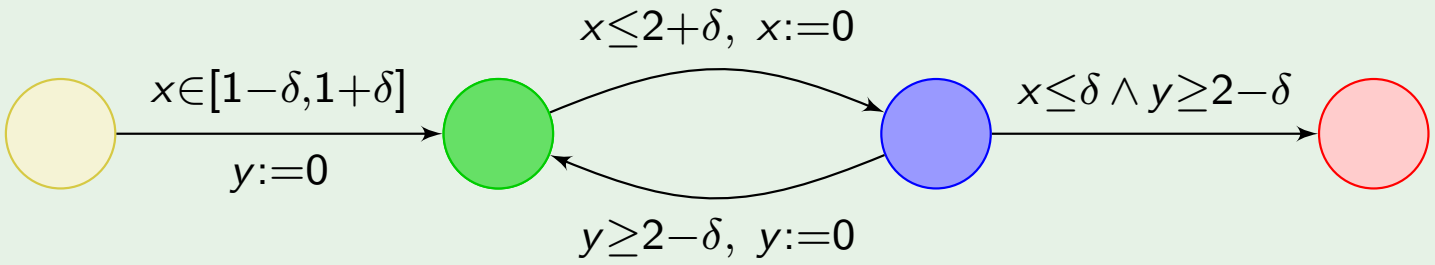
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



