

Modeling & Control of Hybrid Systems

Chapter 1 — Introduction

Overview

1. Overview of the course
2. Hybrid systems: Motivating examples
3. Hybrid automata
4. Examples of hybrid systems
5. Examples with Zeno behavior

1. Overview of the course

- Lecturers: Bart De Schutter and Maurice Heemels
- Web site:
`http://www.dcsc.tudelft.nl/~disc_hs/course`
- Lecture notes: link will be sent via email
- Slides: see website
- Home works: see website
- Final grade: average of 4 home works
+ bonus points (by reporting errors)
results will be communicated by end of April 2017

1. Overview of the course (continued)

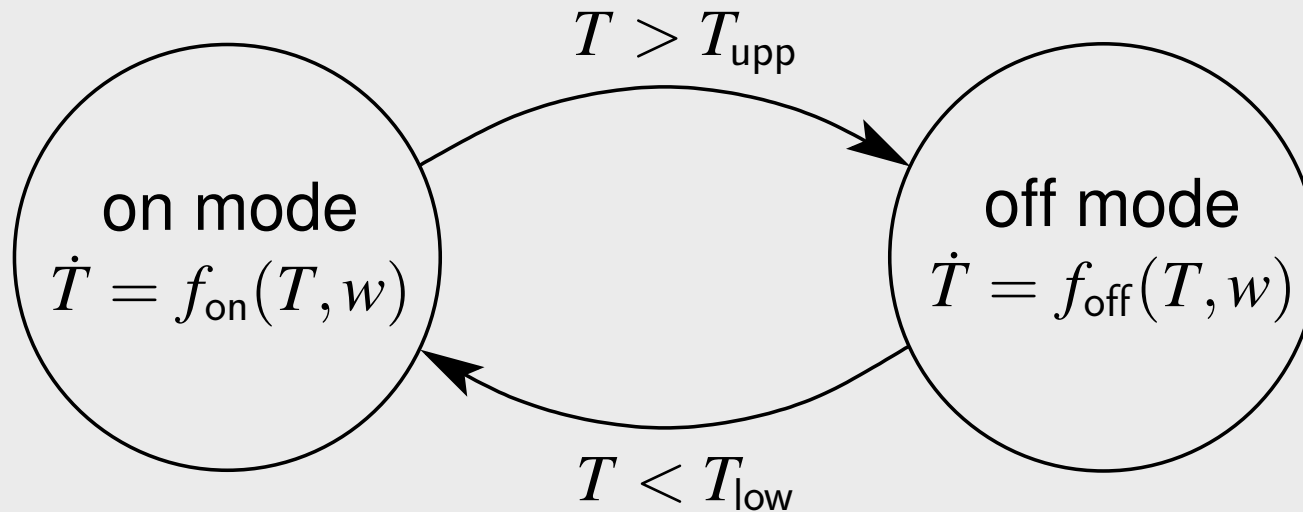
- Email addresses:
 - b.deschutter@tudelft.nl
 - w.p.m.h.heemels@tue.nl

Contents

1. Introduction (January 16)
2. Models (January 16)
3. Dynamics & well-posedness (January 23)
4. Stability (January 23 & 30)
5. Switched control (January 30)
6. Optimization-based control (February 6)
7. Model checking and timed automata (February 6)

2. Hybrid systems: Motivating examples

- Hybrid: combination of continuous and discrete dynamics
- Temperature control system:



Motivating examples

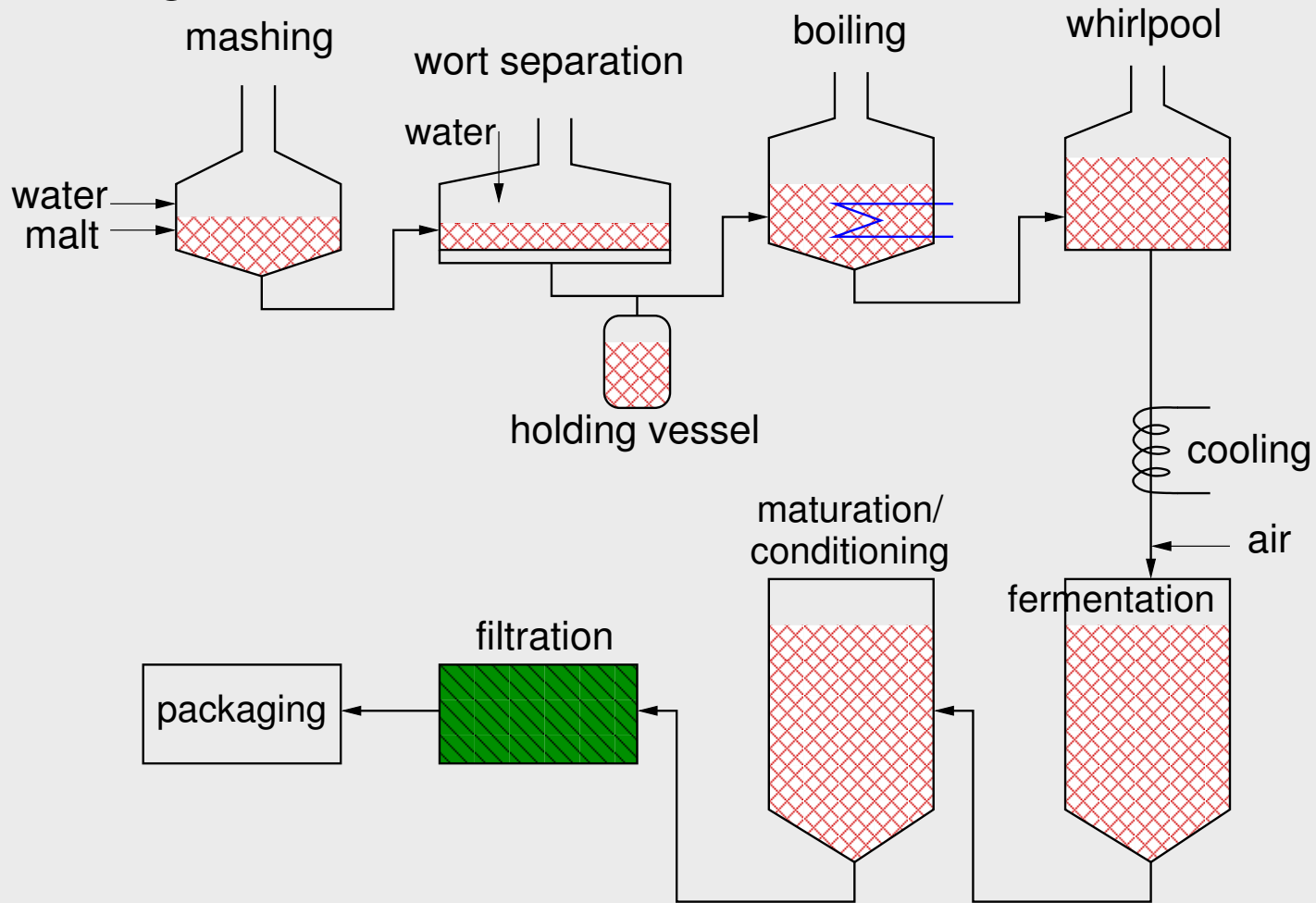
- Hierarchical control in process industry
- Telecommunication systems
- Manufacturing systems
- Airplane coordination control
- Beer brewing



Human intervention in smooth systems → hybrid

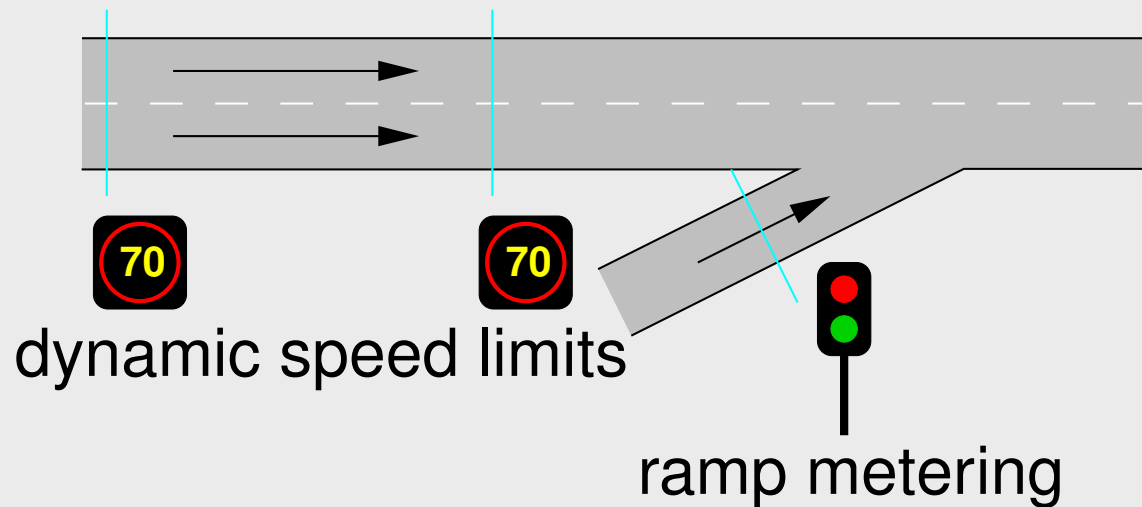
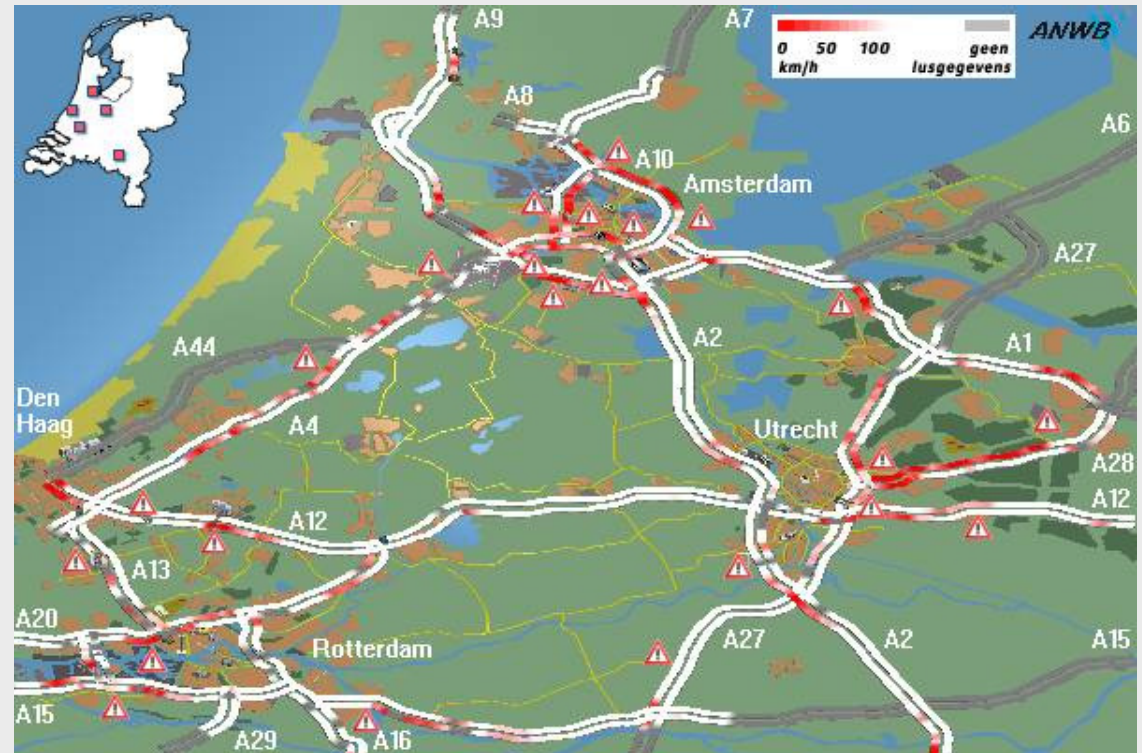
Motivating examples

