Embedded Systems
2012/13

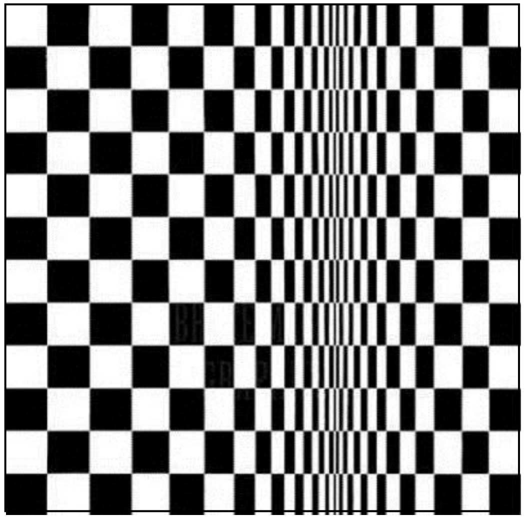

Basilica di Santa Maria di Collemaggio, 1287, L'Aquila

Lecture 4
Overview of Symbolic Models

Maria Domenica Di Benedetto
Giordano Pola
Alessandro Borri
University of L'Aquila

Continuous and Hybrid Systems

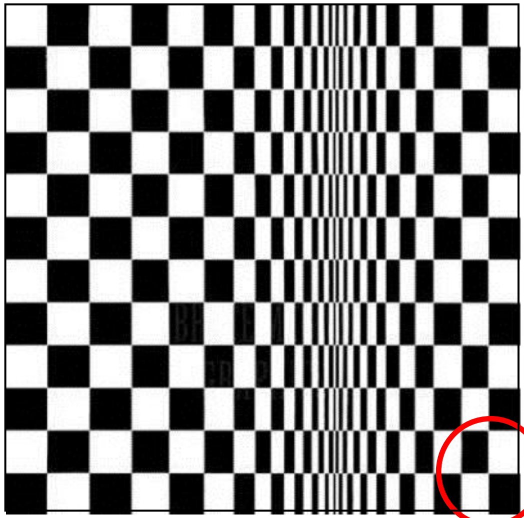
Salvador Dali, The Temptation of St. Anthony



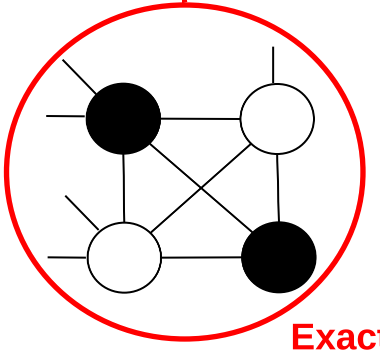
Bridget Riley, Movement in Squares

Continuous and Hybrid Systems

Salvador Dali, The Temptation of St. Anthony



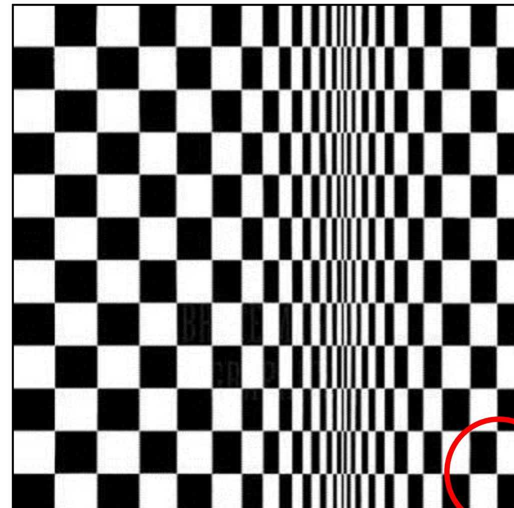
Bridget Riley, Movement in Squares



Exact

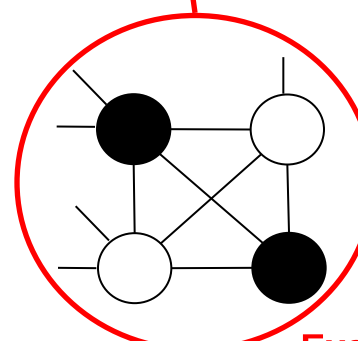
Continuous and Hybrid Systems

Salvador Dali, The Temptation of St. Anthony



Bridget Riley, Movement in Squares

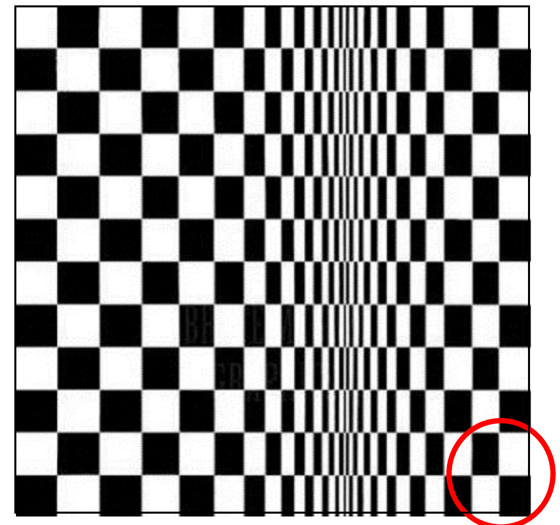
Exact ?



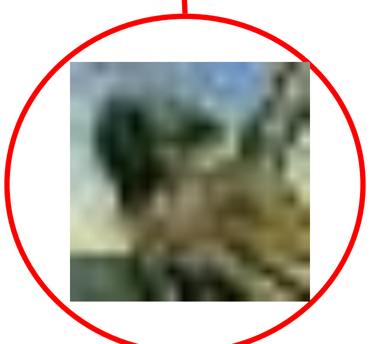
Exact

Continuous and Hybrid Systems

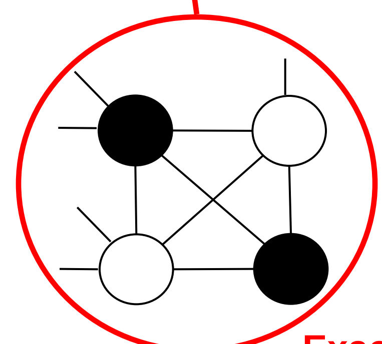
Salvador Dali, The Temptation of St. Anthony