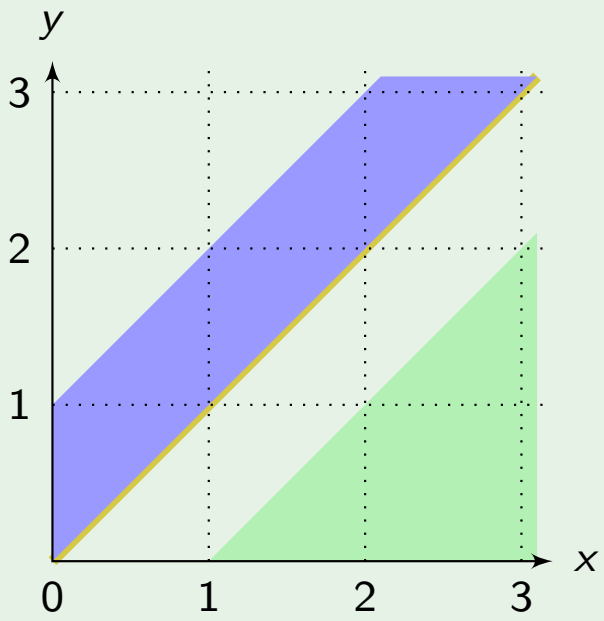


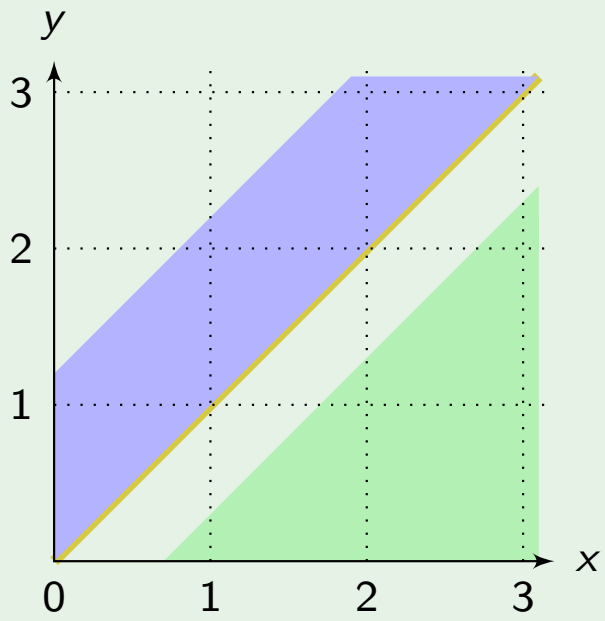


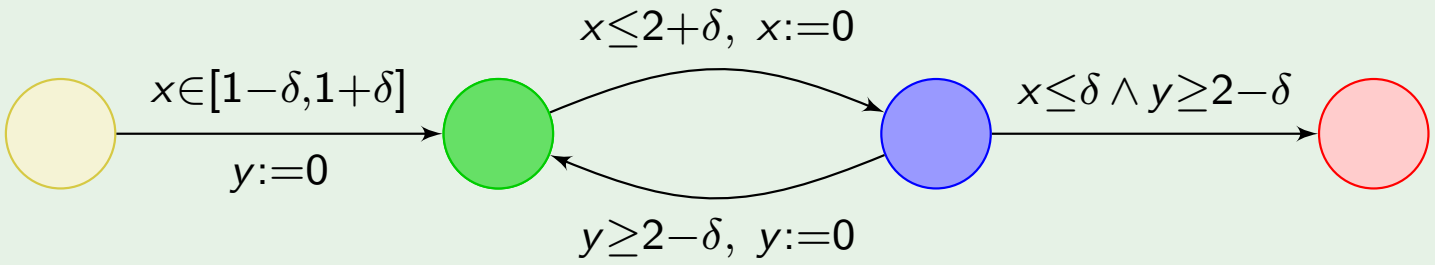
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