- Beer brewing



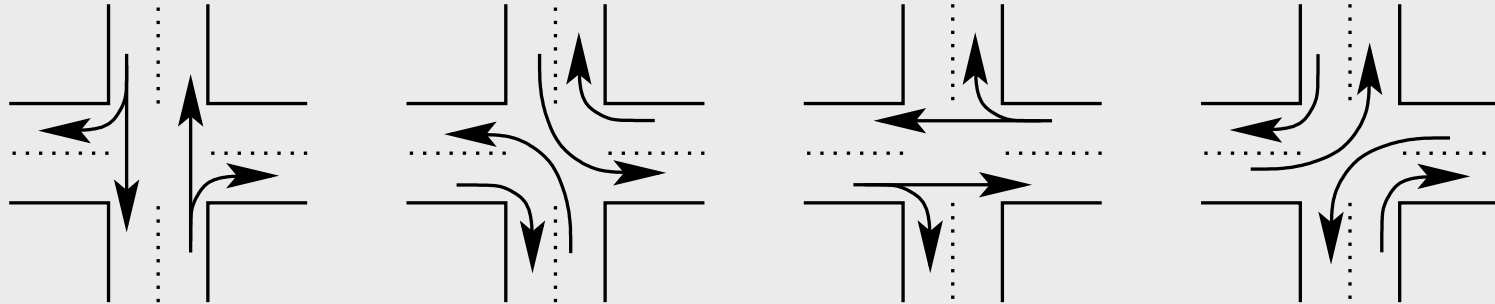
Motivating examples

- Traffic control systems



Motivating examples

- Intersection with traffic signals



4 modes, states: queue lengths

- Automatic platooning merging & splitting



Motivating examples

- Evolution of rigid bodies (contact/no contact)
- Electrical networks (switching, diodes)
- Fermentation process (lag, growth, stationary, inactivation)
- Saturation, hysteresis
- Actuator and sensor failures

Switching between dynamical regimes → hybrid

Challenges

- Analysis and control
- Nowadays:
 - often heuristic & ad-hoc
 - focus exclusively on either continuous or discrete dynamics
- structured approach necessary
- Consider hybrid nature of systems
- Combination of systems & control, computer science, mathematics, and simulation

3. Hybrid automata

“System”

- Example:

$$\dot{x}(t) = f(x(t), u(t))$$

t : time

x : state

u : input

- More formal definition:

$$x(\sigma) = \phi(\tau, \sigma, x(\tau), u)$$

τ : initial time

σ : current time

u : input function (over $[\tau, \sigma]$)

ϕ : transition map

Classification

- Continuous-state / discrete-state
- Continuous-time / discrete-time
- Time-driven / event-driven
 - time-driven → state changes as time progresses, i.e., continuously (for continuous-time), or at every tick of clock (for discrete-time)
 - event-driven → state changes due to occurrence of event event:
 - * start or end of an activity
 - * asynchronous (occurrence times not necessarily equidistant)

Combinations → “hybrid”

Models for time-driven systems

- Continuous-time time-driven systems:

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= g(x(t), u(t))\end{aligned}$$

- Discrete-time (or sampled) time-driven systems:

$$\begin{aligned}x(k+1) &= f(x(k), u(k)) \\ y(k) &= g(x(k), u(k))\end{aligned}$$

Models for event-driven systems

Automaton

Automaton is defined by triple $\Sigma = (\mathcal{Q}, \mathcal{U}, \phi)$ with

- \mathcal{Q} : finite or countable set of discrete states
- \mathcal{U} : finite or countable set of discrete inputs (“input alphabet”)
- $\phi : \mathcal{Q} \times \mathcal{U} \rightarrow P(\mathcal{Q})$: partial *transition function*.

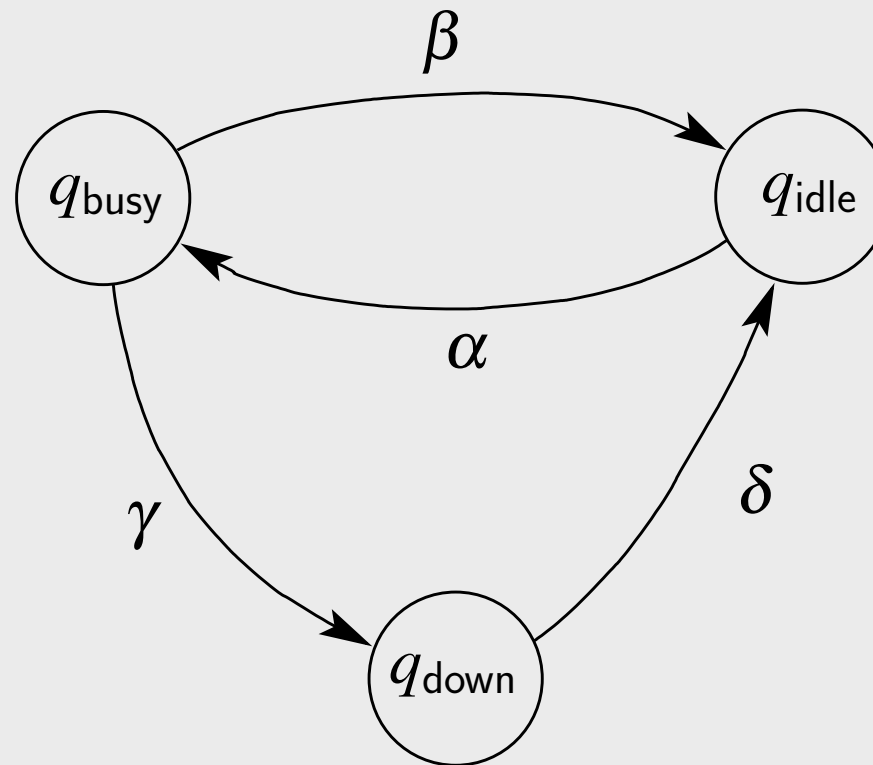
where $P(\mathcal{Q})$ is power set of \mathcal{Q} (set of all subsets)

Finite automaton: \mathcal{Q} and \mathcal{U} finite

Evolution of automaton

- Given state $q \in \mathcal{Q}$ and discrete input symbol $u \in \mathcal{U}$, transition function ϕ defines collection of next possible states:
 $\phi(q, u) \subseteq \mathcal{Q}$
- If **each** set of next states has 0 or 1 element:
→ “deterministic” automaton
- If **some** set of next states has more than 1 element:
→ “non-deterministic” automaton

Deterministic automaton



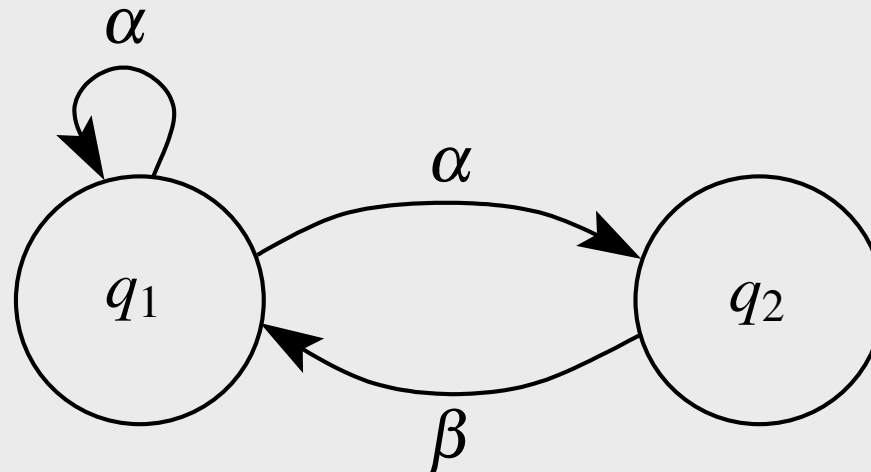
$$\phi(q_{\text{busy}}, \beta) = \{q_{\text{idle}}\}$$

$$\phi(q_{\text{busy}}, \gamma) = \{q_{\text{down}}\}$$

$$\phi(q_{\text{idle}}, \alpha) = \{q_{\text{busy}}\}$$

$$\phi(q_{\text{down}}, \delta) = \{q_{\text{idle}}\}$$

Non-deterministic automaton

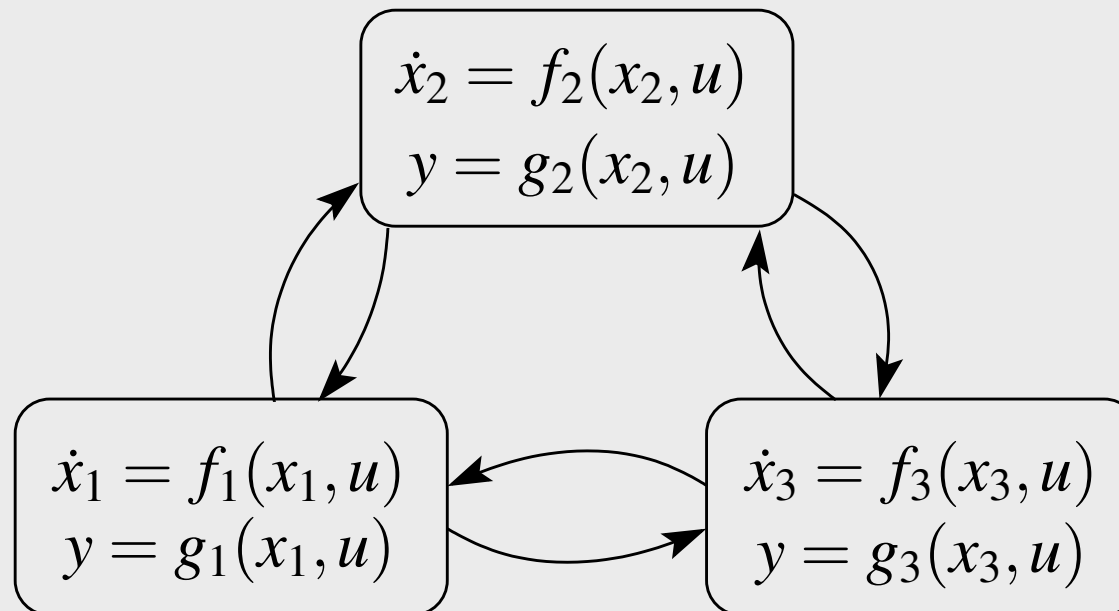


$$\phi(q_1, \alpha) = \{q_1, q_2\} \quad \phi(q_2, \beta) = \{q_1\}$$

→ unmodeled dynamics

Hybrid system

- System can be in one of several modes
- In each mode: behavior described by system of difference or differential equations
- Mode switches due to occurrence of “events”