Bridget Riley, Movement in Squares



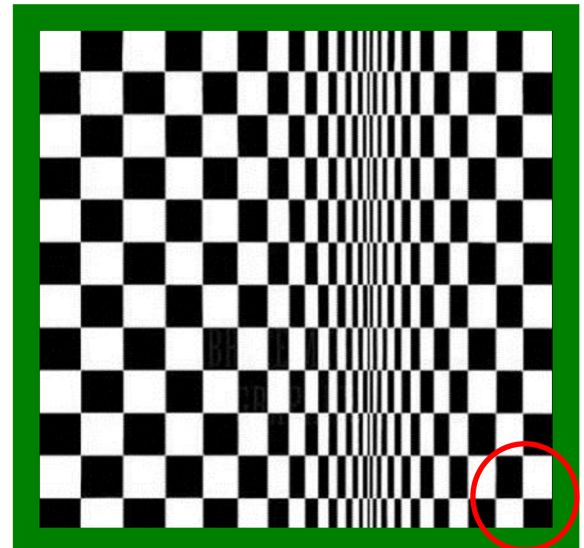
Approximated



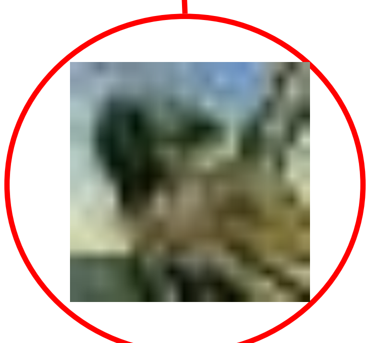
Exact

Continuous and Hybrid Systems

Salvador Dali, The Temptation of St. Anthony



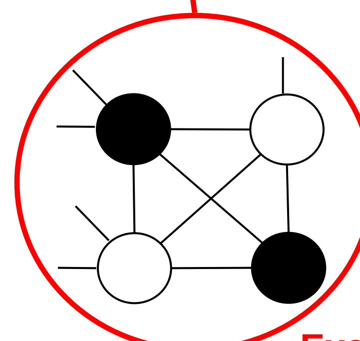
Bridget Riley, Movement in Squares



Approximated

There are systems that:

- Admit symbolic models
- Can be approximated by symbolic models
- Do not admit symbolic models



Exact

Continuous and Hybrid Systems

There are systems that:

- Admit symbolic models
  - Autonomous Systems (no input)
    - Timed Automata *[Alur and Dill, 1992]*
    - Multirate Automata *[Alur et al., 1993]*
    - Rectangular Automata *[Henzinger et al., 1998]*
    - o-minimal hybrid systems *[Lafferriere et al., 2000]*
  - Discrete-time linear control systems *[Tabuada et al., 2006]*
- Can be approximated by symbolic models
  - Incrementally stable nonlinear control systems *[Pola et al., 2008]*
  - Incrementally stable switched nonlinear systems *[Girard et al., 2008]*
  - Incrementally stable nonlinear control systems with disturbances *[Pola et al. 2008]*
- Do not admit symbolic models

Recall: Hybrid Systems

Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \text{I}nit, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states***;*
- $X = \mathbb{R}^n$ *is a set of* **continuous states***;*
- $f(\cdot, \cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field***;*
- $\text{I}nit \subseteq Q \times X$ *is a set of* **initial states***;*
- $\text{D}om(\cdot) : Q \to P(X)$ *is a* **domain***;*
- $E \subseteq Q \times Q$ *is a set of* **edges***;*
- $G(\cdot) : E \to P(X)$ *is a* **guard condition***;*
- $R(\cdot, \cdot) : E \times X \to P(X)$ *is a* **reset map**.

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function $\text{D}om$ assigns a set of continuous states $\text{D}om(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

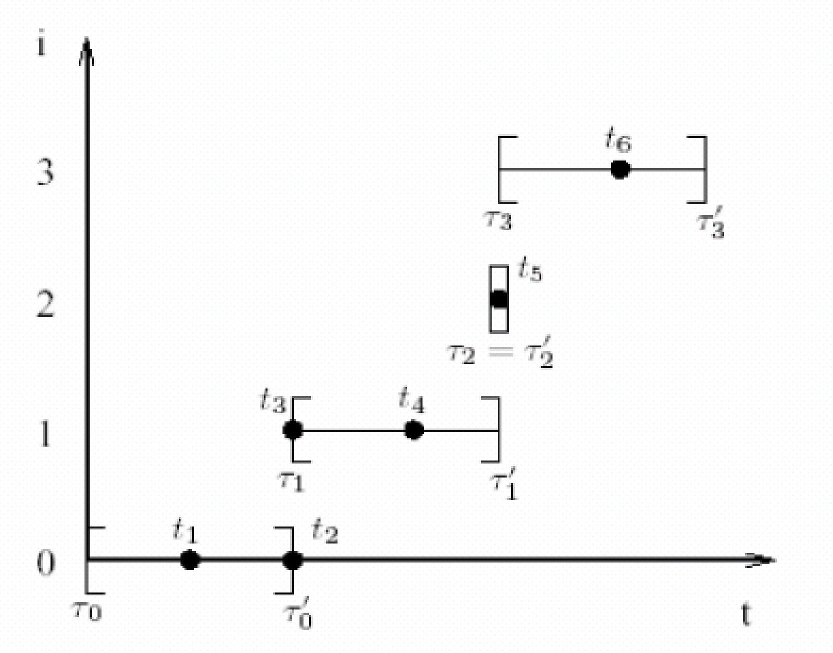
Base temporale

Base temporale (time basis): sequenza di $N+1$ intervalli $I_i = [\tau_i \, , \, \tau_i{}'] \; (i = 0, \ldots, N)$ tali che

- per ogni $i < N$, $\tau_i \leq \tau_i' = \tau_{i+1}$
- se $N < \infty$, $I_N = [\tau_N \, , \, \tau_N']$ o $I_N = [\tau_N \, , \, \tau_N')$

$\mathcal{T}$ denota l'insieme delle basi temporali

Recall: Hybrid Systems



Esecuzione

Una ***esecuzione*** di un sistema ibrido autonomo

$$H = (\ Q, X, Init, f, Dom, E, G, R\ )$$

è una ***traiettoria*** $\chi = (\tau\,, q\,, x)$

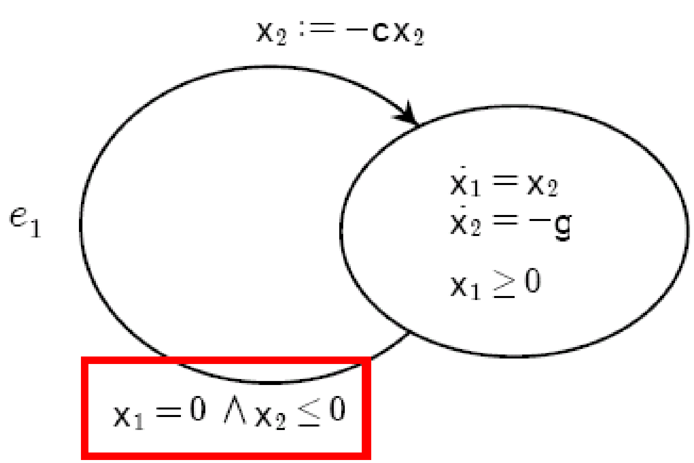
con $\tau \in \mathcal{T}$, $q : i \to Q$, $x : i \to x^i(t)$ tali che