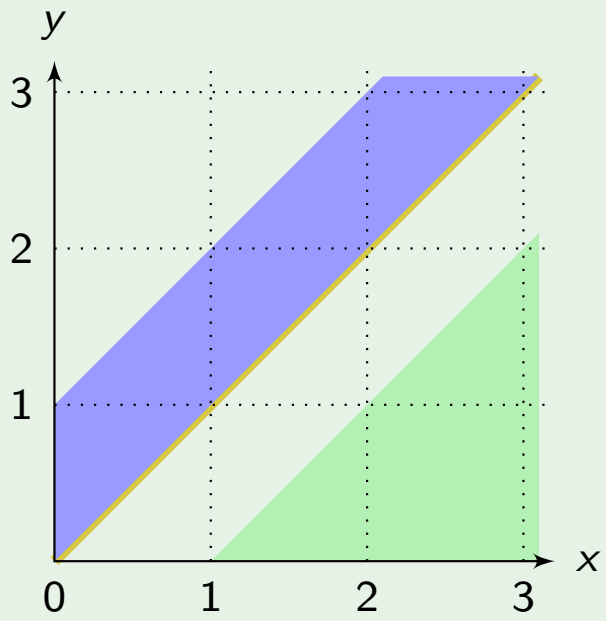


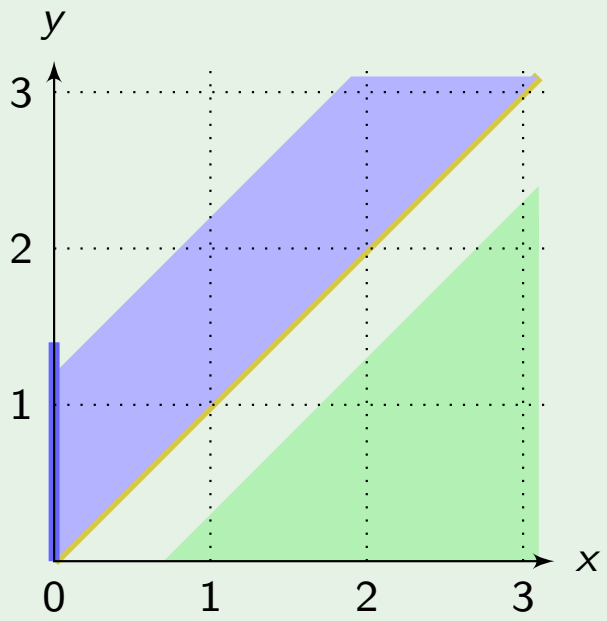


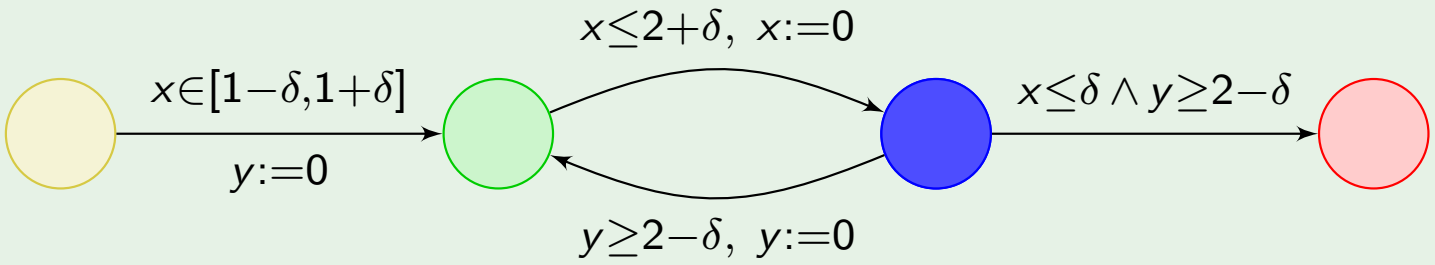
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



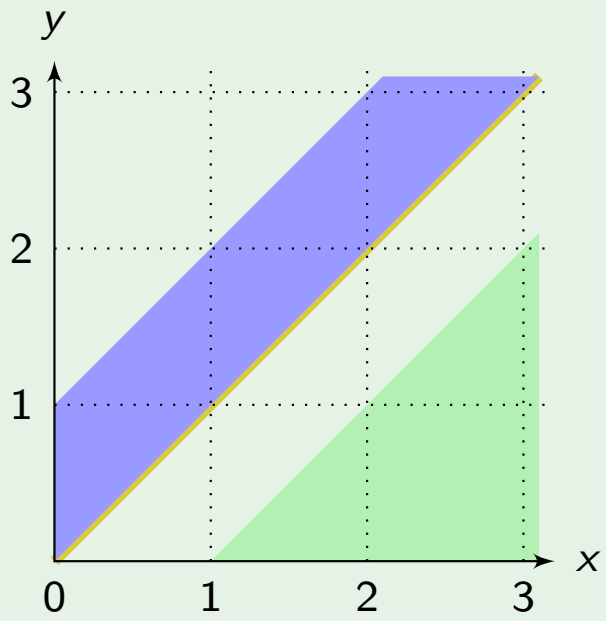


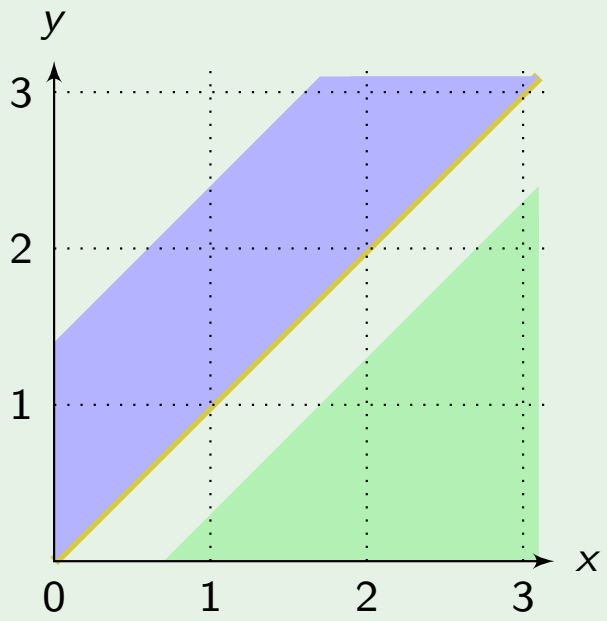


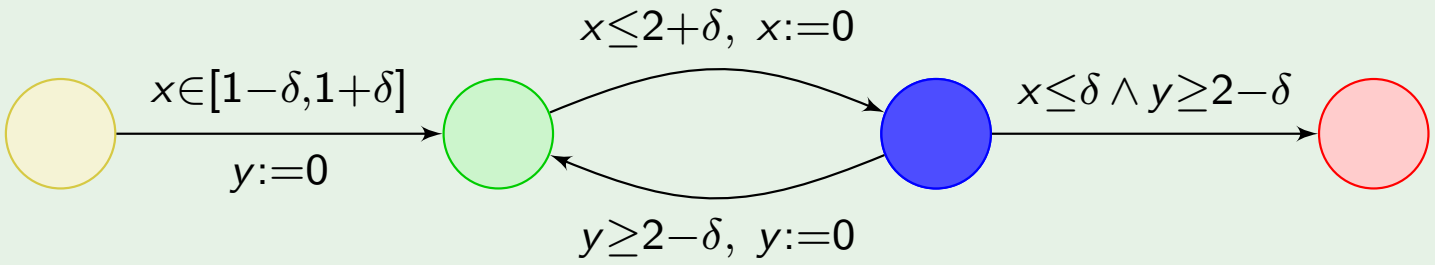
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