Hybrid system

- At switching time instant:
 - possible state reset or state dimension change
- Mode transitions may be caused by
 - external control signal
 - internal control signal
 - dynamics of system itself (crossing of boundary in state space)

Models for hybrid systems

- 👉 • timed or hybrid Petri nets
 - differential automata
- 👉 • hybrid automata
 - Brockett's model
- 👉 • mixed logical dynamic models
 - real-time temporal logics
 - timed communicating sequential processes
 - switched bond graphs
 - predicate calculus
- 👉 • piecewise-affine models
 - . . .

Analysis techniques:

- formal verification
- computer simulation
- analytic techniques (for special subclasses)
- ...

⇒ no general modeling & analysis framework

modeling power ↔ decision power

+ computational complexity (NP-hard, undecidable)

⇒ special subclasses (Chapter 2)
hierarchical / modular approach

- **Undecidable problems**

- no algorithm at all can be given for solving the problem in general, i.e., finite termination cannot be guaranteed

- **NP-complete and NP-hard problems**

- *decision problem*: solution is either “yes” or “no”

- e.g., traveling salesman decision problem:

- Given a network of cities, intercity distances, and a number B , does there exist a tour with length $\leq B$?

- *search problem*

- e.g., traveling salesman problem:

- Given a network of cities, intercity distances, what is the shortest tour?

P and NP-complete decision problems

- Time complexity function $T(n)$: largest amount of time needed to solve problem instance of size n (*worst case!*)

- Polynomial time algorithm:

$$T(n) \leq |p(n)| \quad \text{for some polynomial } p$$

→ class P: solvable by polynomial time algorithm

- Nondeterministic computer:

– guessing stage (tour)

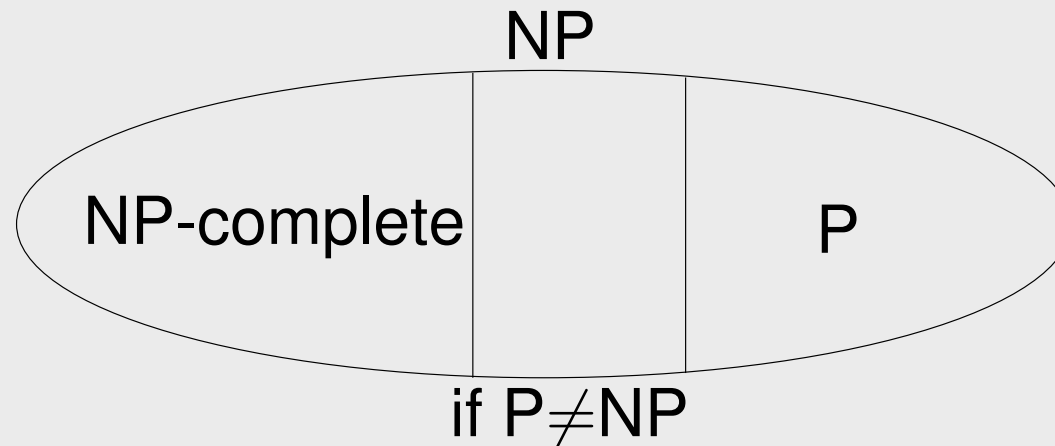
– checking stage (compute length of tour + compare it with B)

→ class NP: “*nondeterministically polynomial*”

i.e., time complexity of checking stage is polynomial

P and NP-complete decision problems

- Each problem in NP can be solved in exponential time: $T(n) \leq 2^{n^k}$
- NP-complete problems: “hardest” class in NP:
 - any NP-complete problem solvable in polynomial time
 \Rightarrow every problem in NP solvable in polynomial time
 - any problem in NP intractable
 \Rightarrow NP-complete problems also intractable



NP-hard problems

- Decision problem is NP-complete \Rightarrow search problem is NP-hard
- NP-hard problems: at least as hard as NP-complete problems
 - **NP-complete** (decision problem)
 - \rightarrow solvable in polynomial time *if and only if* $P = NP$
 - **NP-hard** (search problem)
 - \rightarrow cannot be solved in polynomial time *unless* $P = NP$

Examples of NP-hard and undecidable problems

- Consider simple hybrid system:

$$x(k+1) = \begin{cases} A_1 x(k) & \text{if } c^\top x(k) \geq 0 \\ A_2 x(k) & \text{if } c^\top x(k) < 0 \end{cases}$$

- deciding whether system is stable or not is NP-hard
- Given two Petri nets, do they have the same reachability set?
 - undecidable

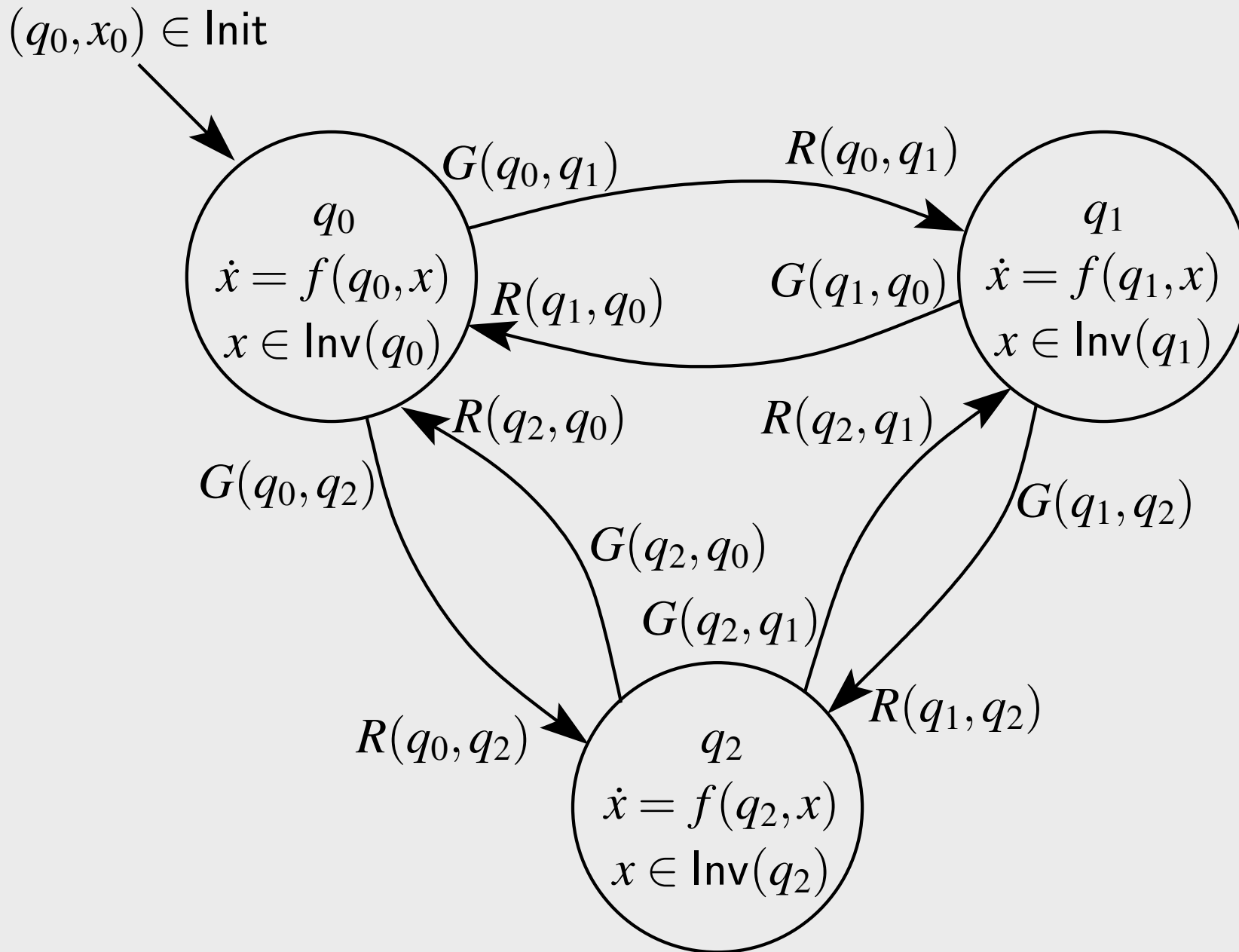
Hybrid automaton

Hybrid automaton H is collection $H = (Q, X, f, \text{Init}, \text{Inv}, E, G, R)$ where

- $Q = \{q_1, \dots, q_N\}$ is finite set of discrete states or *modes*
- $X = \mathbb{R}^n$ is set of continuous states
- $f : Q \times X \rightarrow X$ is vector field
- $\text{Init} \subseteq Q \times X$ is set of initial states
- $\text{Inv} : Q \rightarrow P(X)$ describes *invariants*
- $E \subseteq Q \times Q$ is set of edges or *transitions*
- $G : E \rightarrow P(X)$ is *guard condition*
- $R : E \rightarrow P(X \times X)$ is *reset map*