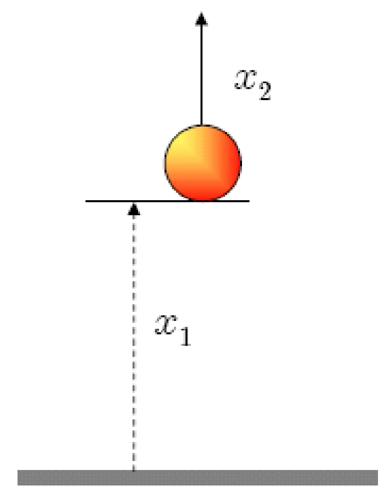
- $(\ q(0)\,, x^0(0)\ ) \in Init$
- evoluzione continua: $t \in [\tau_i\ , \tau_i{}')$, $\dot{x}^i(t) = f(q(i), x^i(t))$ e $x^i(t) \in Dom\,(q(i))$
- evoluzione discreta: per ogni $i = 0, \ldots, N - 1$,

$e = (q(i), q(i+1)) \in E$,

$x^i(\tau_i{}') \in G(e)$ e $x^{i+1}(\tau_{i+1}) \in R(e,\ x^i(\tau_i{}'))$

Recall: Hybrid Systems





Rectangular Automata

Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \mathrm{Init}, D, E, G, R)$, where*

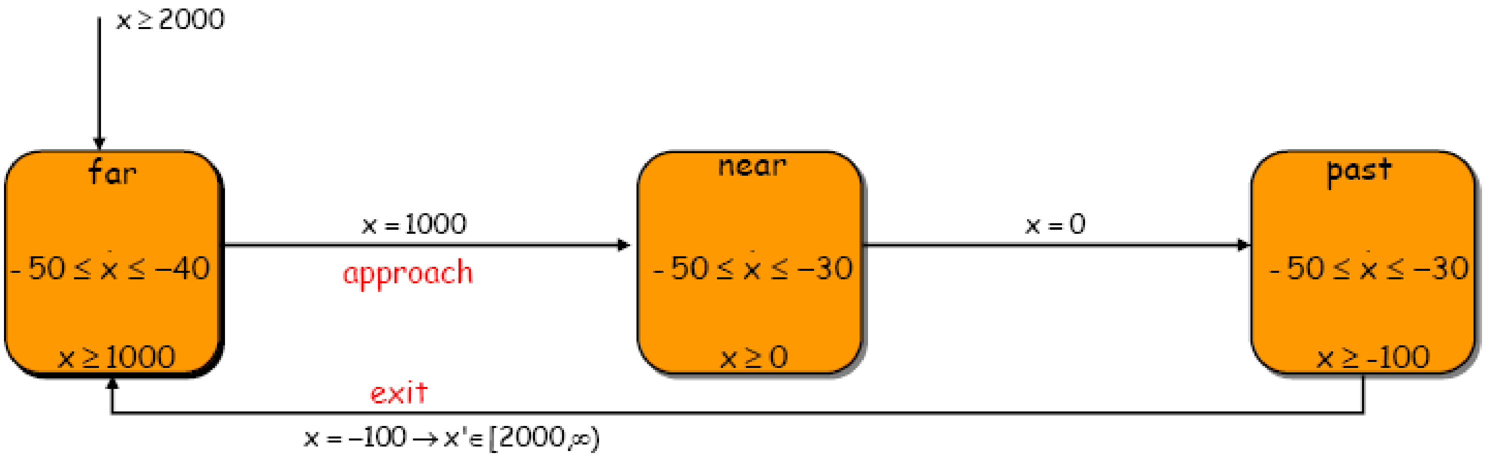
- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot,\cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; Rectangular set
- $\mathrm{Init} \subseteq Q \times X$ *is a set of* **initial states**; Rectangular set
- $\mathrm{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; Rectangular set
- $R(\cdot,\cdot) : E \times X \to P(X)$ *is a* **reset map**. Rectangular set

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\mathrm{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Rectangular Automata

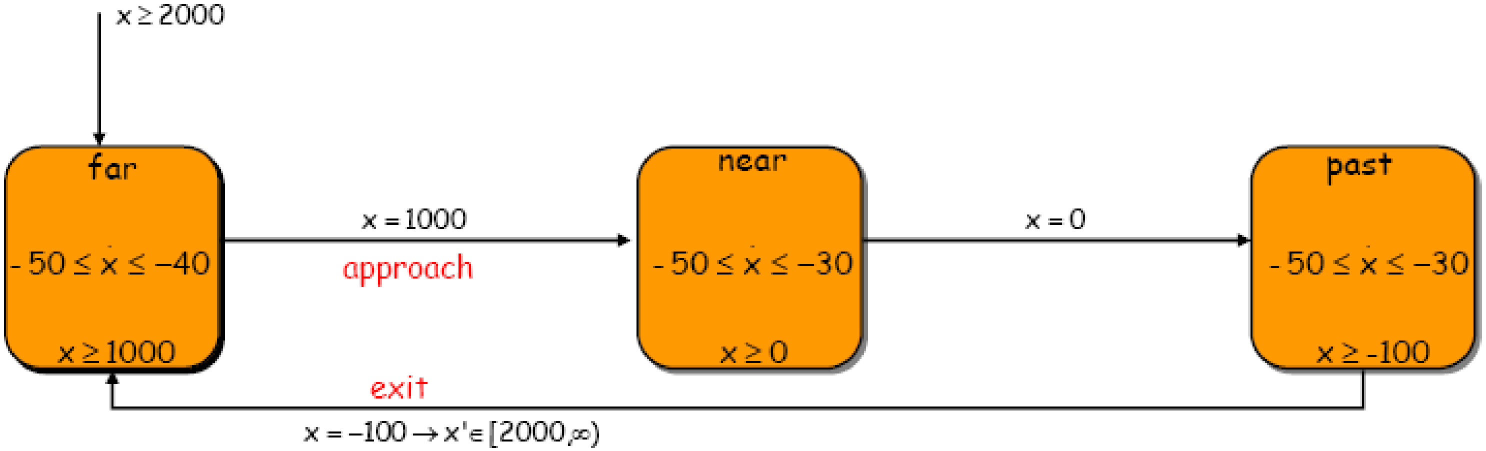
Rectangular sets : $\bigwedge_i x_i \sim c_i \quad \sim \in \{<, \leq, =, \geq, >\}, c_i \in Q$



Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

Rectangular Automata

Rectangular sets : $\bigwedge_i x_i \sim c_i \quad \sim \in \{<, \leq, =, \geq, >\}, c_i \in Q$



Rectangular hybrid automata are **initialized** if the following holds:

After a discrete transition, if the differential equation for a variable changes, then the variable must be reset to a fixed interval.