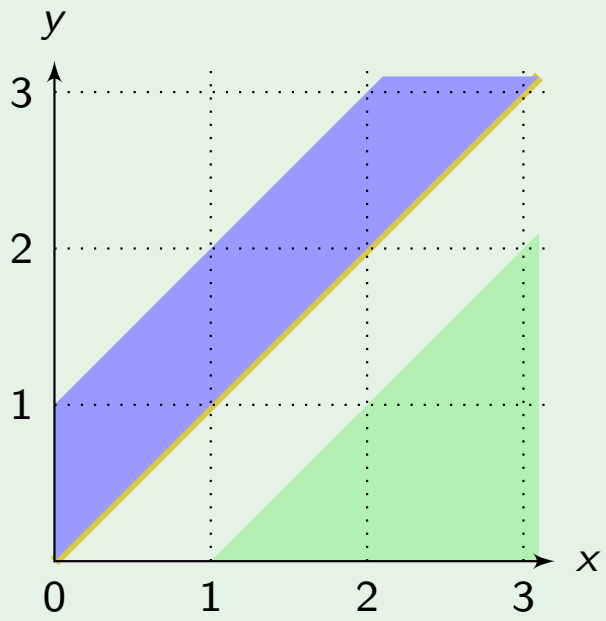


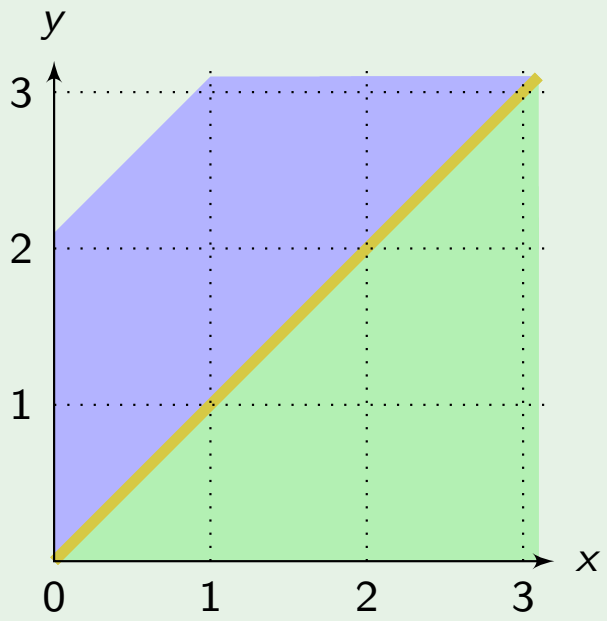


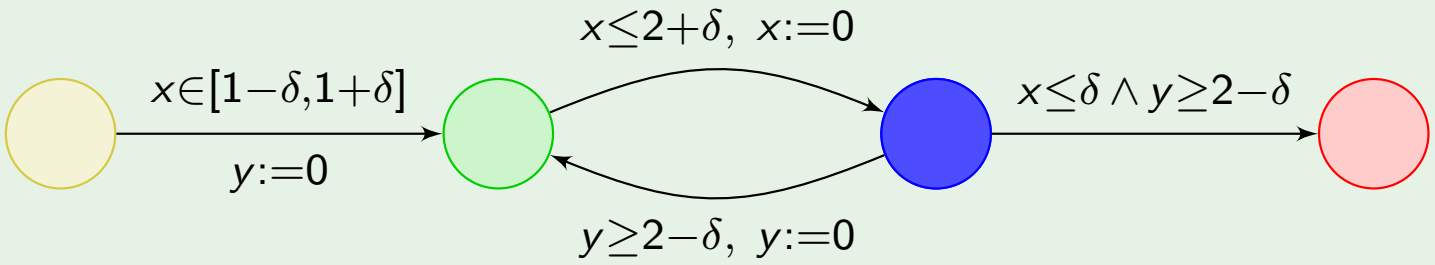
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