Hybrid automaton $H = (Q, X, f, \text{Init}, \text{Inv}, E, G, R)$

- Hybrid state: (q, x)
- Evolution of continuous state in mode q : $\dot{x} = f(q, x)$
- Invariant Inv : describes conditions that continuous state has to satisfy in given mode
- Guard G : specifies subset of state space where certain transition is enabled
- Reset map R : specifies how new continuous states are related to previous continuous states



Evolution of hybrid automaton

- Initial hybrid state $(q_0, x_0) \in \text{Init}$
- Continuous state x evolves according to

$$\dot{x} = f(q_0, x) \quad \text{with } x(0) = x_0$$

discrete state q remains constant: $q(t) = q_0$

- Continuous evolution can go on as long as $x \in \text{Inv}(q_0)$
- If at some point state x reaches guard $G(q_0, q_1)$, then
 - transition $q_0 \rightarrow q_1$ is enabled
 - discrete state *may* change to q_1 , continuous state then jumps from current value x^- to new value x^+ with $(x^-, x^+) \in R(q_0, q_1)$
- Next, continuous evolution resumes and whole process is repeated

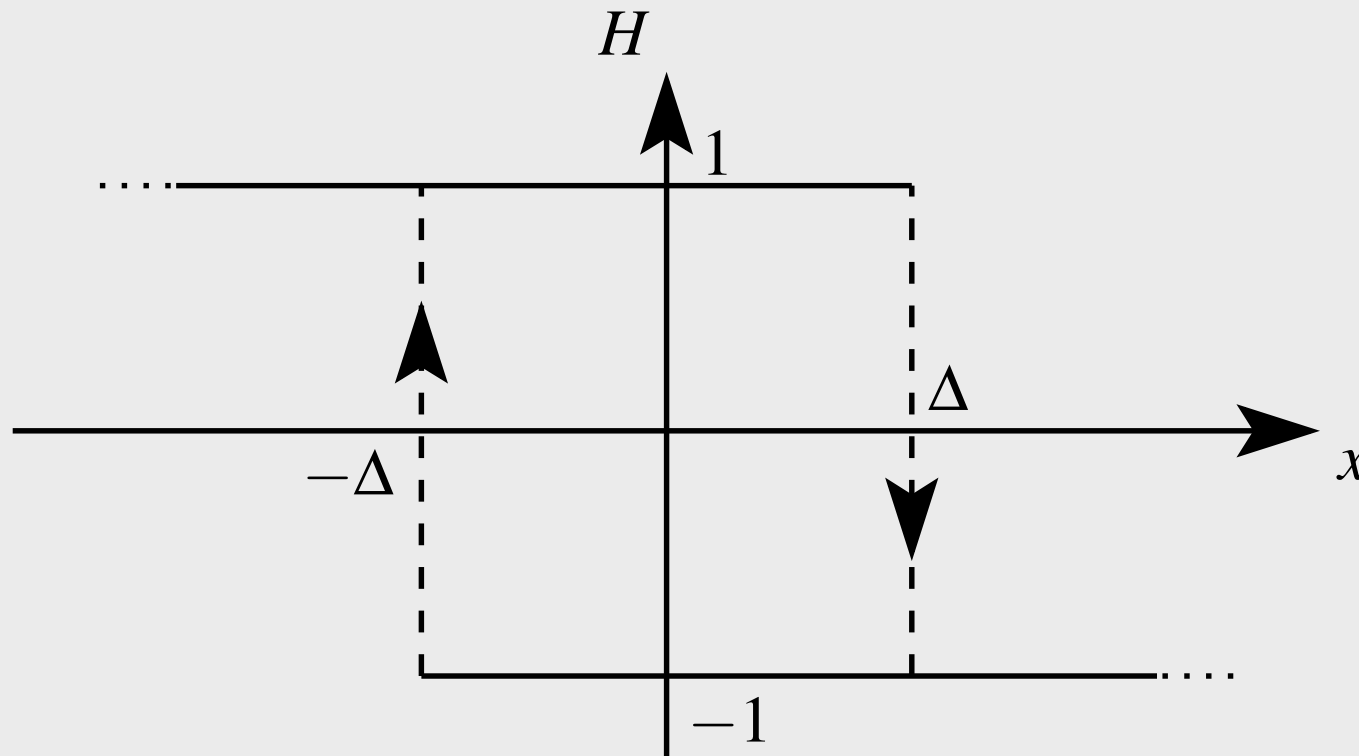
4. Examples of hybrid systems

1. Hysteresis
2. Manual transmission
3. Water-level monitor
4. Supervisor
5. Two-carts system
6. Boost converter

4.1 Hysteresis

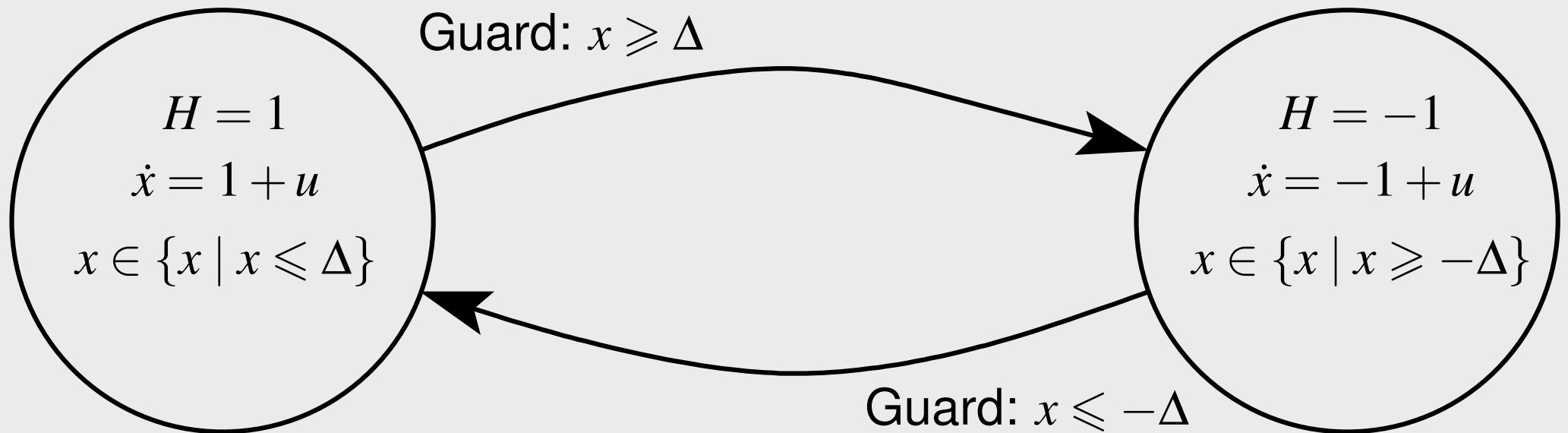
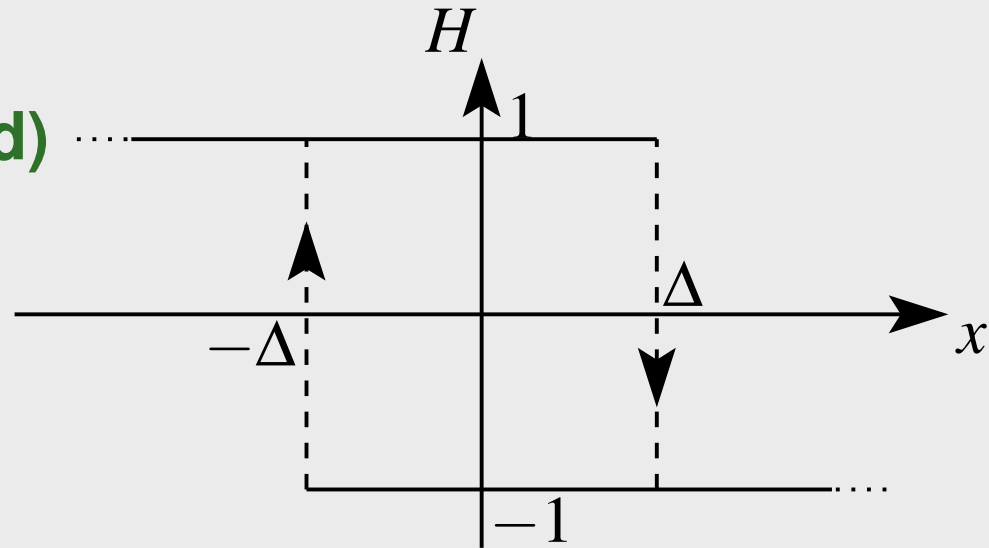
Control system with hysteresis element in the feedback loop :

$$\dot{x} = H(x) + u$$



4.1 Hysteresis (continued)

$$\dot{x} = H(x) + u$$



4.2 Manual transmission

Simple model of manual transmission

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{-ax_2 + u}{1 + v}\end{aligned}$$