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

Rectangular Automata

Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \text{Init}, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot, \cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; — Rectangular set
- $\text{Init} \subseteq Q \times X$ *is a set of* **initial states**; — Rectangular set
- $\text{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; — Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; — Rectangular set
- $R(\cdot, \cdot) : E \times X \to P(X)$ *is a* **reset map**. — Rectangular set

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\text{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Multirate Automata

Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \mathrm{Init}, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot,\cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; Rectangular set
- $\mathrm{Init} \subseteq Q \times X$ *is a set of* **initial states**; Singleton set ~~Rectangular set~~
- $\mathrm{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; Rectangular set
- $R(\cdot,\cdot) : E \times X \to P(X)$ *is a* **reset map**. Singleton set ~~Rectangular set~~

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\mathrm{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Multirate Automata

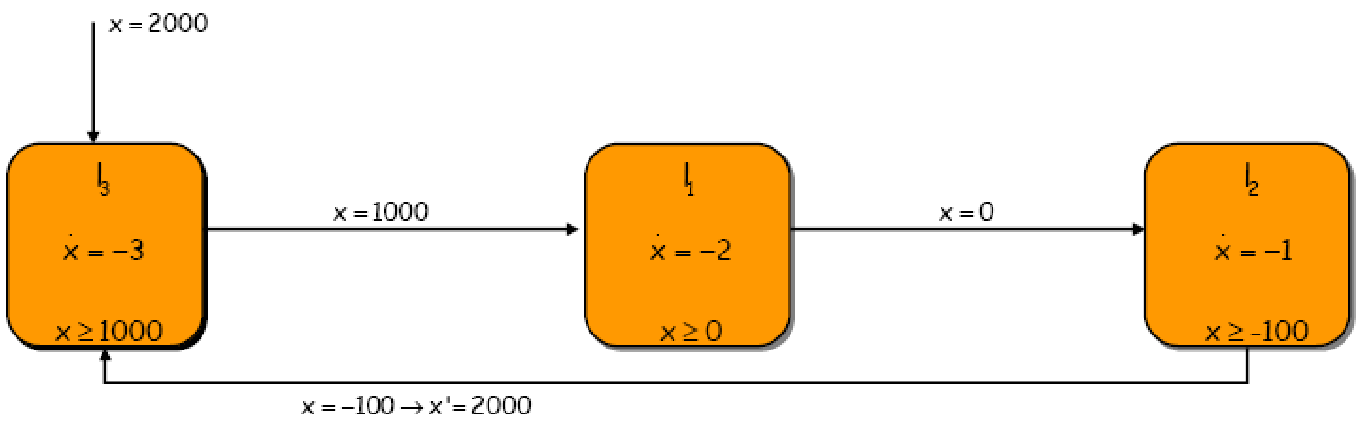
Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \text{Init}, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot, \cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; Rectangular set
- $\text{Init} \subseteq Q \times X$ *is a set of* **initial states**; Singleton set
- $\text{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; Rectangular set
- $R(\cdot, \cdot) : E \times X \to P(X)$ *is a* **reset map**. Singleton set

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\text{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Multirate Automata



Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

Multirate Automata

Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \text{Init}, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot,\cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; Rectangular set
- $\text{Init} \subseteq Q \times X$ *is a set of* **initial states**; Singleton set
- $\text{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; Rectangular set
- $R(\cdot,\cdot) : E \times X \to P(X)$ *is a* **reset map**. Singleton set

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\text{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Timed Automata

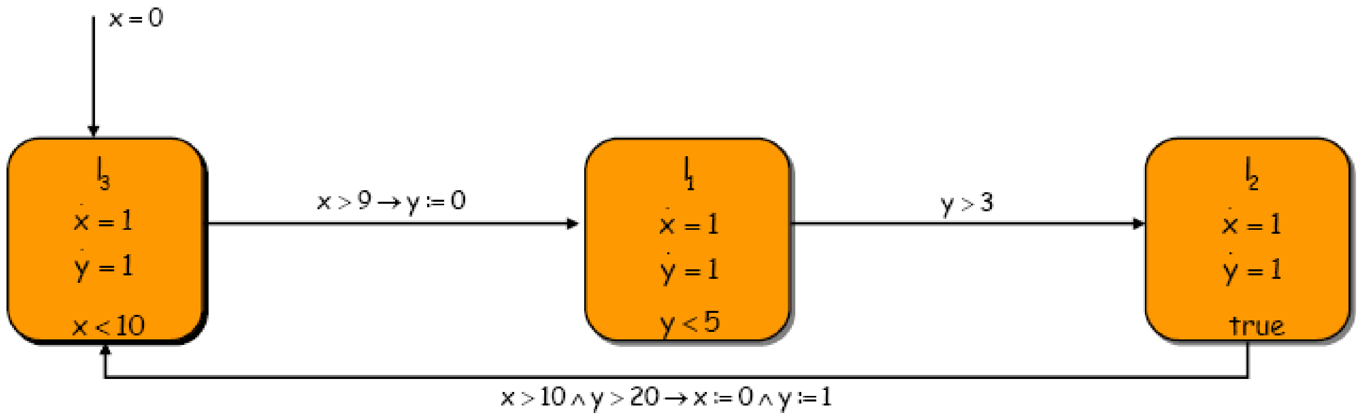
Definition 3.1 (Hybrid Automaton) *A hybrid automaton H is a collection $H = (Q, X, f, \mathrm{Init}, D, E, G, R)$, where*

- $Q = \{q_1, q_2, \ldots\}$ *is a set of* **discrete states**;
- $X = \mathbb{R}^n$ *is a set of* **continuous states**;
- $f(\cdot,\cdot) : Q \times X \to \mathbb{R}^n$ *is a* **vector field**; Clock f(q,x) = 1
- $\mathrm{Init} \subseteq Q \times X$ *is a set of* **initial states**; Singleton set
- $\mathrm{Dom}(\cdot) : Q \to P(X)$ *is a* **domain**; Rectangular set
- $E \subseteq Q \times Q$ *is a set of* **edges**;
- $G(\cdot) : E \to P(X)$ *is a* **guard condition**; Rectangular set
- $R(\cdot,\cdot) : E \times X \to P(X)$ *is a* **reset map**. Singleton set