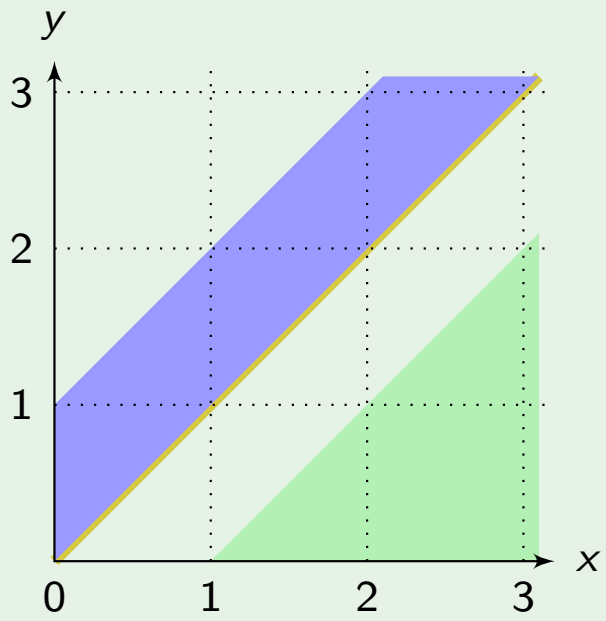


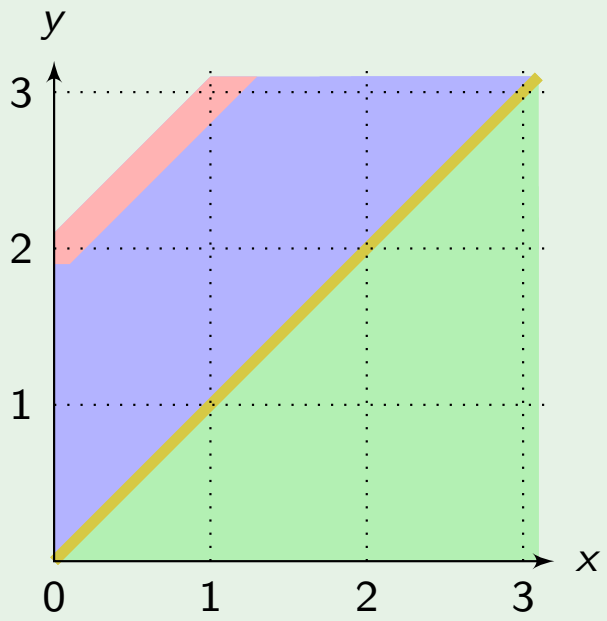


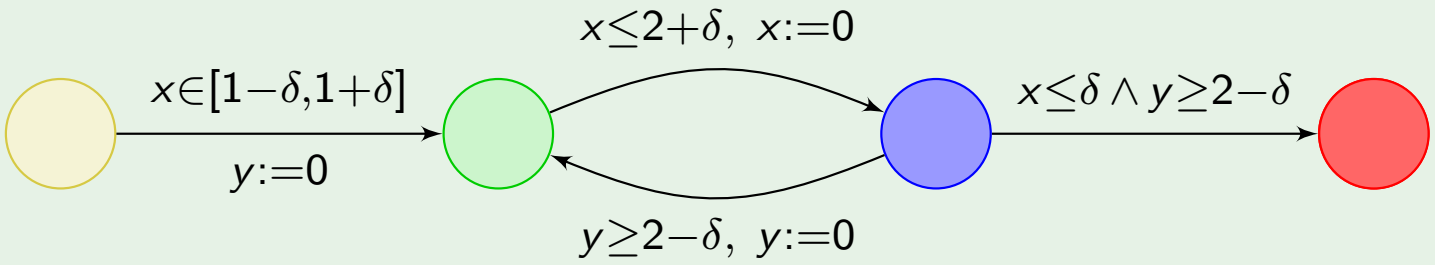
Enlarged semantics for timed automata

a transition can be taken at any time in $[t-\delta; t+\delta]$.