with v : gear shift position $v \in \{1, 2, 3, 4\}$

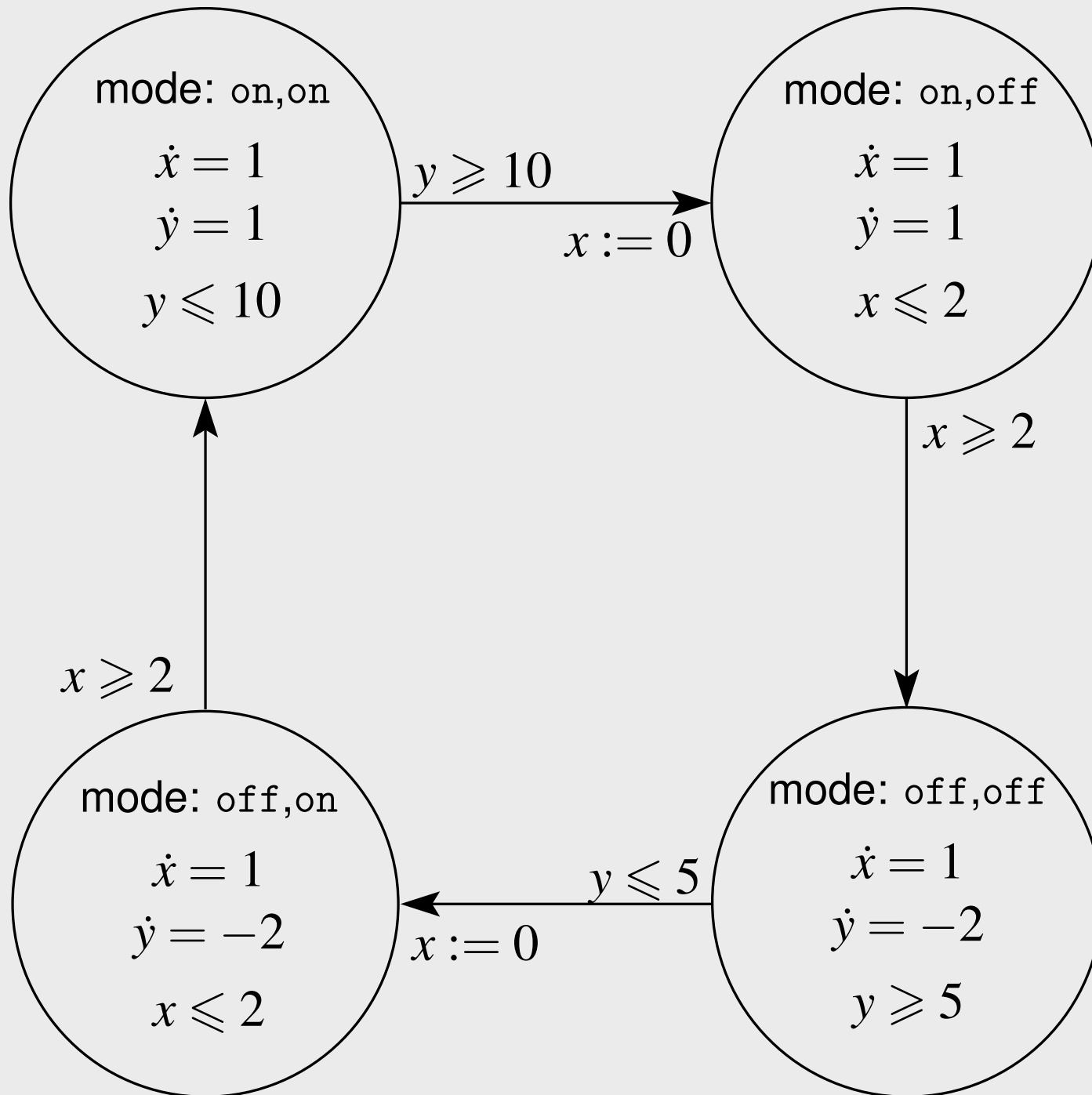
u : acceleration

a : parameter

→ hybrid system with four modes, 2-dimensional continuous state, controlled transitions (switchings), and no resets

4.3 Water-level monitor

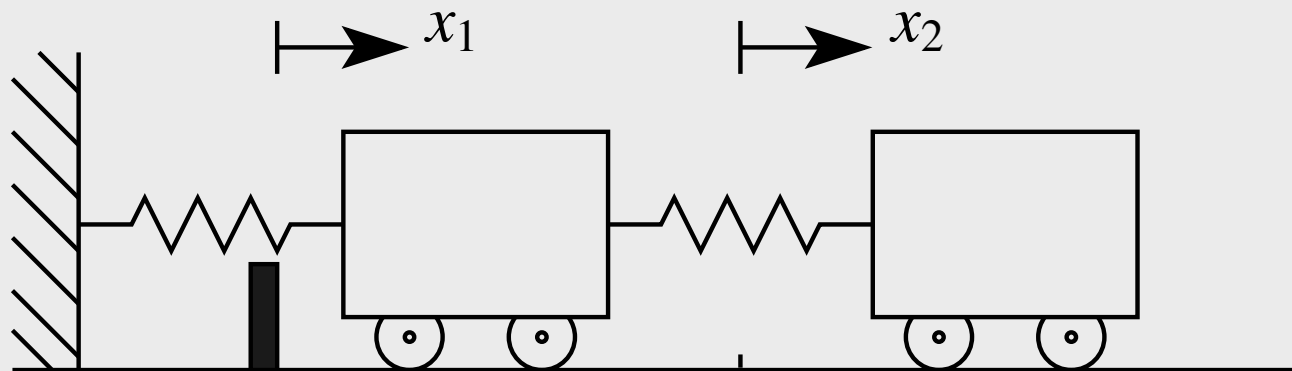
- Variables:
 - $y(t)$: water level, continuous
 - $x(t)$: time elapsed since last signal was sent by monitor, continuous
 - $P(t)$: status of pump, $\in \{\text{on}, \text{off}\}$
 - $S(t)$: nature of signal last sent by monitor, $\in \{\text{on}, \text{off}\}$
- Dynamics of system:
 - water level rises 1 unit per second when pump is on and falls 2 units per second when pump is off
 - when water level rises to 10 units, monitor sends switch-off signal; after delay of 2 seconds pump turns off
 - when water level falls to 5 units, monitor sends switch-on signal; after delay of 2 seconds pump switches on



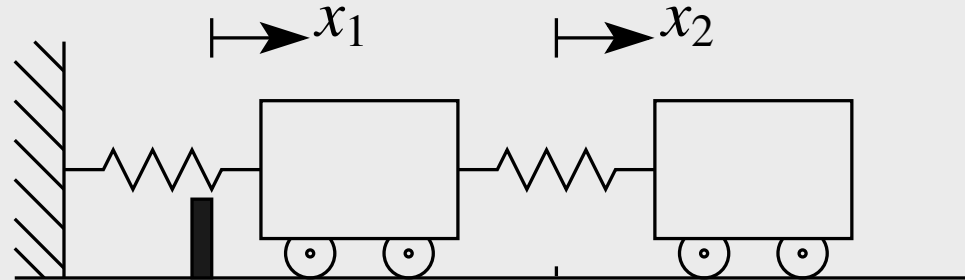
y : water level
 x : time since last signal

4.4 Two-carts system

- Two carts connected by spring
- Left cart attached to wall by spring;
motion constrained by completely inelastic stop
Stop is placed at equilibrium position of left cart
- Masses of carts and spring constants = 1

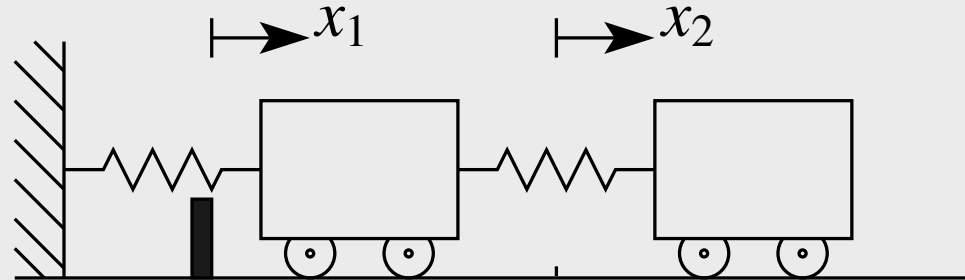


4.4 Two-carts system (continued)



- x_1, x_2 : deviations of left and right cart from equilibrium position
- x_3, x_4 : velocities of left and right cart
- z : reaction force exerted by stop
- Evolution: $\dot{x}_1(t) = x_3(t)$
 $\dot{x}_2(t) = x_4(t)$
 $\dot{x}_3(t) = -2x_1(t) + x_2(t) + z(t)$
 $\dot{x}_4(t) = x_1(t) - x_2(t)$

4.4 Two-carts system (continued)



To model stop:

- Define $w(t) = x_1(t)$
- $w(t) \geq 0$ (since w is position of left cart w.r.t. stop)
- Force exerted by stop can act only in positive direction $\rightarrow z(t) \geq 0$
- If left cart not at stop ($w(t) > 0$), reaction force vanishes: $z(t) = 0$
- If $z(t) > 0$ then cart must necessarily be at the stop: $w(t) = 0$

$$0 \leq w(t) \perp z(t) \geq 0$$

4.4 Two-carts system (continued)

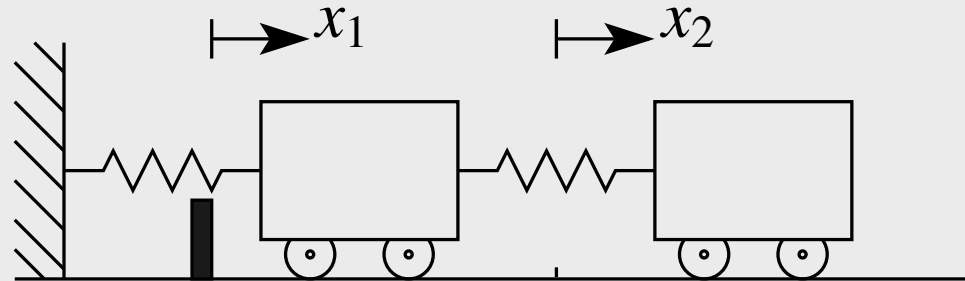
System can be represented by two modes (stop active or not)

$z = 0$	<u>unconstrained</u>	<u>constrained</u>	$w = 0$
	$\dot{x}_1(t) = x_3(t)$	$\dot{x}_1(t) = x_3(t)$	
	$\dot{x}_2(t) = x_4(t)$	$\dot{x}_2(t) = x_4(t)$	
	$\dot{x}_3(t) = -2x_1(t) + x_2(t)$	$\dot{x}_3(t) = -2x_1(t) + x_2(t) + z(t)$	
	$\dot{x}_4(t) = x_1(t) - x_2(t)$	$\dot{x}_4(t) = x_1(t) - x_2(t)$	
	$z(t) = 0$	$w(t) = x_1(t) = 0$	
	ODE (in state)	DAE (as z is not explicit)	

System stays in mode as long as

<u>unconstrained</u>	<u>constrained</u>
$z(t) = 0, w(t) > 0$	$w(t) = 0, z(t) > 0$

Mode transitions for two-carts system

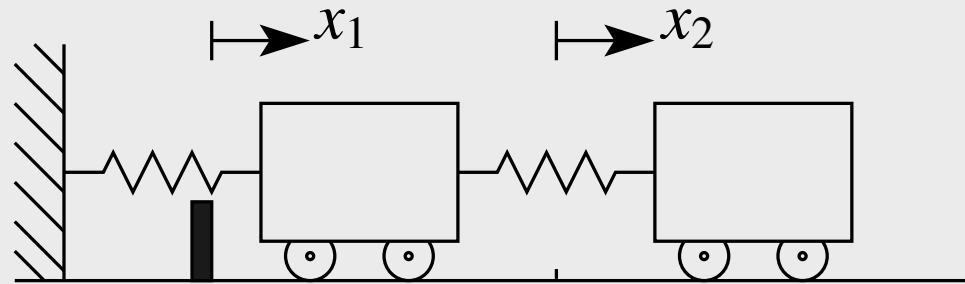


- **Unconstrained** \rightarrow **constrained**

Suppose $x(\tau) = (0^+, -1, -1, 0)^\top \rightarrow w(t) > 0$ tends to be violated
Left cart hits stop and stays there. Velocity of left cart is reduced to zero instantaneously (purely inelastic collision)