Recall that $P(X)$ denotes the power set (set of all subsets) of X. The notation of Definition 3.1 suggests, for example, that the function Dom assigns a set of continuous states $\mathrm{Dom}(q) \subseteq \mathbb{R}^n$ to to each discrete state $q \in Q$. We refer to $(q, x) \in Q \times X$ as the *state* of H.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Timed Automata

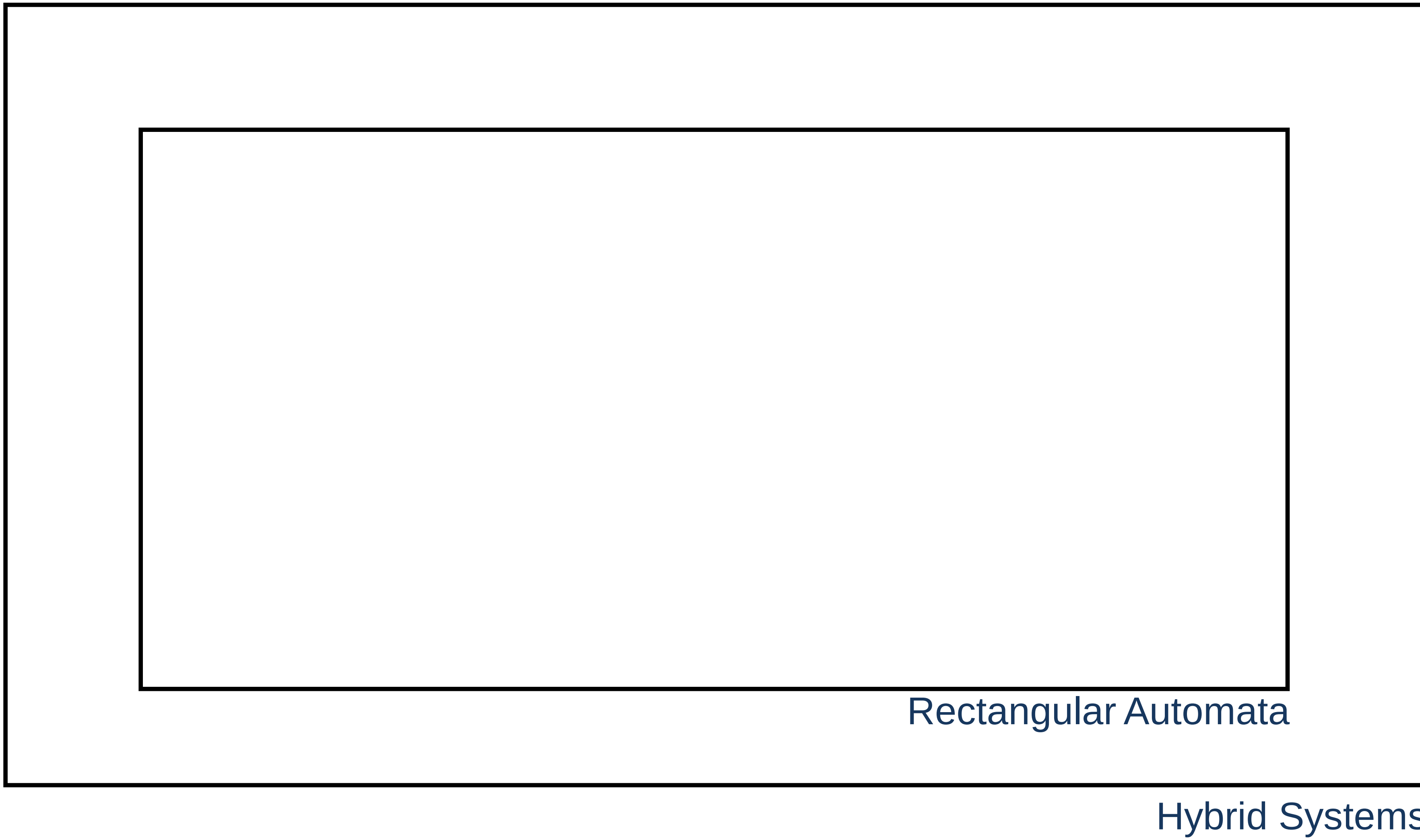


Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

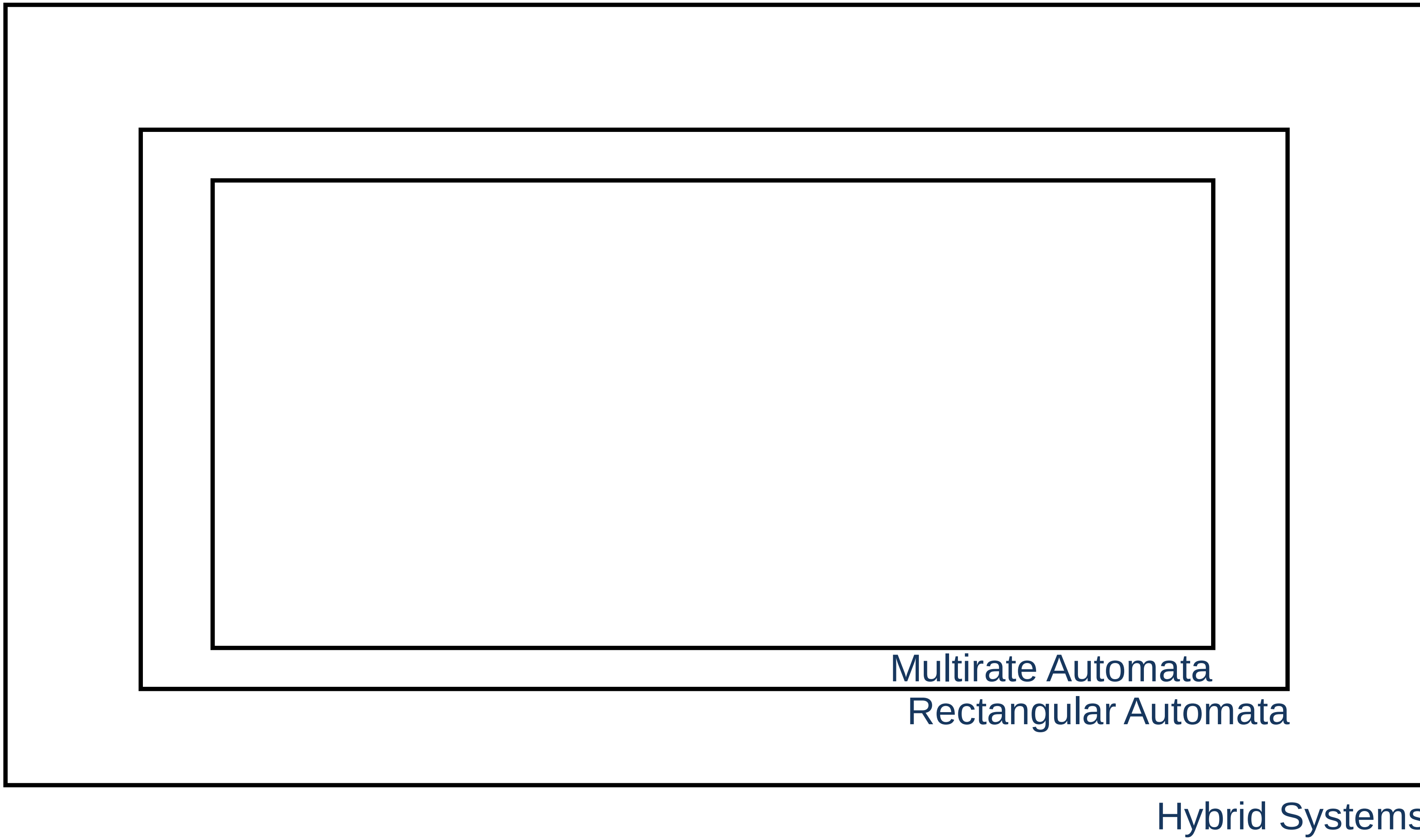
Summarizing …

Hybrid Systems

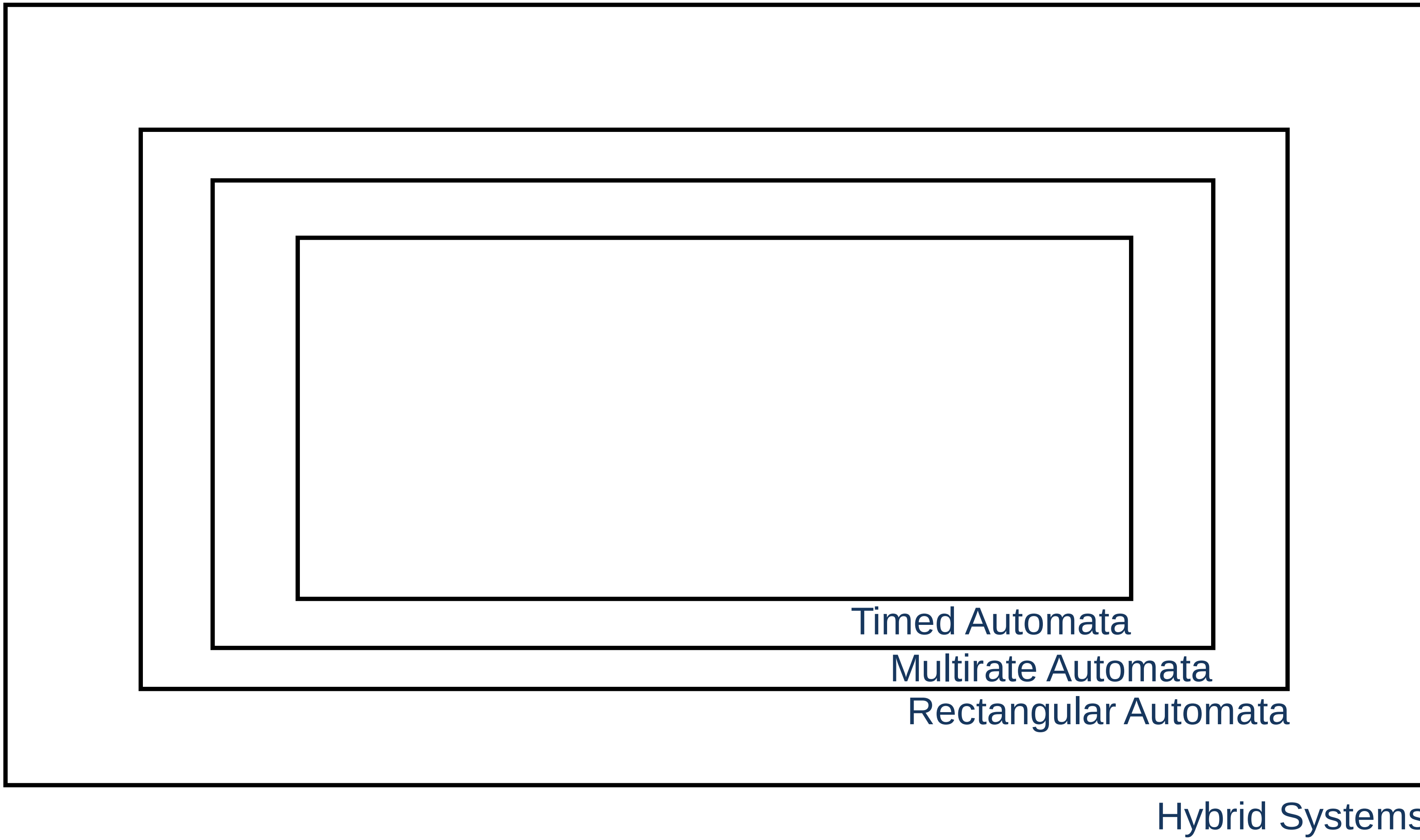
… for the time being…



… for the time being…