Example



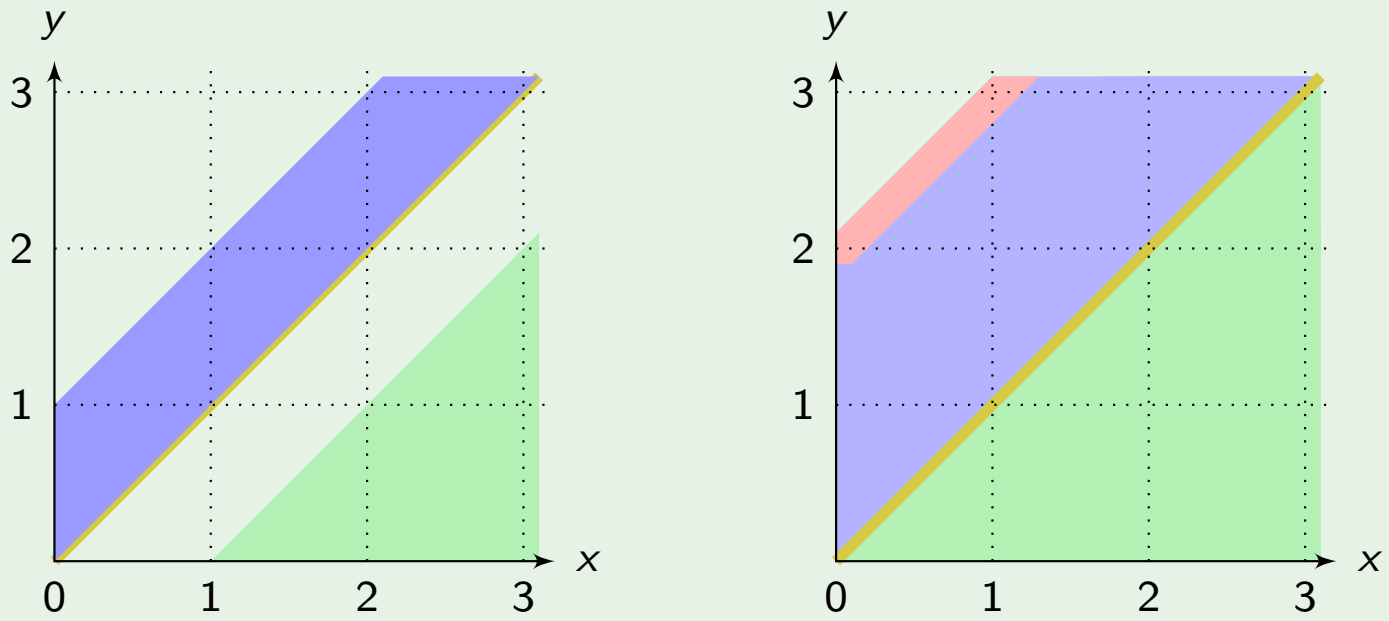




Enlarged semantics for timed automata

a transition can be taken at any time in $[t - \delta; t + \delta]$.

Example



Theorem

Parametrised robust reachability is decidable.

Future challenges

synthesis of **robust** real-time systems

- partial solutions in restricted cases;
- should include imperfect information, ...

Future challenges

synthesis of **robust** real-time systems

- partial solutions in restricted cases;
- should include imperfect information, ...

probabilistic approach to robustness

- our worst-case approach is over-pessimistic;
- average time before exhibiting unexpected behaviours...

Future challenges

synthesis of **robust** real-time systems

- partial solutions in restricted cases;
- should include imperfect information, ...

probabilistic approach to robustness

- our worst-case approach is over-pessimistic;
- average time before exhibiting unexpected behaviours...

quantitative approach to robustness

- there is a global trend towards measures of correctness;
- quantitative robustness: how do such measures evolve when we slightly change the constants of the automaton?