- **Constrained** \rightarrow **unconstrained**

Suppose $x(\tau) = (0, 0, 0, 1)^\top \rightarrow z(t) > 0$ tends to be violated
Right cart is moving to right of its equilibrium position, so spring between carts pulls left cart away from stop



- **Unconstrained** → **unconstrained with re-initialization according to constrained mode**

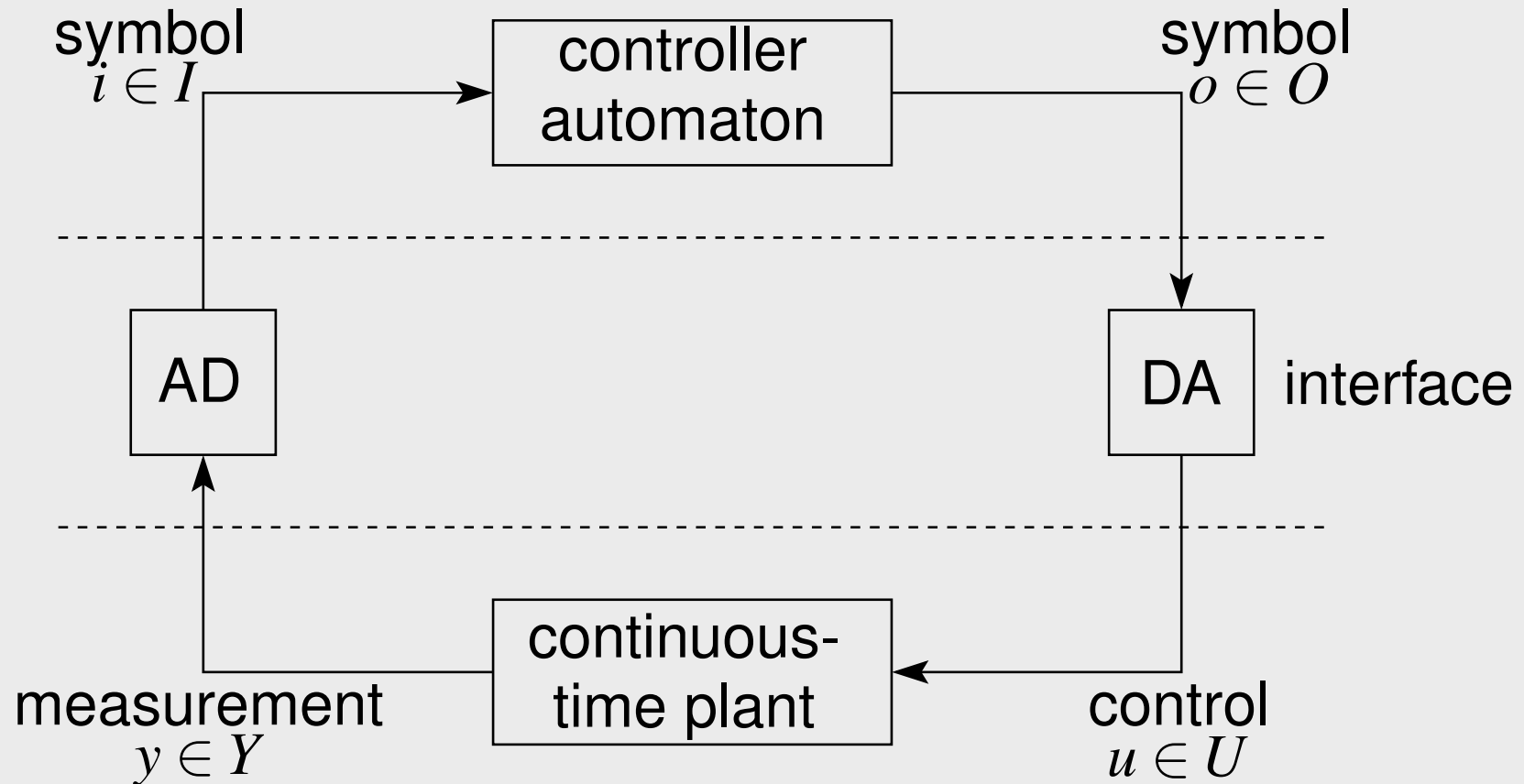
Consider $x(\tau) = (0^+, 1, -1, 0)^\top \rightarrow w(t) > 0$ tends to be violated

At impact, velocity of left cart is reduced to 0, i.e., state reset to $(0, 1, 0, 0)^\top$

Right cart is at right of its equilibrium position, pulls left cart away from stop → smooth continuation in unconstrained mode

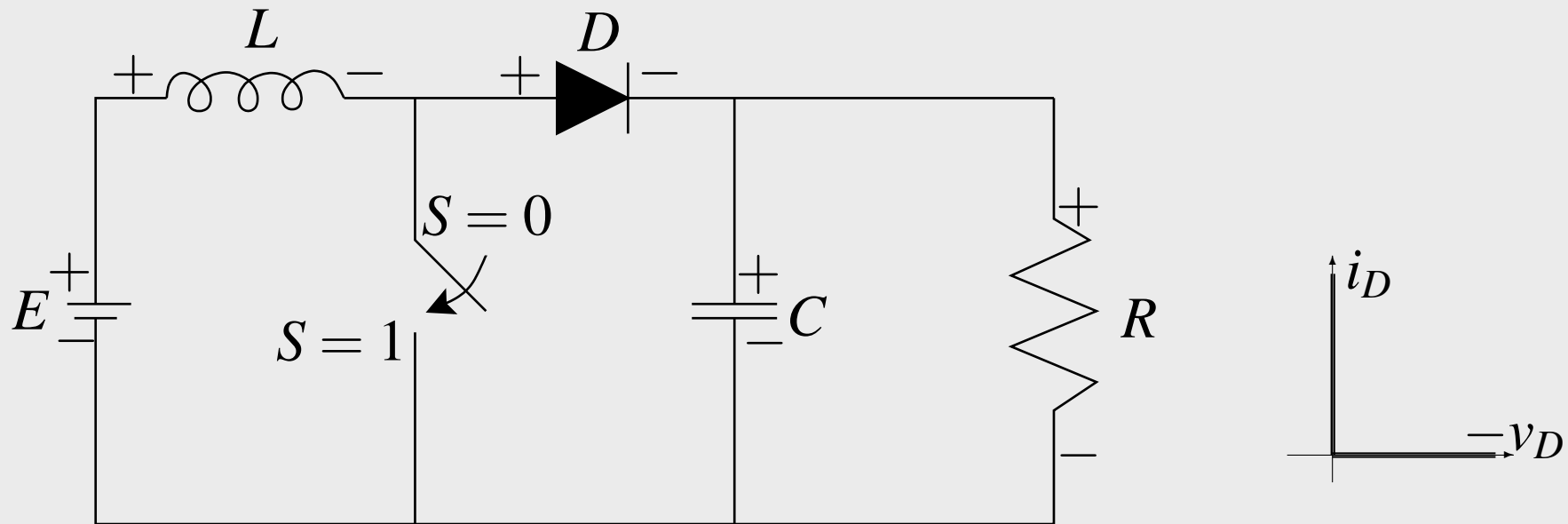
So: After the reset, no smooth continuation is possible in constrained mode → second mode change, back to unconstrained mode

4.5 Supervisor model



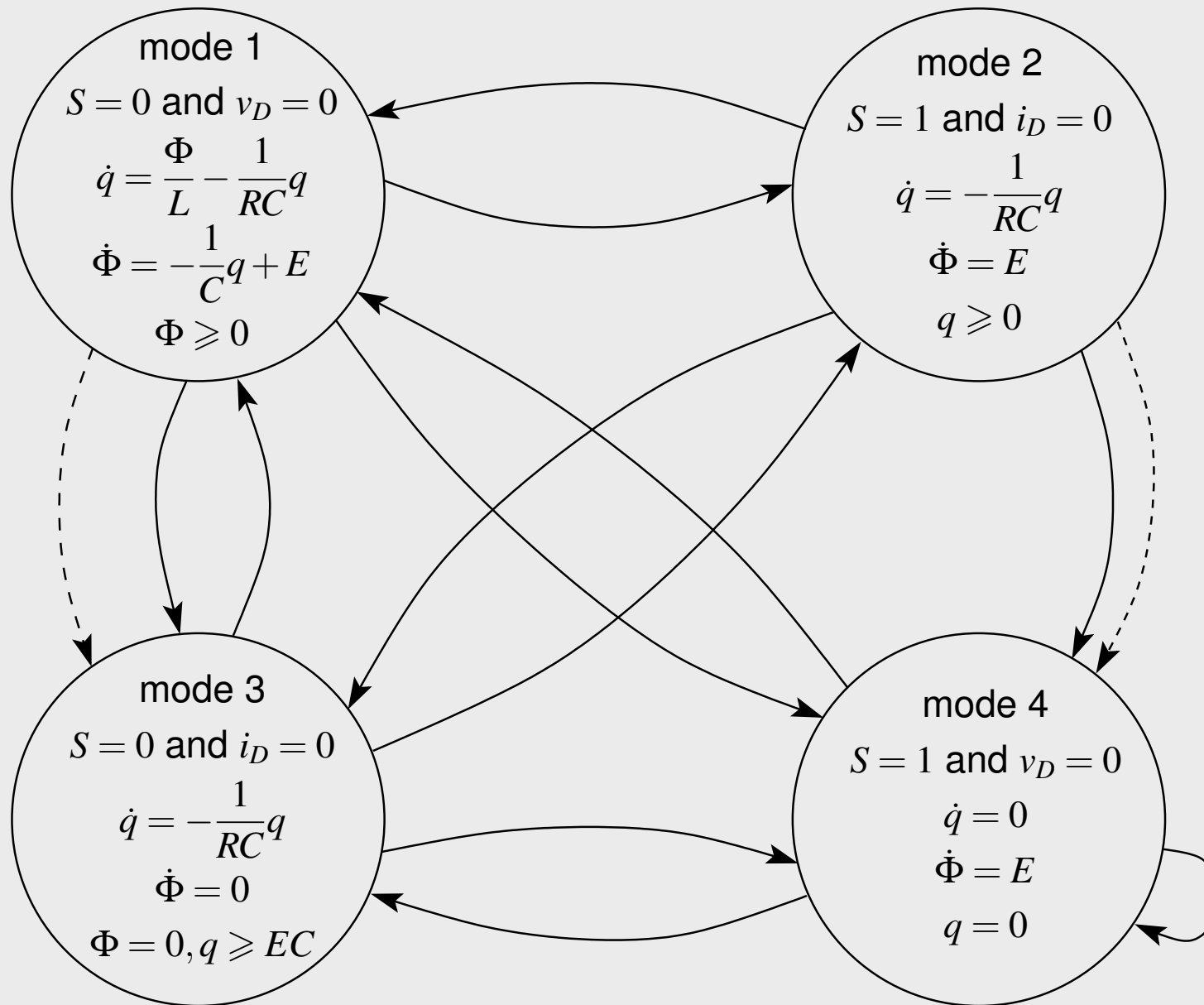
Controller is input-output automaton: $q^\# = v(q, i)$
 $o = \eta(q, i)$