… for the time being…



Timed Automata

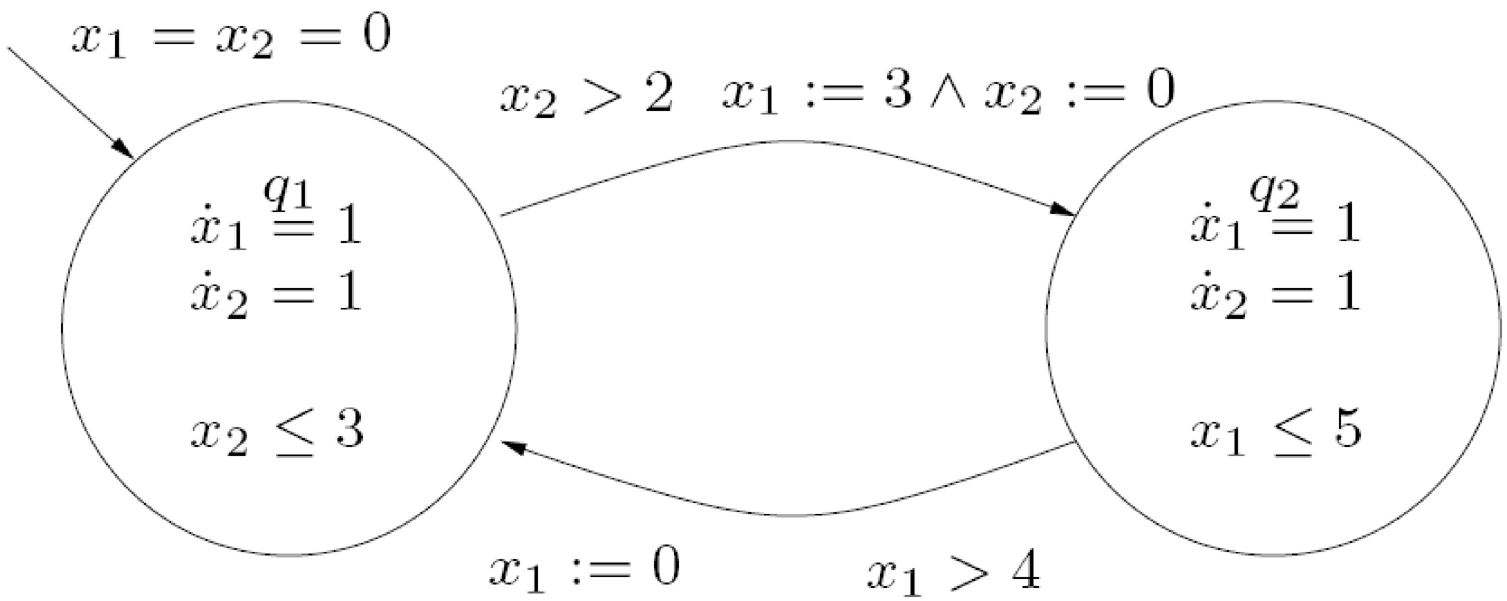


Figure 6.2: Example of a timed automaton.

Timed Automata

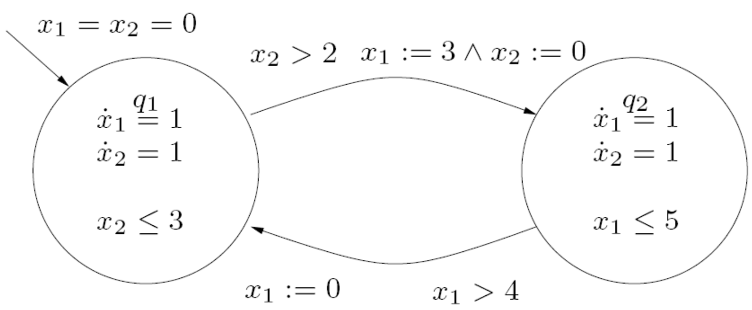


Figure 6.2: Example of a timed automaton.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

Hybrid Automaton

- $Q = \{q_1, q_2\}$;
- $X = \mathbb{R}^2$;
- $f(q_1, x) = f(q_1, x) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$;
- $\text{I}nit = \{(q_1, (0, 0))\}$;
- $\text{D}om(q_1) = \{x \in \mathbb{R}^2 \mid x_2 \leq 3\}$, $\text{D}om(q_2) = \{x \in \mathbb{R}^2 \mid x_1 \leq 5\}$;
- $E = \{(q_1, q_2), (q_2, q_1)\}$;