4.6 Boost converter



- Presence of switch and diode introduces hybrid dynamics
- 4 modes:
 $(v_S = 0, v_D = 0)$, $(v_S = 0, i_D = 0)$, $(i_S = 0, v_D = 0)$, $(i_S = 0, i_D = 0)$

transition	guard	reset
mode 1 \rightarrow mode 2	$S = 1$ and $q \geq 0$	
mode 1 \rightarrow mode 3	$\phi = 0$ and $q > CE$	
mode 1 $--\rightarrow$ mode 3	$\phi < 0$	$\phi^+ = 0$
mode 1 \rightarrow mode 4	$S = 1$ and $q \leq 0$	$q^+ = 0$
mode 2 \rightarrow mode 1	$S = 0$ and $\phi \geq 0$	
mode 2 \rightarrow mode 3	$S = 0$ and $\phi \leq 0$	$\phi^+ = 0$
mode 2 \rightarrow mode 4	$q = 0$	
mode 2 $--\rightarrow$ mode 4	$q < 0$	$q^+ = 0$
mode 3 \rightarrow mode 1	$q = CE$	
mode 3 \rightarrow mode 2	$S = 1$ and $q \geq 0$	
mode 3 \rightarrow mode 4	$S = 1$ and $q \leq 0$	$q^+ = 0$
mode 4 \rightarrow mode 1	$S = 0$ and $\phi \geq 0$	
mode 4 \rightarrow mode 3	$S = 0$ and $\phi \leq 0$	$\phi^+ = 0$
mode 4 \rightarrow mode 4	$q < 0$	$q^+ = 0$



4.6 Boost converter (continued)

Hybrid automaton model is very involved

Alternatively, one may use the more compact model

$$\begin{aligned}\dot{q} &= -\frac{1}{RC}q + i_D \\ \dot{\phi} &= v_S + E \\ -v_D &= \frac{1}{C}q + v_S \\ i_S &= \frac{1}{L}\phi - i_D \\ 0 \leq i_D \perp -v_D \geq 0 \\ v_S \perp i_S\end{aligned}$$

→ also complementarity relation (as in two-carts system)

5. Examples with Zeno behavior

- *Zeno behavior*: infinitely many mode switches in finite time interval
- Examples
 1. bouncing ball
 2. reversed Filippov's system
 3. two-tank system
 4. three-balls example

5.1 Bouncing ball

- Dynamics: $\ddot{x} = -g$ subject to $x \geq 0$ ($x(t)$: height)
- Newton's restitution rule ($0 < e < 1$):

$$\dot{x}(\tau+) = -e\dot{x}(\tau-) \quad \text{when } x(\tau-) = 0, \dot{x}(\tau-) < 0$$

- Assuming $x(0) = 0, \dot{x}(0) > 0$, event times are related through

$$\tau_{i+1} = \tau_i + \frac{2e^i \dot{x}(0)}{g}$$

- Sequence has finite limit $\tau^* = \frac{2\dot{x}(0)}{g-ge} < \infty$ (geometric series)
- Physical interpretation: ball is at rest within finite time span, but after infinitely many bounces \rightarrow Zeno behavior

In this case: infinite number of state re-initializations, set of event times contains *right-accumulation point*

5.2 Reversed Filippov's example

- Dynamics:

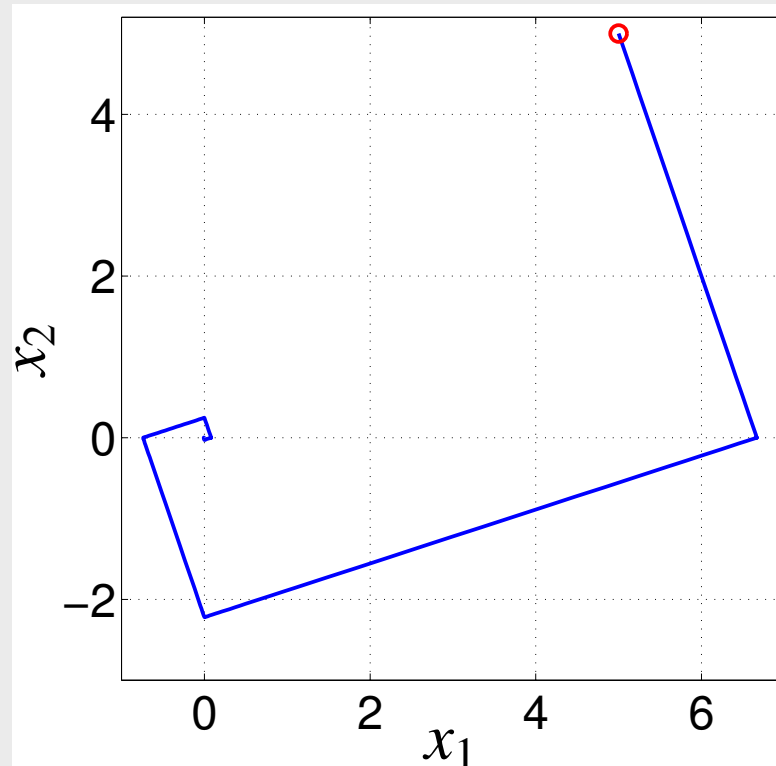
$$\begin{aligned}\dot{x}_1 &= -\operatorname{sgn}(x_1) + 2\operatorname{sgn}(x_2) \\ \dot{x}_2 &= -2\operatorname{sgn}(x_1) - \operatorname{sgn}(x_2),\end{aligned}$$

with

$$\begin{cases} \operatorname{sgn}(x) = 1 & \text{if } x > 0 \\ \operatorname{sgn}(x) = -1 & \text{if } x < 0 \\ \operatorname{sgn}(x) \in [-1, 1] & \text{when } x = 0 \end{cases}$$

- Solutions system are spiraling towards origin, which is an equilibrium

5.2 Reversed Filippov's example (continued)



- Since $\frac{d}{dt}(|x_1(t)| + |x_2(t)|) = -2$, solutions reach origin in finite time
- Solutions go through infinite number of mode transitions (relay switches) \rightarrow Zeno behavior

5.2 Reversed Filippov's example (continued)

- Dynamics:

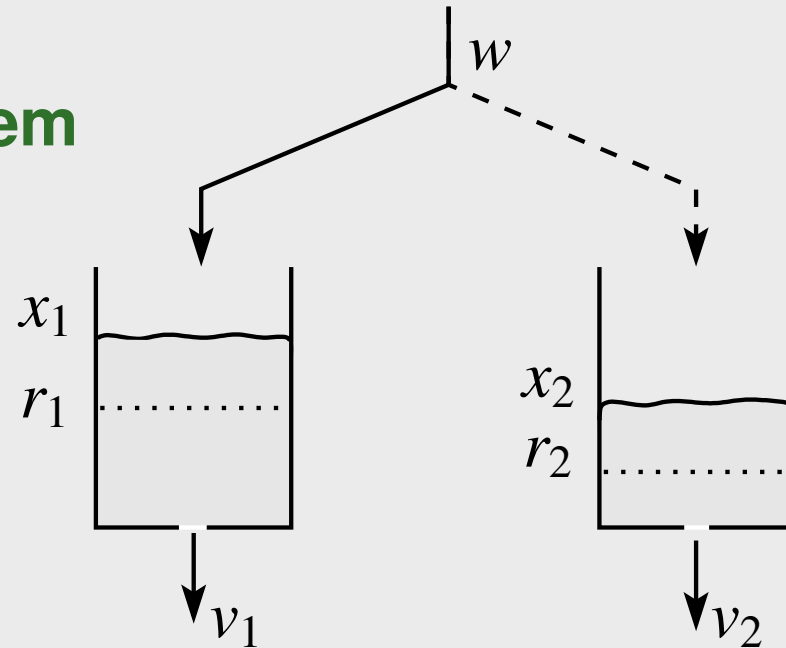
$$\begin{aligned}\dot{x}_1 &= -\operatorname{sgn}(x_1) + 2\operatorname{sgn}(x_2) \\ \dot{x}_2 &= -2\operatorname{sgn}(x_1) - \operatorname{sgn}(x_2),\end{aligned}$$

- “Derivative” of absolute value function: $\frac{d}{dx}|x| = \operatorname{sgn}(x)$

- So

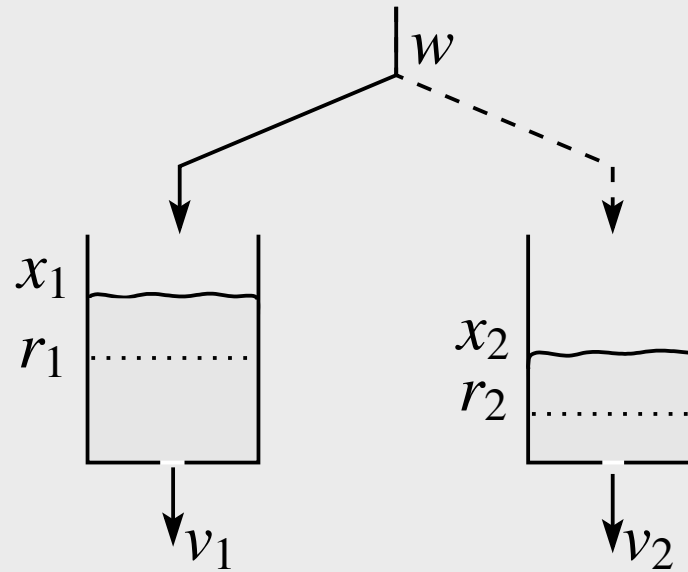
$$\begin{aligned}\frac{d}{dt}(|x_1(t)| + |x_2(t)|) &= \dot{x}_1 \operatorname{sgn}(x_1) + \dot{x}_2 \operatorname{sgn}(x_2) \\ &= -\operatorname{sgn}^2(x_1) + 2\operatorname{sgn}(x_2)\operatorname{sgn}(x_1) - 2\operatorname{sgn}(x_1)\operatorname{sgn}(x_2) - \operatorname{sgn}^2(x_2) \\ &= -1 - 1 \\ &= -2\end{aligned}$$

5.3 Two-tank system



- Two tanks (x_i : volume of water in tank)
- Tanks are leaking at constant rate $v_i > 0$
- Water is added at constant rate w through hose, which at any point in time is dedicated to either one tank or the other
- Objective: keep water volumes above r_1 and r_2
- Controller that switches inflow to tank 1 whenever $x_1 \leq r_1$ and to tank 2 whenever $x_2 \leq r_2$

Description of two-tank system as hybrid automaton



- Two modes: filling tank 1 (mode q_1) or tank 2 (mode q_2)
- Evolution of continuous state:

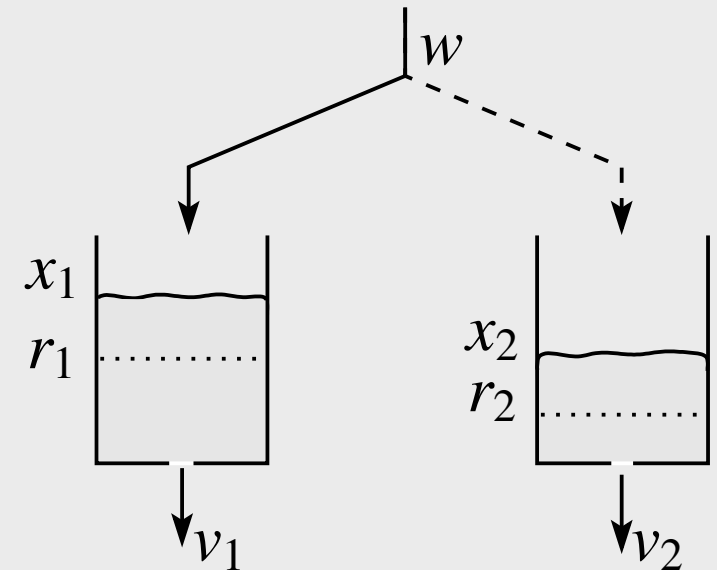
$$\begin{cases} \dot{x}_1 = w - v_1 \\ \dot{x}_2 = -v_2 \end{cases} \quad \text{in mode } q_1 \qquad \begin{cases} \dot{x}_1 = -v_1 \\ \dot{x}_2 = w - v_2 \end{cases} \quad \text{in mode } q_2$$

- $\text{Init} = \{q_1, q_2\} \times \{(x_1, x_2) \mid x_1 \geq r_1 \text{ and } x_2 \geq r_2\}$

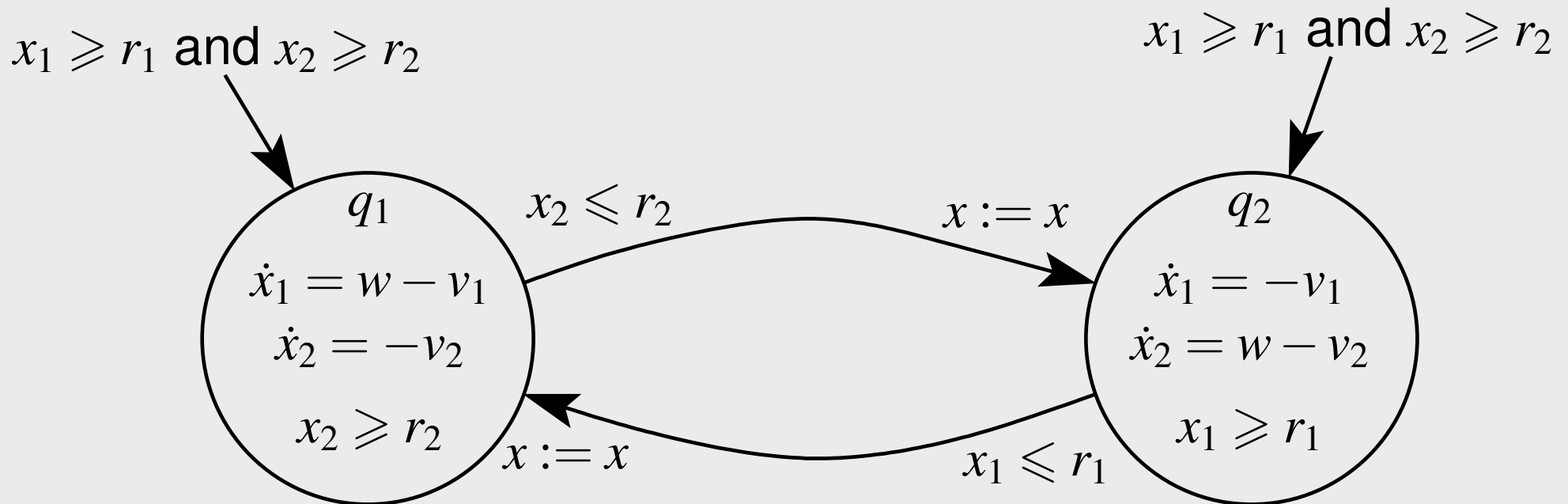
Description of two-tank system as hybrid automaton (cont.)

- Invariants: $\text{Inv}(q_1) = \{x \in \mathbb{R}^2 \mid x_2 \geq r_2\}$
 $\text{Inv}(q_2) = \{x \in \mathbb{R}^2 \mid x_1 \geq r_1\}$
- Guards: $G(q_1, q_2) = \{x \in \mathbb{R}^2 \mid x_2 \leq r_2\}$
 $G(q_2, q_1) = \{x \in \mathbb{R}^2 \mid x_1 \leq r_1\}$
- No resets:

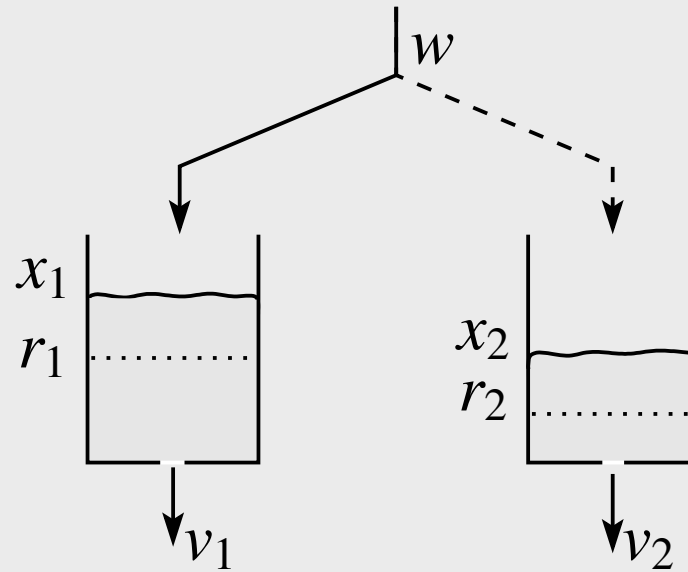
$$R(q_1, q_2) = R(q_2, q_1) = \{(x^-, x^+) \mid x^-, x^+ \in \mathbb{R}^2 \text{ and } x^- = x^+\}$$



Description of two-tank system as hybrid automaton (cont.)

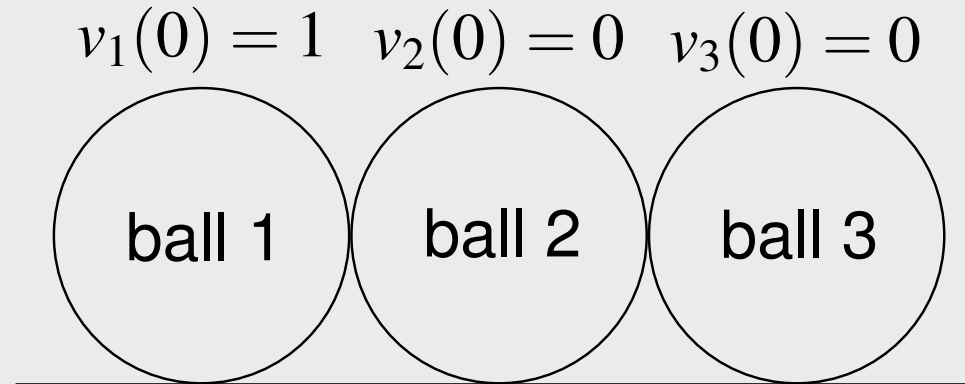


Two-tank system and Zeno behavior



- Assume total outflow $v_1 + v_2 > w$
- Control objective cannot be met and tanks will empty in finite time
- Infinitely many switchings in finite time \rightarrow Zeno behavior

5.4 Three-balls example



- System consisting of three balls
- Inelastic impacts modeled by successions of simple impacts
- Suppose unit masses, touching at time 0, and $v_1(0) = 1, v_2(0) = v_3(0) = 0$
- We model all impacts separately \rightarrow
 - first, inelastic collision between balls 1 and 2, resulting in $v_1(0+) = v_2(0+) = 0.5, v_3(0+) = 0$

5.4 Three-balls example (continued)

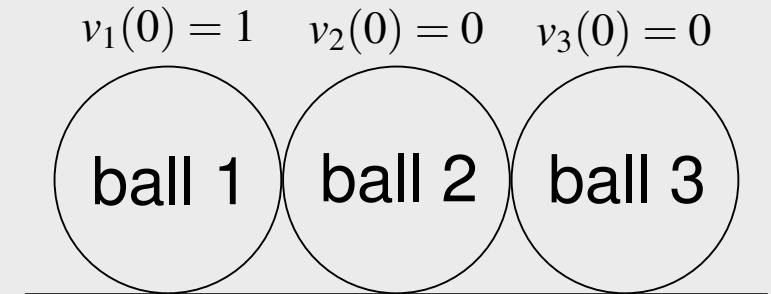
- next, ball 2 hits ball 3, resulting in $v_1(0++) = \frac{1}{2}$, $v_2(0++) = v_3(0++) = \frac{1}{4}$
- next, ball 1 hits ball 2 again, etc.

→ sequence of resets:

v_1	:	1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{3}{8}$	$\frac{3}{8}$	$\frac{11}{32}$...
v_2	:	0	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{3}{8}$	$\frac{5}{16}$	$\frac{11}{32}$...
v_3	:	0	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{5}{16}$	$\frac{5}{16}$...

converges to $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})^T$

- Afterwards, smooth continuation is possible with constant and equal velocity for all balls
- Infinite number of events (resets) at one time instant, sometimes called *live-lock* → another special case of Zeno behavior



6. Summary

- Definition and examples of hybrid systems
- Hybrid automaton
- Complexity issues: modeling power vs decision power
- Zeno behavior