- $G(q_1, q_2) = \{x \in \mathbb{R}^2 \mid x_2 > 2\}$, $G(q_2, q_1) = \{x \in \mathbb{R}^2 \mid x_1 > 4\}$;
- $R(q_1, q_2, x) = \{(3, 0)\}$, $R(q_2, q_1, x) = \{(0, x_2)\}$

Timed Automata

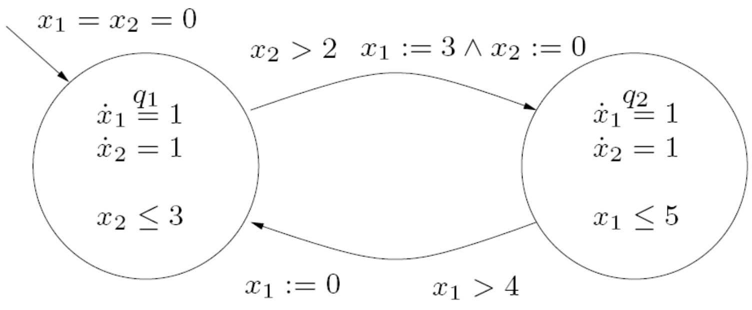


Figure 6.2: Example of a timed automaton.

Taken from: John Lygeros, Notes for an ENSIETA short course, February, 2004

I can partition the state space in
- open (possibly infinite) rectangles
- open triangles
- open lines
- points

Timed Automata **admit** Finite Bisimulation!

Region Graph

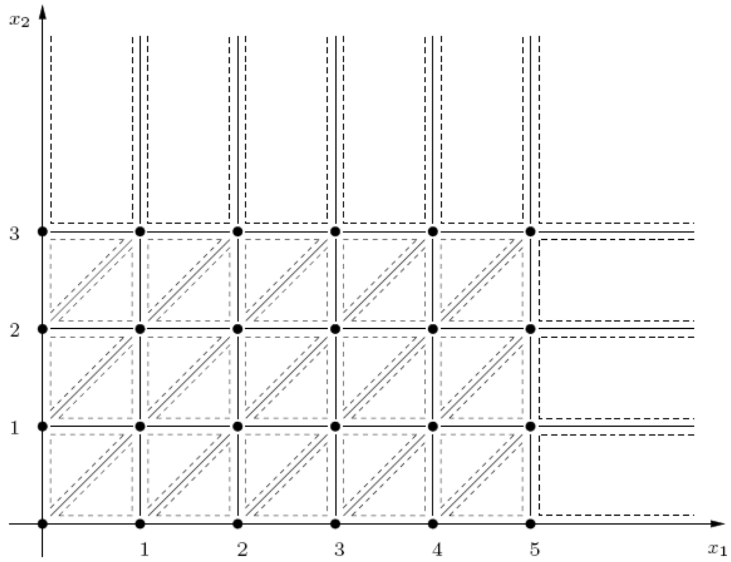


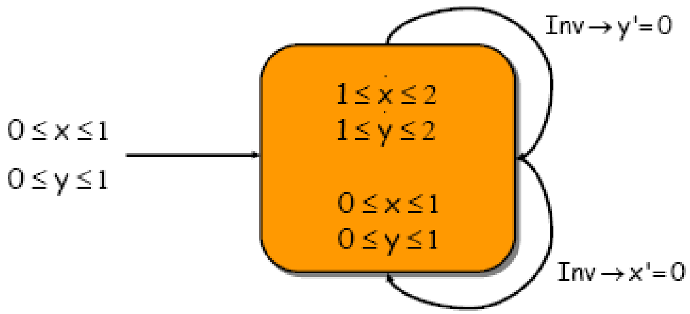
Figure 6.3: Region graph for the timed automaton of Figure 6.2.

Multirate Automata

Inizialized Multirate Automata **admit** Finite Bisimulation!

Rectangular Automata

Inizialized Rectangular Automata **do not admit** Finite Bisimulation!

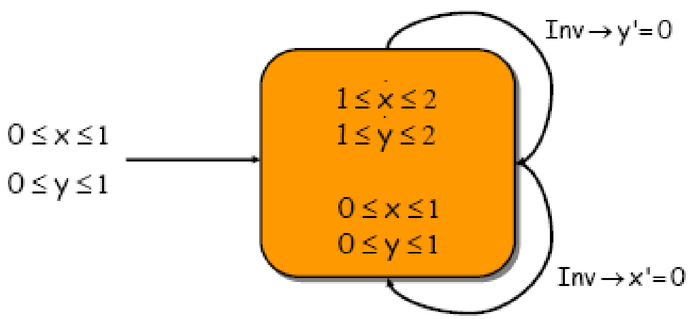


Bisimulation algorithm never terminates

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

Rectangular Automata

Inizialized Rectangular Automata **do not admit** Finite Bisimulation!



Bisimulation algorithm never terminates

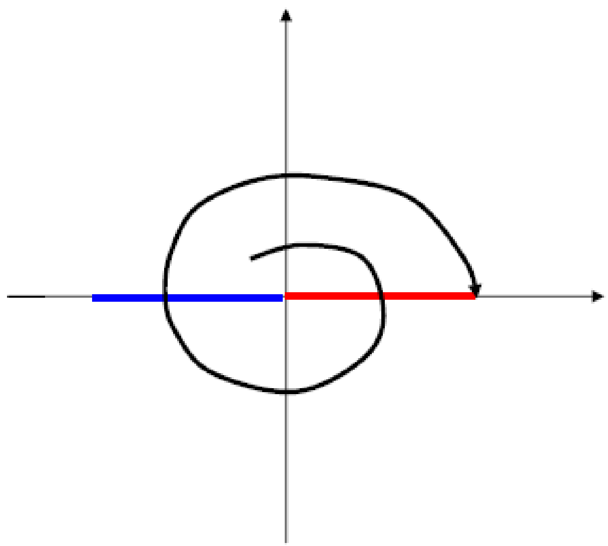
Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

… but

… all initialized rectangular automata admit a finite language equivalence quotient which can be constructed effectively.

… bad news

More complicated dynamics?



Sets

$P_1 = \{(x,0) \mid 0 \le x \le 4\}$

$P_2 = \{(x,0) \mid -4 \le x < 0\}$

$P_3 = R^2 \setminus (P_1 \cup P_2)$

Dynamics

$$\dot{x}_1 = 0.2x_1 + x_2$$

$$\dot{x}_2 = -x_1 + 0.2x_2$$

Bisimulation algorithm never terminates !!

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

Basic problems

Finite bisimulations of continuous dynamical systems

Given a vector field $F(x)$ and a finite partition of R^n

1. Does there exist a finite bisimulation ?
2. Can we compute it ?

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

o-minimal hybrid systems

Reminder

Representation issues

- Symbolic representation for infinite sets
- Rectangular sets ? Semi-linear ? Semi-algebraic ?

Operations on sets

- Boolean (logical) operations
- Can we compute Pre and Post ?
- Is our representation closed under Pre and Post ?

Algorithmic termination (decidability)

- No guarantee for infinite transition systems
- We need "nice" alignment of sets and flows
- Globally finite properties

Penn

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

o-minimal hybrid systems

First-order logic

Every theory of the reals has an associated language

$$(\Re, <, +, -, 0, 1)$$

Universe: $\Re$ — Relation: $<$ — Functions: $+, -$ — Constants: $0, 1$

Variables : $x_1, x_2, x_3, \ldots$

TERMS : Variables, constants, or functions of them

$$x_1 - x_2 + 1, 1 + 1, -x_3$$

ATOMIC FORMULAS : Apply the relation and equality to the terms

$$x_1 + x_2 < -1, 2x_1 = 1, x_1 = x_3$$

(FIRST ORDER) FORMULAS : Atomic formulas are formulas

If φ_1, φ_2 are formulas, then $\varphi_1 \vee \varphi_2, \neg\varphi_1, \forall x.\varphi_1, \exists x.\varphi_1$

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

o-minimal hybrid systems

First-order logic

Useful languages

$(\Re, <, +, -, 0, 1)$	$\forall x \forall y (x + 2y \geq 0)$
$(\Re, <, +, -, \times, 0, 1)$	$\exists x . ax^2 + bx + c = 0$
$(\Re, <, +, -, \times, e^x, 0, 1)$	$\exists t . (t \geq 0) \wedge (y = e^t x)$

A theory of the reals is **decidable** if there is an algorithm which in a finite number of steps will decide whether a formula is true or not

A theory of the reals admits **quantifier elimination** if there is an algorithm which will eliminate all quantified variables.

$$\exists x . ax^2 + bx + c = 0 \equiv b^2 - 4ac \geq 0$$

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

First-order logic

Theory	Decidable ?	Quant. Elim. ?
$(\Re, <, +, -, 0, 1)$	YES	YES
$(\Re, <, +, -, \times, 0, 1)$	YES	YES
$(\Re, <, +, -, \times, e^x, 0, 1)$	?	NO

Tarski's result : Every formula in $(\Re, <, +, -, \times, 0, 1)$ can be decided

1. Eliminate quantified variables
2. Quantifier free formulas can be decided

Penn

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

o-minimal hybrid systems

O-Minimal Theories

A definable set is $Y = \{(x_1, x_2, \ldots, x_n) \in \Re^n \mid \varphi(x_1, \ldots, x_n)\}$

A theory of the reals is called **o-minimal** if every definable subset of the reals is a finite union of points and intervals

Example: $Y = \{(x) \in \Re \mid p(x) \geq 0\}$ for polynomial p(x)

Recent o-minimal theories

$$(\Re, <, +, -, 0, 1)$$

$$(\Re, <, +, -, \times, 0, 1)$$

$$(\Re, <, +, -, \times, e^x, 0, 1)$$

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA

o-minimal hybrid systems

Basic answers

Finite bisimulations of continuous dynamical systems

Consider a vector field X and a finite partition of R^n where

1. The flow of the vector field is definable in an o-minimal theory
2. The finite partition is definable in the same o-minimal theory

Then a finite bisimulation always exists.

Taken from: Agung Julius, Notes for the course on Hybrid Systems at UPENN, USA